Ildar Batyrshin
Miguel González Mendoza (Eds.)

LNAI 7629

Advances in Artificial Intelligence

11th Mexican International Conference on Artificial Intelligence, MICAI 2012
San Luis Potosí, Mexico, October/November 2012
Revised Selected Papers, Part I

Part I

MICAI
2012

Springer

Ildar Batyrshin
Miguel González Mendoza (Eds.)

Advances in Artificial Intelligence

11th Mexican International Conference
on Artificial Intelligence, MICAI 2012
San Luis Potosí, Mexico, October 27 – November 4, 2012
Revised Selected Papers, Part I

Springer

Volume Editors

Ildar Batyrshin
Mexican Petroleum Institute
Eje Central Lazaro Cardenas Norte, 152
Col. San Bartolo Atepehuacan
México D.F., CP 07730, Mexico
E-mail: batyr1@gmail.com

Miguel González Mendoza
Tecnológico de Monterrey
Campus Estado de México
Carretera Lago de Guadalupe Km 3.5
Atizapán de Zaragoza
Estado de México, CP 52926, Mexico
E-mail: mgonza@itesm.mx

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-37806-5 e-ISBN 978-3-642-37807-2
DOI 10.1007/978-3-642-37807-2
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2013935145

CR Subject Classification (1998): I.2.1, I.2.3-11, I.4, I.5.1-4, J.3, H.4.1-3,
F.2.2, H.3.3-5, H.5.3, F.1.1

LNCS Sublibrary: SL 7 – Artificial Intelligence

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

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

Preface

The Mexican International Conference on Artificial Intelligence (MICAI) is a yearly international conference series that has been organized by the Mexican Society of Artificial Intelligence (SMIA) since 2000. MICAI is a major international AI forum and the main event in the academic life of the country's growing AI community.

MICAI conferences publish high-quality papers in all areas of AI and its applications. The proceedings of the previous MICAI events have been published by Springer in its *Lecture Notes in Artificial Intelligence* (LNAI) series, vol. 1793, 2313, 2972, 3789, 4293, 4827, 5317, 5845, 6437, 6438, 7094, and 7095. Since its foundation in 2000, the conference has been growing in popularity and improving in quality.

The proceedings of MICAI 2012 are published in two volumes. The first volume, *Advances in Artificial Intelligence*, contains 40 papers structured into five sections:

- Machine Learning and Pattern Recognition
- Computer Vision and Image Processing
- Robotics
- Knowledge Representation, Reasoning, and Scheduling
- Medical Applications of Artificial Intelligence

The second volume, *Advances in Computational Intelligence*, contains 40 papers structured into four sections:

- Natural Language Processing
- Evolutionary and Nature-Inspired Metaheuristic Algorithms
- Neural Networks and Hybrid Intelligent Systems
- Fuzzy Systems and Probabilistic Models in Decision Making

Both books will be of interest for researchers in all fields of AI, students specializing in related topics, and for the general public interested in recent developments in AI.

The conference received for evaluation 224 submissions by 461 authors from 28 countries: Algeria, Austria, Belgium, Brazil, Canada, China, Colombia, Cuba, Czech Republic, France, Hungary, India, Iran, Israel, Japan, Luxembourg, Mexico, The Netherlands, New Zealand, Pakistan, Paraguay, Peru, Poland, Russia, Spain, Tunisia, UK, and USA. Of these submissions, 77 papers were selected for publication in these two volumes after a peer-reviewing process carried out by the international Program Committee. The acceptance rate was 34.3%.

MICAI 2012 was honored by the presence of such renowned experts as Ulises Cortés of the Universitat Politècnica de Catalunya, Spain; Joydeep Ghosh of the University of Texas, USA; Jixin Ma of Greenwich College, UK; Roy A. Maxion of

Carnegie Mellon University, USA; Grigori Sidorov of the Instituto Politécnico Nacional, Mexico; and Ian Witten of the University of Waikato, New Zealand, who gave excellent keynote lectures. The technical program of the conference also featured nine tutorials presented by Ulises Cortes, Alexander Gelbukh, Jean Bernard Hayet, Sergio Ledezma, Jixin Ma, Roy A. Maxion, Horacio Rostro González, Grigori Sidorov, and Ian Witten. Two workshops were held jointly with the conference: the 5th Workshop on Intelligent Learning Environments and the 5th Workshop on Hybrid Intelligent Systems.

In particular, in addition to regular papers, the volumes contain three invited papers by keynote speakers and their collaborators:

- "Empirical Study of Machine Learning-Based Approach for Opinion Mining in Tweets," by Grigori Sidorov, Sabino Miranda-Jiménez, Francisco Viveros-Jiménez, Alexander Gelbukh, Noé Castro-Sánchez, Francisco Velásquez, Ismael Díaz-Rangel, Sergio Suárez-Guerra, Alejandro Treviño, and Juan Gordon
- "AI-Based Fall Management Services. The Role of the i-Walker in I-DONTFALL," by Ulises Cortés, Antonio Martínez-Velasco, Cristian Barrué, and Roberta Annicchiarico
- "Syntactic Dependency-Based N-grams as Classification Features," by Grigori Sidorov, Francisco Velasquez, Efstathios Stamatatos, Alexander Gelbukh, and Liliana Chanona-Hernández

The authors of the following papers received the Best Paper Award on the basis of the paper's overall quality, significance, and originality of the reported results:

1st place: "Toward the Creation of Semantic Models Based on Computer-Aided Designs," by Nestor Velasco Bermeo, Miguel González Mendoza, Alexander García Castro, and Irais Heras Dueñas (Mexico, USA)

2nd place: "A New Branch and Bound Algorithm for the Cyclic Bandwidth Problem," by Hillel Romero-Monsivais, Eduardo Rodriguez-Tello, and Gabriel Ramírez (Mexico)

3rd place: "Modelling, Aggregation, and Simulation of a Dynamic Biological System Through Fuzzy Cognitive Maps," by Gonzalo Nápoles, Isel Grau, Maikel León, and Ricardo Grau (Cuba)

The authors of the following paper selected among all papers of which the first author was a full-time student, excluding the papers listed above, received the Best Student Paper Award:

1st place: "Fuzzy Clustering for Semi-Supervised Learning—Case Study: Construction of an Emotion Lexicon," by Soujanya Poria, Alexander Gelbukh, Dipankar Das, and Sivaji Bandyopadhyay (India, Mexico).

In addition, the attendees of the Special Session voted for the following works to receive the Best Poster Award (these papers are included in a separate Special Session proceedings volume):

1st place: "EMG Pattern Recognition System Based on Neural Networks," by Juan Carlos Gonzalez-Ibarra, Carlos Soubervielle-Montalvo, Omar Vital-Ochoa, and Hector Gerardo Perez-Gonzalez (Mexico)

2nd place: "Conflict Resolution in Multiagent Systems: Balancing Optimality and Learning Speed," by Aaron Rocha-Rocha, Enrique Munoz de Cote, Saul Pomares Hernandez, and Enrique Sucar Succar (Mexico)

3rd place: "Conception and Implementation of a Supermarket Shopping Assistant System," by Antonio Marin-Hernandez, Guillermo de Jesús Hoyos-Rivera, Marlon García-Arroyo, and Luis Felipe Marin-Urias (Mexico)

We want to thank all the people involved in the organization of this conference. In the first place, these are the authors of the papers published in this book: it is their research work that gives value to the book and to the work of the organizers. We thank the Track Chairs for their hard work, the Program Committee members and additional reviewers for their great effort spent on reviewing the submissions.

We would like to express our sincere gratitude to the Coordinación para la Innovación y Aplicación de la Ciencia y la Tecnología (CIACyT) of the Universidad Autónoma de San Luis Potosí, CONSYS TELECOM, ACOMEE, Magnetica FM, Faragauss System, Fundación Nikola Tesla, Panadería Penny, and Universidad Politécnica de San Luis Potosí (UPSLP) for their generous support, and in particular to UPSLP for providing the infrastructure for the keynote talks, tutorials, workshops, and technical presentations. We also thank the Oficina de Congresos y Convenciones de San Luis Potosí (OCCSLP) for its valuable effort in organizing the cultural program as well as other logistics activities.

We are deeply grateful to the conference staff and to all members of the Local Committee headed by Omar Montaño Rivas. In particular, we thank Rafael Llamas Contreras, the Logistics Chair, and Gabriela Zárate Rasillo, the OCCSLP representative, for their great effort in resolving logistics issues. Very special thanks go to Liliana Gámez Zavala and Ana María González Ávila for their help in advertizing MICAI and handling all publicity-related issues.

We are in indebted to José Antonio Loyola Alarcón, the Rector of the Universidad Politécnica de San Luis Potosí, for giving us the wonderful opportunity of organizing the conference at this university and for his unconditional support in the organization of the conference in all its stages. We would also like to express our sincere gratitude to the Vice Rector's Office of the UPSLP headed by Francisco Javier Delgado Rojas; to Igor León O'Farrill, Treasurer General of the UPSLP, and Francisco Cruz Oradaz Salazar, Head of the Facultad de Tecnologías de Información y Telemática, for their warm hospitality.

We gratefully acknowledge support received from the following projects: WIQ-EI (Web Information Quality Evaluation Initiative, European project 269180), PICCO10-120 (ICYT, Mexico City Government), and CONACYT 122030-DST India project “Answer Validation Through Textual Entailment.” The entire submission, reviewing, and selection process, as well as preparation of the proceedings, was supported for free by the EasyChair system (www.easychair.org). Last but not least, we are grateful to Springer for their patience and help in the preparation of this volume.

November 2012

Ildar Batyrshin
Miguel González Mendoza

Organization

Conference Organization

MICAI 2012 was organized by the Mexican Society of Artificial Intelligence (SMIA, Sociedad Mexicana de Inteligencia Artificial) in collaboration with the Universidad Politécnica de San Luis Potosí (UPSLP), Centro de Investigación en Computación del Instituto Politécnico Nacional (CIC-IPN), Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Universidad Nacional Autónoma de México (UNAM), Universidad Autónoma de México (UAM), Instituto Tecnológico de Estudios Superiores de Monterrey (ITESM), Universidad Autónoma de Estado de Hidalgo (UAEH), and Instituto Mexicano de Petroleo (IMP).

The MICAI series website is: www.MICAI.org. The website of the Mexican Society of Artificial Intelligence, SMIA, is: www.SMIA.mx. Contact options and additional information can be found on these websites.

Conference Committee

General Chairs	Raúl Monroy Alexander Gelbukh
Program Chairs	Miguel González Mendoza Ildar Batyrshin
Workshop Chair	Alexander Gelbukh
Tutorials Chairs	Felix Castro Espinoza
Keynote Talks Chair	Jesus A. Gonzalez
Financial Chair	Grigori Sidorov
Grant Chairs	Grigori Sidorov Miguel González Mendoza Ildar Batyrshin
Best Thesis Awards Chair	Miguel González Mendoza
Doctoral Consortium Chairs	Oscar Herrera Alcántara Miguel González Mendoza
Organizing Committee Chair	Omar Montaño Rivas
Supervising Ex-President of the SMIA	Carlos Alberto Reyes García

Track Chairs

Natural Language Processing	Sofia Natalia Galicia Haro
Machine Learning and Pattern Recognition	Johan van Horebeek
Data Mining	Felix Castro Espinoza

Intelligent Tutoring Systems	Alexander Gelbukh
Evolutionary and Nature-Inspired Metaheuristic Algorithms	Nareli Cruz Cortés Oliver Schütze
Computer Vision and Image Processing	Oscar Herrera Alcántara
Robotics, Planning and Scheduling	Fernando M. MontesGonzalez
Neural Networks and Hybrid Intelligent Systems	Sergio Ledesma Orozco
Logic, Knowledge-Based Systems	Mauricio Osorio
Multi-Agent Systems, and Distributed AI	Claudia Zepeda Cortés
Fuzzy Systems and Probabilistic Models in Decision Making	Alexander Tulupyev
Bioinformatics and Medical Applications	Jesús A. González

Program Committee

Charan Kumari Achanta
Carlos Acosta
Hector-Gabriel Acosta-Mesa
Ruth Aguilar
Carlos Aguilar Ibáñez
Teresa Alarcón
Alfonso Alba
Rafik Aliev
Leopoldo Altamirano
Gustavo Arroyo
Serge Autexier
J. Gabriel Avina-Cervantes
Victor Ayala-Ramirez
Bogdan Babych
Sivaji Bandyopadhyay
Maria Lucia Barrón-Estrada
Ildar Batyrshin
Hector Becerra
Mokhtar Beldjehem
Leopoldo Bertossi
Albert Bifet
Leticia Cagnina
Felix Calderon
Hiram Calvo
Nicoletta Calzolari
Oscar Camacho Nieto
Jose Luis Carballido
Mario Castelán
Oscar Castillo
Felix Castro Espinoza
Martine Ceberio
Gustavo Cerda-Villafana
Mario Chacon
Niladri Chatterjee
Edgar Chavez
Zhe Chen
Stefania Costantini
Broderick Crawford
Nareli Cruz Cortés
Rolando Cruz Ramírez
Heriberto Cuayahuitl
Iria da Cunha
Oscar Dalmau
Enrique de La Rosa Miranda
Carlos Delgado-Mata
Bernabe Dorronsoro

Beatrice Duval
Asif Ekbal
Michael Emmerich
Alexandr Eremeev
Hugo-Jair Escalante
Boris Escalante Ramírez
Ponciano Jorge Escamilla-Ambrosio
Susana C. Esquivel
Vladimir Estivill-Castro
Julio Cesar Estrada Rico
Gibran Etcheverry
Eugene C. Ezin
Claudia Feregrino
Denis Filatov
Juan J. Flores
Andrea Formisano
Anilu Franco-Arcega
Alfredo Gabaldon
Sofia N. Galicia-Haro
Ana-Gabriela Gallardo-Hernández
Carlos Hugo Garcia Capulin
Ma. de Guadalupe Garcia-Hernandez
Alexander Gelbukh
Arturo Gonzalez
Jesus A. Gonzalez
Miguel Gonzalez Mendoza
José-Joel Gonzalez-Barbosa
Felix F. Gonzalez-Navarro
D. Gutiérrez
Joaquín Gutiérrez Jagüey
Rafael Guzman
Hartmut Haehnel
Jin-Kao Hao
Yasunari Harada
Rogelio Hasimoto
Jean-Bernard Hayet
Antonio Hernandez
Donato Hernández
Arturo Hernandez-Aguirre
Oscar Herrera
Dieter Hutter
Pablo H. Ibarguengoytia
Mario Alberto Ibarra-Manzano
Hugo Jair Escalante
Berend Jan van der Zwaag
Timoleon Kipouros
Ryszard Klempous
Olga Kolesnikova
Vladik Kreinovich
Reinhard Langmann
Adriana Lara
Bruno Lara
Yulia Ledeneva
Sergio Ledesma-Orozco
Yoel Ledo Mezquita
Pierrick Legrand
Guillermo Leguizamón
Eugene Levner
Aurelio Lopez
Omar Lopez
Juan Manuel Lopez Hernandez
Francisco Luna
Gabriel Luque
Tanja Magoc
Luis Ernesto Mancilla
Nashat Mansour
Alexis Marcano Cedeño
J. Raymundo Marcial-Romero
Antonio Marin Hernandez
Luis Felipe Marin Urias
Georgiana Marsic
Luis Martí
Rene Alfredo Martinez Celorio
Edgar Martinez-Garcia
José Fco. Martínez-Trinidad
Jerzy Martyna
Carolina Medina
Patricia Melin
Martin Mendez
Carlos Merida-Campos
Ivan V. Meza
Efrén Mezura-Montes
Mikhail Mikhailov
Gabriela Minetti
Dieter Mitsche
Luís Moniz Pereira
Raul Monroy
Fernando Martin Montes-Gonzalez
Manuel Montes-y-Gómez
Guillermo Morales-Luna

Victor Muñiz
Masaki Murata
Juan Antonio Navarro Perez
Jesús Emeterio Navarro-Barrientos
Juan Carlos Nieves
Sergey Nikolenko
Juan Arturo Nolazco Flores
Leszek Nowak
Iván Olier
Ivan Olmos
Fernando Orduña Cabrera
Felipe Orihuela-Espina
Eber Orozco
Magdalena Ortiz
Mauricio Osorio
Elvia Palacios
Ted Pedersen
Héctor Manuel Pérez
Thierry Peynot
David Pinto
Eunice E. Ponce-de-Leon
Natalia Ponomareva
Volodymyr Ponomaryov
Edgar Alfredo Portilla-Flores
Joel Quintanilla
Julio Cesar Ramos
Carolina Reta
Alberto Reyes
Orion Fausto Reyes-Galaviz
Carlos A. Reyes-Garcia
Jean-Michel Richer
Mariano Rivera
Antonio Robles-Kelly
Erik Rodner
Arles Rodriguez
Eduardo Rodriguez-Tello
Katya Rodriguez-Vazquez
Leandro Fermín Rojas Peña
Paolo Rosso
Horacio Rostro Gonzalez
Jianhua Ruan
Imre Rudas
Salvador Ruiz Correa
Jose Ruiz-Pinales
Leszek Rutkowski
Andriy Sadovnychyy
Carolina Salto
Gustavo Sanchez
Guillermo Sanchez-Diaz
Abraham Sánchez López
Raul E. Sanchez-Yanez
Jose Santos
Oliver Schütze
Nikolay Semenov
Pinar Senkul
Shahnaz N. Shahbazova
Grigori Sidorov
Gerardo Sierra
Alexander Sirotkin
Peter Sosnin
Humberto Sossa Azuela
Marta Rosa Soto
Ramon Soto
Ricardo Soto
Juan-Manuel Torres-Moreno
Heike Trautmann
Leonardo Trujillo
Alexander Tulupyev
Johan van Horebeek
Massimiliano Vasile
Francois Vialatte
Javier Vigueras
Manuel Vilares Ferro
Andrea Villagra
Toby Walsh
Cornelio Yáñez-Márquez
Ramon Zatarain
Claudia Zepeda Cortes

Additional Reviewers

Mehran Ahmadlou
Israel Becerra
Eduardo Cabal
Nina Dethlefs
Elva Diaz
Christian Dominguez Medina
Milagros Fernández Gavilanes
Benjamin Hernandez
Md. Jamiul Jahid
Rafael Murrieta
José Carlos Ortiz-Bayliss
Partha Pakray
Obdulia Pichardo
Soujanya Poria
Francisco Jose Ribadas-Pena
Edgar Rodriguez
Jose Rodriguez
Octavio Augusto Sánchez Velázquez
José Arturo Tejeda-Gómez

Organizing Committee

Local Chair	Omar Montaño Rivas
Logistics Chairs	Rafael Llamas Contreras Cesar Guerra
Registration Chairs	Hugo Gonzalez Victor Fernández
Sponsorship Chair	Jorge Simón
Publicity Chair	Liliana Gámez
Financial Chair	Mario Alberto Martínez

Table of Contents – Part I

Machine Learning and Pattern Recognition

Invited Paper

Empirical Study of Machine Learning Based Approach for Opinion Mining in Tweets 1
Grigori Sidorov, Sabino Miranda-Jiménez, Francisco Viveros-Jiménez, Alexander Gelbukh, Noé Castro-Sánchez, Francisco Velásquez, Ismael Díaz-Rangel, Sergio Suárez-Guerra, Alejandro Treviño, and Juan Gordon

An Empirical Evaluation of Different Initializations on the Number of K-Means Iterations 15
Renato Cordeiro de Amorim

Intelligent Feature and Instance Selection to Improve Nearest Neighbor Classifiers 27
Yenny Villuendas-Rey, Yailé Caballero-Mota, and María Matilde García-Lorenzo

A Method for Building Prototypes in the Nearest Prototype Approach Based on Similarity Relations for Problems of Function Approximation 39
Marilyn Bello-García, María Matilde García-Lorenzo, and Rafael Bello

A Diversity Production Approach in Ensemble of Base Classifiers 51
Hamid Parvin, Sara Ansari, and Sajad Parvin

A New Overlapping Clustering Algorithm Based on Graph Theory 61
Airel Pérez-Suárez, José Fco. Martínez-Trinidad, Jesús A. Carrasco-Ochoa, and José E. Medina-Pagola

Fuzzy Clustering for Semi-supervised Learning – Case Study: Construction of an Emotion Lexicon 73
Soujanya Poria, Alexander Gelbukh, Dipankar Das, and Sivaji Bandyopadhyay

Statistical Framework for Facial Pose Classification 87
Ajay Jaiswal, Nitin Kumar, and R.K. Agrawal

Forecast of Air Quality Based on Ozone by Decision Trees and Neural Networks 97
Nahun Loya, Iván Olmos Pineda, David Pinto, Helena Gómez-Adorno, and Yuridiana Alemán

Automatic Discovery of Web Content Related to IT in the Mexican Internet Based on Supervised Classifiers 107
José-Lázaro Martínez-Rodríguez, Víctor-Jesús Sosa-Sosa, and Iván López-Arévalo

Automatic Monitoring the Content of Audio Broadcasted by Internet Radio Stations 119
Omar Nuñez and Antonio Camarena-Ibarrola

Computer Vision and Image Processing

Using Hidden Markov Model and Dempster-Shafer Theory for Evaluating and Detecting Dangerous Situations in Level Crossing Environments 131
Houssam Salmane, Yassine Ruichek, and Louahdi Khoudour

Local Features Classification for Adaptive Tracking 146
Alan I. Torres-Nogales, Santiago E. Conant-Pablos, and Hugo Terashima-Marín

Two Adaptive Methods Based on Edge Analysis for Improved Concealing Damaged Coded Images in Critical Error Situations 158
Alejandro Alvaro Ramírez-Acosta and Mireya S. García-Vázquez

Phase Correlation Based Image Alignment with Subpixel Accuracy 171
Alfonso Alba, Ruth M. Aguilar-Ponce, Javier Flavio Vigueras-Gómez, and Edgar Arce-Santana

A Novel Encryption Method with Associative Approach for Gray-Scale Images 183
Federico Felipe, Ángel Martínez, Elena Acevedo, and Marco Antonio Acevedo

FPGA-Based Architecture for Extended Associative Memories and Its Application in Image Recognition 194
Guzmán-Ramírez Enrique, Arroyo-Fernández Ignacio, González-Rojas Carlos, Linares-Flores Jesús, and Pogrebnyak Oleksiy

Salient Features Selection for Multiclass Texture Classification 205
Bharti Rana and R.K. Agrawal

Robotics

Robotic Behavior Implementation Using Two Different Differential Evolution Variants 216
Víctor Ricardo Cruz-Álvarez, Fernando Montes-Gonzalez, Efrén Mezura-Montes, and José Santos

Robust Visual Localization of a Humanoid Robot in a Symmetric Space 227
Mauricio J. García Vazquez, Jorge Francisco Madrigal, Oscar Mar, Claudia Esteves, and Jean-Bernard Hayet

DeWaLoP In-Pipe Robot Position from Visual Patterns 239
Luis A. Mateos and Markus Vincze

Electric Vehicle Automation through a Distributed Control System for Search and Rescue Operations 249
Marco Polo Cruz-Ramos, Christian Hassard, and J.L. Gordillo

Determination of the Instantaneous Initial Contact Point on a Parallel Gripper Using a Multi Input Fuzzy Rules Emulated Network Controller with Feedback from Ultrasonic and Force Sensors 261
César Navarro, Chidentree Treesatayapun, and Arturo Baltazar

Integration of Directional Smell Sense on an UGV 273
B. Lorena Villarreal, Christian Hassard, and J.L. Gordillo

Automatic 3D City Reconstruction Platform Using a LIDAR and DGPS 285
Angel-Iván García-Moreno, José-Joel Gonzalez-Barbosa, Francisco-Javier Ornelas-Rodríguez, Juan-Bautista Hurtado-Ramos, Alfonso Ramirez-Pedraza, and Erick-Alejandro González-Barbosa

Knowledge Representation, Reasoning, and Scheduling

Towards the Creation of Semantic Models Based on Computer-Aided Designs 298
Nestor Velasco Bermeo, Miguel González Mendoza, Alexander García Castro, and Irais Heras Dueñas

Process of Concept Alignment for Interoperability between Heterogeneous Sources 311
María de Lourdes Guadalupe Martínez-Villaseñor and Miguel González-Mendoza

A Defeasible Logic of Intention 321
José Martín Castro-Manzano

Towards Computational Political Philosophy 334
José Martín Castro-Manzano

Exploring the Solution of Course Timetabling Problems through Heuristic Segmentation 347
Dulce J. Magaña-Lozano, Santiago E. Conant-Pablos, and Hugo Terashima-Marín

Verifying Nested Workflows with Extra Constraints 359
Roman Barták and Vladimír Rovenský

Shortest Stochastic Path with Risk Sensitive Evaluation 371
Renato Minami and Valdinei Freire da Silva

Forward and Backward Feature Selection in Gradient-Based MDP Algorithms 383
Karina Olga Maizman Bogdan and Valdinei Freire da Silva

Medical Applications of Artificial Intelligence

Invited Paper

AI Based Fall Management Services – The Role of the *i-Walker* in I-DONTFALL 395
Ulises Cortés, Antonio Martínez-Velasco, Cristian Barrué, and Roberta Annicchiarico

Performance Evaluation of Ranking Methods for Relevant Gene Selection in Cancer Microarray Datasets 407
Manju Sardana, Baljeet Kaur, and R.K. Agrawal

Assessment of Bayesian Network Classifiers as Tools for Discriminating Breast Cancer Pre-diagnosis Based on Three Diagnostic Methods 419
Ameca-Alducin Maria Yaneli, Cruz-Ramírez Nicandro, Mezura-Montes Efrén, Martín-Del-Campo-Mena Enrique, Pérez-Castro Nancy, and Acosta-Mesa Héctor Gabriel

Identification of Risk Factors for TRALI Using a Hybrid Algorithm 432
María Dolores Torres, Aurora Torres, Felipe Cuellar, María de la Luz Torres, Eunice Ponce de León, and Francisco Pinales

Generation and Exploitation of Semantic Information Using an Epidemiological Relational Database as a Primary Source of Information 446
David González-Marrón, Miguel González-Mendoza, and Neil Hernández-Gress

A Graph Cellular Automata Model to Study the Spreading of an Infectious Disease 458
Maria Jose Fresnadillo Martínez, Enrique García Merino, Enrique García Sánchez, Jose Elias García Sánchez, Angel Martín del Rey, and Gerardo Rodríguez Sánchez

Modelling Interpersonal Relations in Surgical Teams with Fuzzy Logic 469
Andrzej Romanowski, Pawel Wozniak, Tomasz Jaworski, Pawel Fiderek, and Jacek Kucharski

Author Index 481

Table of Contents – Part II

Natural Language Processing

Invited paper

Syntactic Dependency-Based N-grams as Classification Features 1
Grigori Sidorov, Francisco Velasquez, Efstathios Stamatatos, Alexander Gelbukh, and Liliana Chanona-Hernández

Semantic Annotation for Textual Entailment Recognition 12
Assaf Toledo, Sophia Katrenko, Stavroula Alexandropoulou, Heidi Klockmann, Asher Stern, Ido Dagan, and Yoad Winter

Recognizing Textual Entailment in Non-english Text via Automatic Translation into English 26
Partha Pakray, Snehasis Neogi, Sivaji Bandyopadhyay, and Alexander Gelbukh

SMSFR: SMS-Based FAQ Retrieval System 36
Partha Pakray, Santanu Pal, Soujanya Poria, Sivaji Bandyopadhyay, and Alexander Gelbukh

Extrinsic Evaluation on Automatic Summarization Tasks: Testing Affixality Measurements for Statistical Word Stemming 46
Carlos-Francisco Méndez-Cruz, Juan-Manuel Torres-Moreno, Alfonso Medina-Urrea, and Gerardo Sierra

Extracting Domain-Specific Opinion Words for Sentiment Analysis 58
Ivan Shamshurin

Measuring Feature Distributions in Sentiment Classification 69
Diego Uribe

Identification of the Minimal Set of Attributes That Maximizes the Information towards the Author of a Political Discourse: The Case of the Candidates in the Mexican Presidential Elections 81
Antonio Neme, Sergio Hernández, and Vicente Carrión

Document Categorization Based on Minimum Loss of Reconstruction Information 91
Juan Carlos Gomez and Marie-Francine Moens

Semantic Classification of Posts in Social Networks by Means of Concept Hierarchies ... 104
Karina Ruiz-Mireles, Ivan Lopez-Arevalo, and Victor Sosa-Sosa

Evolutionary and Nature-Inspired Metaheuristic Algorithms

A Simple Adaptive Algorithm for Numerical Optimization ... 115
Francisco Viveros-Jiménez, Jose A. León-Borges, and Nareli Cruz-Cortés

A Post-optimization Strategy for Combinatorial Testing: Test Suite Reduction through the Identification of Wild Cards and Merge of Rows ... 127
Loreto Gonzalez-Hernandez, Jose Torres-Jimenez, Nelson Rangel-Valdez, and Josue Bracho-Rios

A New Branch and Bound Algorithm for the Cyclic Bandwidth Problem ... 139
Hillel Romero-Monsivais, Eduardo Rodriguez-Tello, and Gabriel Ramírez

The Evolution of Cooperation in File Sharing P2P Systems: First Steps ... 151
María Esther Sosa-Rodríguez and Elizabeth Pérez-Cortés

Optimal Joint Selection for Skeletal Data from RGB-D Devices Using a Genetic Algorithm ... 163
Pau Climent-Pérez, Alexandros Andre Chaaraoui, Jose Ramón Padilla-López, and Francisco Flórez-Revuelta

Dynamic Estimation of Phoneme Confusion Patterns with a Genetic Algorithm to Improve the Performance of Metamodels for Recognition of Disordered Speech ... 175
Santiago Omar Caballero-Morales and Felipe Trujillo-Romero

Modelling, Aggregation and Simulation of a Dynamic Biological System through Fuzzy Cognitive Maps ... 188
Gonzalo Nápoles, Isel Grau, Maikel León, and Ricardo Grau

Neural Networks and Hybrid Intelligent Systems

Artificial Neural Network for Optimization of a Synthesis Process of γ-Bi_2MoO_6 Using Surface Response Methodology ... 200
Guillermo González-Campos, Edith Luévano-Hipólito, Luis Martin Torres-Treviño, and Azael Martinez-De La Cruz

Recurrent Neural Control of a Continuous Bioprocess Using First and Second Order Learning 211
Carlos-Román Mariaca-Gaspar, Julio-César Tovar Rodríguez, and Floriberto Ortiz-Rodríguez

A Conic Higher Order Neuron Based on Geometric Algebra and Its Implementation 223
Juan Pablo Serrano Rubio, Arturo Hernández Aguirre, and Rafael Herrera Guzmán

Vehicle Lateral Dynamics Fault Diagnosis Using an Autoassociative Neural Network and a Fuzzy System 236
Juan Pablo Nieto González and Pedro Pérez Villanueva

Modular Neural Networks Optimization with Hierarchical Genetic Algorithms with Fuzzy Response Integration for Pattern Recognition ... 247
Daniela Sánchez, Patricia Melin, Oscar Castillo, and Fevrier Valdez

Neural Network with Type-2 Fuzzy Weights Adjustment for Pattern Recognition of the Human Iris Biometrics 259
Fernando Gaxiola, Patricia Melin, Fevrier Valdez, and Oscar Castillo

Model Reference Adaptive Position Controller with Smith Predictor for a Shaking-Table in Two Axes 271
Carlos Esparza, Rafael Núñez, and Fabio González

Implementation of a Single Chaotic Neuron Using an Embedded System 283
Luis González-Estrada, Gustavo González-Sanmiguel, Luis Martin Torres-Treviño, and Angel Rodríguez

Application of a Method Based on Computational Intelligence for the Optimization of Resources Determined from Multivariate Phenomena 292
Angel Kuri-Morales

Recurrent Neural Identification and I-Term Sliding Mode Control of a Vehicle System Using Levenberg-Marquardt Learning 304
Ieroham Baruch, Sergio-Miguel Hernandez-Manzano, and Jacob Moreno-Cruz

Multiple Fault Diagnosis in Electrical Power Systems with Dynamic Load Changes Using Soft Computing 317
Juan Pablo Nieto González

A Hybrid Algorithm for Crustal Velocity Modeling 329
José Federico Ramírez Cruz, Olac Fuentes, Rodrigo Romero, and Aaron Velasco

Relationship between Petri Nets and Cellular Automata for the Analysis of Flexible Manufacturing Systems 338
Irving Barragán, Juan Carlos Seck-Tuoh, and Joselito Medina

An Integrated Strategy for Analyzing Flow Conductivity of Fractures in a Naturally Fractured Reservoir Using a Complex Network Metric ... 350
Elizabeth Santiago, Manuel Romero-Salcedo, Jorge X. Velasco-Hernández, Luis G. Velasquillo, and J. Alejandro Hernández

Fuzzy Systems and Probabilistic Models in Decision Making

Comparative Study of Type-1 and Type-2 Fuzzy Systems for the Three-Tank Water Control Problem 362
Leticia Cervantes, Oscar Castillo, Patricia Melin, and Fevrier Valdez

Parallel Particle Swarm Optimization with Parameters Adaptation Using Fuzzy Logic 374
Fevrier Valdez, Patricia Melin, and Oscar Castillo

Indirect Adaptive Control with Fuzzy Neural Networks via Kernel Smoothing 386
Israel Cruz Vega, Luis Moreno-Ahedo, and Wen Yu Liu

Search and Detection of Failed Components in Repairable Complex Systems under Imperfect Inspections 399
Boris Kriheli and Eugene Levner

Distance Aproximator Using IEEE 802.11 Received Signal Strength and Fuzzy Logic 411
Carlos Fco Álvarez Salgado, Luis E. Palafox Maestre, Leocundo Aguilar Noriega, and Juan R. Castro

Towards Automated Extraction of Expert System Rules from Sales Data for the Semiconductor Market 421
Jesús Emeterio Navarro-Barrientos, Dieter Armbruster, Hongmin Li, Morgan Dempsey, and Karl G. Kempf

Bayesian Networks for Micromanagement Decision Imitation in the RTS Game Starcraft 433
Ricardo Parra and Leonardo Garrido

Designing and Implementing Affective and Intelligent Tutoring Systems in a Learning Social Network 444
Ramón Zatarain-Cabada, María Lucía Barrón-Estrada, Yasmín Hernández Pérez, and Carlos Alberto Reyes-García

Towards a Personality Fuzzy Model Based on Big Five Patterns for Engineers Using an ANFIS Learning Approach 456
Luis G. Martínez, Juan R. Castro, Guillermo Licea, Antonio Rodríguez-Díaz, and Reynaldo Salas

Author Index 467

Empirical Study of Machine Learning Based Approach for Opinion Mining in Tweets

Grigori Sidorov[1], Sabino Miranda-Jiménez[1], Francisco Viveros-Jiménez[1],
Alexander Gelbukh[1], Noé Castro-Sánchez[1], Francisco Velásquez[1],
Ismael Díaz-Rangel[1], Sergio Suárez-Guerra[1],
Alejandro Treviño[2], and Juan Gordon[2]

[1] Center for Computing Research,
Instituto Politécnico Nacional,
Av. Juan de Dios Bátiz, s/n, esq. Mendizabal,
Col. Nueva Industrial Vallejo, 07738, Mexico City, Mexico
[2] Intellego SC, Mexico City, Mexico
sabino_m@hotmail.com
www.cic.ipn.mx/~sidorov

Abstract. Opinion mining deals with determining of the sentiment orientation—positive, negative, or neutral—of a (short) text. Recently, it has attracted great interest both in academia and in industry due to its useful potential applications. One of the most promising applications is analysis of opinions in social networks. In this paper, we examine how classifiers work while doing opinion mining over Spanish Twitter data. We explore how different settings (n-gram size, corpus size, number of sentiment classes, balanced vs. unbalanced corpus, various domains) affect precision of the machine learning algorithms. We experimented with Naïve Bayes, Decision Tree, and Support Vector Machines. We describe also language specific preprocessing—in our case, for Spanish language—of tweets. The paper presents best settings of parameters for practical applications of opinion mining in Spanish Twitter. We also present a novel resource for analysis of emotions in texts: a dictionary marked with probabilities to express one of the six basic emotions—Probability Factor of Affective use (PFA)—Spanish Emotion Lexicon that contains 2,036 words.

Keywords: Opinion mining, sentiment analysis, sentiment classification, Spanish Twitter corpus, Spanish Emotion Lexicon.

1 Introduction

Opinion mining (or sentiment analysis[1]) has attracted great interest in recent years, both in academia and industry due to its potential applications. One of the most

[1] The terms "opinion mining" and "sentiment analysis" usually are used to denote essentially the same phenomenon, thus, they can be considered synonyms. It should be mentioned, though, that if we say "opinion", we can refer to much broader sense, appealing to substantial characteristics of our object, like, for example, size of a product, its weight, etc. While saying "sentiment", we mean only positive or negative feelings. If we would like to analyze more detailed feelings, we would say "emotion analysis/mining".

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 1–14, 2013.

promising applications is analysis of opinions in social networks. Lots of people write their opinions in forums, microblogging or review websites. This data is very useful for business companies, governments, and individuals, who want to track automatically attitudes and feelings in those sites. Namely, there is a lot of data available that contains much useful information, so it can be analyzed automatically. For instance, a customer who wants to buy a product usually searches the Web trying to find opinions of other customers or reviewers about this product. In fact, these kinds of reviews affect customer's decision.

Opinion mining in a broad sense is defined as the computational study of opinions, sentiments and emotions expressed in texts [1]. Opinions exist on the Web for any entity or object (person, product, service, etc.), and for the features or components of these objects, like, a cell phone battery, keyboard, touch screen display, etc.

Detecting sentiments is considered a difficult task. Say, in the example '*la aplicación responde muy rápido* (*the application responds very fast*)'; the sentiment of the opinion is positive, because the word '*rápido* (*fast*)' implies a good thing—it is good that applications run fast. However, the same word in other context, like in the sentence '*la batería se descargó muy rápido* (*the battery discharged very fast*)', implies a negative sentiment—it is bad that batteries reduce their power quickly. So, the problem implies using of world knowledge, which is very vast and complex problem.

Formally, we say that an **opinion** of a feature ***f*** has a **sentiment** attached, commonly positive or negative. The person who emits the opinion is known as **opinion holder**. Thus, an opinion is defined as a quintuple (o_j, f_{jk}, oo_{ijkl},h_i, t_l) [2], where:

- o_j is the object of the opinion.
- f_{jk} is a feature of the object o_j about which the opinion is expressed. When no feature is detected, we use "**general opinion**" as the object feature.
- oo_{ijkl} is the sentiment polarity of the opinion about the feature f_{jk} of the object o_j—positive, negative, neutral.
- h_i is the opinion holder.
- t_l is the time when the opinion is expressed by h_i.

For our work we use messages posted in Spanish Twitter. In this work, the opinion quintuple matches a message as follows:

- o_j is the entity the tweet deals with. A tweet contains one or more entities. Entities are sets of synonyms defined by a user.
- f_{jk}, feature is ignored for the moment, i.e., general opinion is used as the object feature.
- oo_{ijkl} is the message global polarity: positive, negative, neutral, or informative (news).
- h_i is the user who posted the message.
- t_l is the time when the message was posted.

In the following message '*@user: Mi iPhone se calienta mucho (@user: My iPhone gets overheated*)'. The object (o) is *iPhone*; the feature (f) is related to the temperature, but in this work we will not try to detect it; the assigned polarity (oo) is negative, because it is bad that a cellphone gets overheated; the holder (h) is @user; and time (t) is the Twitter publication time. We use this formalism because it suits our domain (Twitter, see section 2.1). In case of Twitter, we use short text in contrast to reviews that are longer texts [3].

Many systems and approaches have been implemented for detecting sentiments in texts [1, 4]. We can distinguish two main methodologies used in opinion mining: machine learning approaches and the so-called symbolic approaches—approaches that use manually crafted rules and lexicons [5, 6]. This paper focuses on machine learning approaches.

Opinion mining task can be transformed into classification task, so machine learning techniques can be used for opinion mining. Machine learning approaches require a corpus containing a wide number of manually tagged examples, in our case, tweets with a sentiment assigned manually by a human annotator. In our corpus, text is represented as a set of features for classification. These features are traditional word n-grams extracted from each tweet in the corpus.

Let us explain briefly the n-gram representation for the sentence '*battery discharges very fast*'. When using n-gram features, an opinion is represented as independent n-grams of various orders: unigram (*battery, discharge, very, fast*), bigrams (combination of two words: *battery-discharge, discharge-very, very-fast*), trigrams (combination of three words: *battery-discharge-very, discharge-very-fast*), and so on. Note that we use morphologically normalized representation. When using POS n-grams, a POS-tag is used instead of each text word. For example, when using POS unigrams in the features set *battery* is changed to *Noun, discharge* is changed to *Verb, very* is changed to *Adverb, fast* is changed to *Adjective*, etc.

For the classification task, the most frequently used machine learning methods are: Naïve Bayes (NB), Maximum Entropy (MaxEnt), and Support Vector Machines (SVM). Machine learning approaches report relatively high results, i.e., Pang *et al.* [4] obtained 81.0% (NB), 80.4% (MaxEnt), and 82.9% (SVM). They used two classes: positive and negative, and worked using product reviews that are longer texts than tweets. In the domain of English Twitter, Go *et al.* [7] report similar results 81.3%, 80.5%, and 82.2% for the same classifiers. They use three classes: positive, negative, and neutral.

In this research, our aim is to find out what are the best settings of parameters for classifying of Spanish tweets. Our research questions are:

(1) Proper size for the training corpus,
(2) The best classifier for the task,
(3) Optimal size of n-grams,
(4) How using more classes—positive, negative, neutral, and informative (news)—affect~~s~~ precision,
(5) If balanced corpus improves precision (vs. unbalanced),
(6) How classifiers perform when the testing set and the training corpus belong to different domains, and
(7) The cases where the best classifier fails, i.e., the reason behind the errors.

There is little research on opinion mining in Spanish [8, 9], so we also describe specific preprocessing techniques for this language. This is the first research that uses Spanish Twitter corpus.

The paper is organized as follows. In section 2, we describe the used corpus, data preprocessing, and selected classifiers. In section 3, we present the results of the analysis for different settings of parameters. In section 4, the common errors are analyzed. In section 5, we describe novel resource for analysis of emotions on texts: a dictionary marked with probabilities to express one of the six basic emotions—Spanish Emotion Lexicon that contains 2,043 words. In section 6, conclusions are presented.

2 Opinion Mining Method

The general scheme of our processing is composed of several stages. First, a Spanish Twitter corpus is compiled (see section 2.1). After this, the data is modified in order to prepare the necessary information for classifiers (see section 2.2). Finally, the classifiers are trained with different settings as shown in section 2.3.

2.1 Corpus of Tweets

We chose to work with Spanish Twitter for our experiments. Twitter is a microblogging platform where users post their messages, opinions, comments, etc. Contents of the messages range from personal thoughts to public statements. A Twitter message is known as tweet. Tweets are very short; the maximum size of a tweet is 140 characters that usually correspond to a phrase. Thus, our work is limited to sentence level.

We use a global polarity rating due to shortness of messages in Twitter, and we do not process cases where tweets have more than one sentiment orientation. In addition, we do not extract features of products as in case of the reviews, neither we use predefined sets of characteristics [4]. In case of free-form texts (our case), it is not easy to determine object features, as, for example, in case of movie reviews, where the predefined sets of characteristics of films are used.

We compiled a corpus based on data extracted form Twitter. The corpus was built using a list of predefined entities about cell phone brands. We collected 32,000 tweets, and around 8,000 tweets were annotated by hand determining one of four classes for each tweet: positive (P), negative (N), neutral (T), or informative (news, I). Each class is described as follows:

1. Positive, if it has a positive sentiment in general, like in '*la aplicación responde muy rápido* (*the application responds very fast*)'.
2. Negative, if it has a negative sentiment in general, like in '*mi iPhone se calienta mucho* (*my iPhone gets overheated*)'.
3. Neutral, if it has no sentiment '*Estoy tuiteando desde el iPhone* (*I am tweeting from my iPhone*)'.

4. Informative (news), if it contains news or advertisements '*Vendo mi Samsung Galaxy, nuevo en caja* (*I sell my Samsung Galaxy, new in the box*)'.

Note that it is common to use just two classes—positive and negative. However, we used these four categories because one of our aims is to find out how the number of classes affects precision of classifiers.

2.2 Preprocessing

Analysis of tweets is complex task because these messages are full of slang, misspellings [7] and words borrowed from other languages. Some examples of errors are shown in Table 1. In order to tackle the problems mentioned in Table 1 and to deal with the noise in texts, we normalize the tweets before training the classifiers with the following procedures:

- Error correction,
- Special tags,
- POS-tagging,
- Negation processing.

Table 1. Common errors in Spanish tweets

Type of error	Example
(1) Slang	**(x fa/please)** *olvidé un celular en un Matiz, x fa que lo devuelvan* (*I forgot a cell phone in a Matiz, please give it back*)
(2) Misspelling	**(muerto~~oo~~/dead)** *tu celular estaba muertooo!* (*your cellphone was dead*!)
(3) Mixed languages	**(bonito/nice)** *ya está aquí, más* nice, *más rápido, el Nokia Lumia* (*It's here, It's very nice, It's faster, the Nokia Lumia*)

Error Correction

In case of orthographic errors like in (1) '*muertooo* (*dead*)', we use an approach based on a Spanish dictionary and a statistical model for common double letters in Spanish. Also, we developed a set of rules made by hand for slang and common words borrowed from the English language. The rules were made after manual analysis of the data from our corpus. We do not detect orthographic mistakes ('*dies*' instead of '*diez* (*ten*)'); '*sincronizacion*' instead of '*sincronización* (*synchronization*)') or split the words that are agglutinated ('*padrepero*' instead of '*padre pero*' (*nice but*)).

Special Tags

For usernames, hash tags, emoticons, and URLs in a tweet, we use an approach similar to [7]. We use special tags (USER_TAG, WINK_TAG, HASH_TAG, and

URL_TAG) to replace the word by the corresponding tag, so that POS-tagger could tag correctly each word of the tweet. For instance, in the tweet '*@user no me arrepiento, soy feliz con iPhone :)* (*I have no regrets, I am happy with iPhone* :))', the user is identified by @ and the wink by the symbols :). List of common winks was compiled manually. The normalized tweet would be '*USER_TAG no me arrepiento, soy feliz con iPhone WINK_TAG*'.

POS-Tagging

After text normalization, we applied a POS-tagger for Spanish using Freeling tool [10]. Freeling is a system for linguistic analysis of texts, like tagging, lemmatization, etc. After applying the POS-tagger, we obtain for each word its corresponding part of speech: verb, adjective, adverb, etc. Freeling follows the EAGLES recommendations for morphosyntactic tag set [13]. Also, we use the lemmatized words in order to reduce the number of word forms, which is important for morphologically rich Spanish language. For example, the tweet mentioned above is tagged as '*USER_TAG_NT000* (noun) *no_RN* (adverb) *me_PP1CS000* (pronoun) *arrepentir_VMIP1S0* (verb) *,_Fc* (punctuation) *ser_VSIP1S0* (verb) *feliz_AQ0CS0* (adjective) *con_SPS00* (preposition) *iPhone_NCMS000* (noun) *WINK_TAG_NT000* (noun)'.

Processing of Negation

Negation affects the value of an opinion. We use a similar approach as in [11] to handle negations. We search and remove the adverb 'no' from opinion, and attach the prefix 'no_' to next word (verb or adjective) to build one unit. For example, '*no_RN* (adverb) *tener_VMIP1S0* (verb) *uno_DI0MS0* (article) *iPhone_NCMS000* (noun) (*no tengo un iPhone* / I do not have an iPhone)' is transformed into '*no_tener_VMIP1S0 uno_DI0MS0 iPhone_NCMS000*'. Rules of transformation were made by hand according to the patterns detected in our corpus.

2.3 Selected Classifiers

Our method uses various machine learning classifiers. The machine learning classifiers we selected were: Naïve Bayes (NB), C4.5 (Decision Tree) and Support Vector Machines (SVM). NB and SVM were used in several experiments with good results for English language [4, 7].

We use WEKA API that implements all above mentioned algorithms [12]. WEKA implements SVM as SMO, and C4.5 as J48 algorithms.

Our input data are two sets of vectors. Each entry in the vector corresponds to a feature. We use the part of speech tags as filters for features. The part of speech tags that we consider as features are verbs, nouns, adjectives, adverbs, and interjections.

A set of 8,000 tweets were manually marked with one of the four categories mentioned above. We use 7,000 tweets as training set and 1,000 tweets as test set. The test set has 236 positive tweets, 145 negative, 342 neutral, and 257 news or advertisements.

3 Experiments and Evaluation

In this section, we describe the experiments that we carried out to determine the influence of corpus size, n-gram size, number of the classes, and balanced vs. unbalanced corpus on machine learning based sentiment classification. The models were trained using different sizes of n-grams. Let us remind that we consider the word and its POS tag together as features, for example, '*trabajar_verbo* (*work_verb*)' is one feature. We compute precision of the classifier on the whole evaluation dataset using equation 1.

$$precision = \frac{correct\ answers}{total\ answers} \tag{1}$$

Our further analysis is based on the following experiments:

- Effect of the corpus size,
- Effect of the n-gram size,
- Effect of the number of the classes,
- Effect of an unbalanced corpus.

We conduct the experiments using the best setting obtained in the previous tests. All precision values of the following tables are given as percentage.

3.1 Effect of the Corpus Size

We tested different corpus sizes for training the three classifiers. In Table 2, we show how the corpus size affects precision. The precision was improved when using more training samples.

Table 2. Precision observed when using 12 different training corpus sizes

Classifier	**Part I. Corpus size (tweets)**						
	1,000	**1,500**	**2,000**	**2,500**	**3,000**	**3,500**	**4,000**
Naïve Bayes	42	46	42	42	45	44	45
J48	43	49	46	49	50	52	54
SVM	48	55	55	59	**61**	59	60

Classifier	**Part II. Corpus size (tweets)**					
	4,500	**5,000**	**5,500**	**6,000**	**6,500**	**7,000**
Naïve Bayes	42	43	44	43	44	46
J48	53	55	52	54	54	57
SVM	59	60	59	60	**61**	**61**

However, it can be observed that after 3,000 tweets precision is improving very slowly. Also, it can be noted that precision in the interval 3,500-6,000 slightly fluctuates. Thus, test results suggest that 3,000 samples are enough as a training set for a selected topic (cell phones in our case).

3.2 Effect of n-Gram Size

We perform tests to study the effect of the n-gram order (size) on the precision of classifiers. We tried six different sizes. Results are shown in Table 3. It confirms that unigram is the best feature size. This conclusion confirms the conclusions obtained in other studies for English language and different corpus domain such as Twitter and films reviews [4, 9].

Table 3. Precision observed when using 6 different n-gram sizes

Classifier	**N-gram size**					
	1	**2**	**3**	**4**	**5**	**6**
Naïve Bayes	46	37	35	35	35	35
J48	57	41	35	35	35	35
SVM	**61**	49	41	35	35	35

3.3 Effect of the Number of Classes

Table 4 describes the values of the number of classes and their composition. For example, in class number 2 there are two types of categories: *positive* and *negative*, but positive value also corresponds to *positive*, *neutral*, and *news* opinions. The class number represents the quantity of classes in the group.

Table 4. Possible combinations for classifying classes

Number of the classes	***Values in the classes***	***Clustering of the values***
2	positive, negative	I. positive: positive, neutral, or news II. negative
3	positive, negative, neutral	I. positive II. negative III. neutral: neutral or news
4	positive, negative, neutral, news	I. positive II. negative III. neutral IV. news

Table 5. Precision observed when using different number of target classes

Classifier	*Number of the classes*		
	2	**3**	**4**
Naïve Bayes	78.2	58.3	46.0
J48	83.6	60.2	57.0
SVM	**85.8**	69.0	61.0

Table 5 shows the effect of the number of classes on the classifier performance. We see that reducing the number of classes increases the classifiers precision. It is not surprising because we decrease the possibility of errors.

3.4 Effect of Balanced vs. Unbalanced Corpus

In this section, our goal was to analyze the effect of balanced vs. unbalanced corpus on classification. We selected 4,000 tweets from the annotated corpus in order to build a balanced subcorpus. Namely, 1,000 tweets were selected for each class, i.e., each class has equal representation. We classified according to the setup of Table 5. Tables 6 and 7 show the results obtained when using an unbalanced and a balanced corpus respectively. We can observe that the best precision was 85.8% for positive and negative classes with the unbalanced corpus. This result is slightly higher than the 82.2% reported for English Twitter in [7]. It is interesting to observe that the precision decreased when using a balanced corpus, though not very much. Perhaps, this behavior was due to the average number of adjectives (1.15) and adverbs (0.58) per tweet in the unbalanced corpus, which is higher than in the balanced corpus (adjectives: 0.98 and adverbs: 0.63). Adjectives and adverbs usually have more sentiment connotations. This phenomenon is part of our future research.

Another interesting point is that the Decision Tree classifier (J48) in general is more stable as far as the effects of balancing of the corpus are concerned.

Table 6. Precision observed when using an unbalanced corpus

Classifier	*Number of classes*		
	2	**3**	**4**
Naïve Bayes	78.2	58.3	46.0
J48	83.6	60.2	57.0
SVM	**85.8**	69.0	61.0

Table 7. Precision observed when using a balanced corpus

Classifier	*Number of classes*		
	2	**3**	**4**
Naïve Bayes	72.0	48.0	31.3
J48	80.9	62.6	46.6
SVM	**81.6**	62.2	54.6

3.5 Effect of Testing on Different Domains

For evaluation of the influence of the domain, the classifiers were trained using the domain of cell phones and were tested both in domain of cell phones and political domain. The data for the political domain were taken from other corpus of tweets, i.e., these are two very different domains. The political domain test set contains 1,400 tweets (positive: 255, negative: 725, neutral: 134, and news: 286). Table 8 shows that training with a corpus that has a domain different from the target domain affects precision very negatively, namely, it is two or three times worse.

Table 8. Precision observed on different domains

Classifier	*Same domain*	*Different domain*
Naïve Bayes	78.2	34.0
J48	83.6	17.0
SVM	**85.8**	28.0

3.6 Best Settings for Practical Applications

We conclude that the best settings for our practical application were:

- Using unigrams (i.e., n-gram size is equal to one),
- Having a training set containing at least 3,000 tweets,
- Using SVM classifier,
- Having two polarity classes (positive and negative) if possible,
- Having training and target sets within the same domain.

When using all the aforementioned settings together we observed a precision of 85.8%.

4 Analysis of Errors

We found some common types of errors when analyzing the misclassified samples. The most frequent errors were:

- Shortened messages,
- Misspelling,
- Humor, irony and sarcasm,
- Human tagging errors.

4.1 Shortened Messages

In the context of Twitter, it is common to see shortened messages like: ‘*mi celular!!! La pantalla* (*my cellphone !!! the display*)’ that correspond to elliptical phrases. Messages like these do not have any sentiment interpretation for other persons; they are understandable basically by the opinion holder himself. The human annotator assigns here usually a negative opinion because he supposes that the display does not work anymore. However, this supposition is based on beliefs of the annotator, and not on the real situation. The SVM classifier assigned here the neutral value, because there is insufficient information for other type of decision.

4.2 Misspelling

Orthographic errors are common in tweets such as ‘*No me gushta le iPhone*’ ‘*Ya tng iPhone de nuevo* (*I don’t like the iPhone; I have an iPhone again*)’. Misspelled words and shortness of the messages make difficult for a classifier to determine the right class. The human annotator marked the first one as negative, and the second one as positive, but the SVM classifier assigned neutral class in both cases.

4.3 Humor, Irony and Sarcasm

Treatment of humor and its subtypes like irony or sarcasm is an interesting but extremely difficult problem. It is very complex because while interpreting humor we often rely on the world knowledge and the (very broad) context, as well as on much more difficult to represent subtle cultural patterns.

Also we should take into account that humor usually is not expressed directly. For example, let us consider the tweet ‘*Mi novio es como mi iPhone. No tengo.* (*My boyfriend is like my iPhone. I don’t have one*)’. It was automatically classified as positive, but the human annotator marked it as neutral. In fact, there is no enough information to guess correctly. These phenomena are difficult to determine without reviewing the context [1].

4.4 Human Tagging Errors

Sometimes, human annotator cannot make decision because of the complexity of the context of a tweet. For example, ‘*Hablar vale más que un iPhone...Yo tengo tu amor*

(*Talking is worthier than having an iPhone, I have your love*)'. The human annotator marked it as negative, while the classifier marked it as neutral. If we analyze deeper the context, then two facts hold. While it is true that in the cell phone context, not having an iPhone is a negative sentiment, but in the human relationships context, usually it is positive to have a partner.

5 Spanish Emotion Lexicon

For automatic analysis of emotions expressed in tweets, specialized lexical resources are necessary. One of these resources is Spanish Emotion Lexicon. It is developed by I. Díaz-Rangel, G. Sidorov, and S. Suárez-Guerra. They submitted a journal paper where they explain detailed methodology of the creation of the dictionary [15]. Here we present only the general idea of the Lexicon and announce its availability for academic usage.

Spanish Emotion Lexicon contains 2,036 words that are associated with the measure of Probability Factor of Affective use (PFA) with respect to at least one basic emotion: joy, anger, fear, sadness, surprise, and disgust.

We selected the words from English SentiWordNet [14] and translated them automatically into Spanish. Then we manually checked 3,591 obtained words using Maria Moliner dictionary and leave only words that had at least one meaning related with the basic emotions.

Then we asked 19 annotators to evaluate how probable is the association of the word with one of the emotions, i.e., how easily a context (with the word) related with the emotion can be imagined. No semantic analysis was performed. We selected the scale: null, low, medium, high.

After this we used the weighted Cohen's kappa [16] for calculation of agreement between annotators (pairwise) and leave only 10 annotators with the best agreement scores. In this manner, we try to improve the objectivity of the values and eliminate "bad" annotators. The values of kappa were improved about 15%.

At the next step we represent the number of evaluations as percentages, as can be seen in Table 9. For example, for the word *abundancia (abundance)*, 50% of annotators chose "medium" and 50% chose "high".

Table 9. Example of average evaluation for "joy"

Word	Null[%]	Low[%]	Medium[%]	High[%]
abundancia (abundance)	0	0	50	50
aceptable (acceptable)	0	20	80	0
acallar (to silence)	50	40	10	0

Now we are ready to calculate a new measure for each word that we called Probability Factor of Affective use (PFA). It is based on the percentages of Table 9. Note that PFA is 1 if 100% of annotators relate it to the "high" value of the association with the emotion, and it is 0 if 100% of annotators relate it to the "null" value. So, intuitively it has very clear meaning: the higher the value of the PFA is, the more probable the association of the word with the emotion is. We present the exact formula in our submitted paper [15]. For example, for the words in Table 9, *abundancia (abundance)* has PFA=0.83, *aceptable (acceptable)* has PFA=0.594, *acallar (to silence)* has PFA=0.198.

Spanish Emotion Lexicon is available from www.cic.ipn.mx/~sidorov.

6 Conclusions

The large amount of information contained in Twitter makes it an attractive source of data for opinion mining and sentiment analysis. Performance of machine learning techniques is relatively good when classifying sentiments in tweets, both in English and in Spanish. We believe that the precision can be further improved using more sophisticated features.

In this research, we presented an analysis of various parameter settings for selected classifiers: Supported Vector Machines, Naïve Bayes and Decision Trees. We used n-grams of normalized words (additionally filtered using their POS-tags) as features and observed the results of various combinations of positive, negative, neutral, and informative sets of classes. We made our experiments in Spanish language for the topic related to cell phones, and also partially used data from tweets related to the recent Mexican presidential elections (for checking the balanced vs. unbalanced corpus).

From the analysis of the results, we found that the best configuration of parameters was: (1) using unigrams as features, (2) using less possible number of classes: positive and negative, (3) using at least 3,000 tweets as training set (incrementing this value does not improve precision significantly), (4) balancing the corpus as regards the proportional representation of all classes gives slightly worse results, and (5) Supported Vector Machines was the classifier with the best precision.

We also present in this paper Spanish Emotion Lexicon that is useful available resource for analysis of emotions in tweets and in any texts, if we do not perform detailed word sense analysis. The resource contains 2,036 words marked for six basic emotions with Probability Factor of Affective use (PFA).

In future work, we plan to explore richer linguistic analysis, for example, parsing, semantic analysis and topic modeling. Also, better preprocessing is needed in order to avoid errors mentioned above.

Acknowledgments. This work was done under partial support of the Mexican Government (SNI, COFAA-IPN, PIFI-IPN, SIP-IPN 20120418, 20121823, 20113295 and 20111146, CONACYT 50206-H and 83270), CONACYT-DST India ("Answer Validation through Textual Entailment"), Mexico City Government (ICYT PICCO10-120), and FP7-PEOPLE-2010-IRSES: Web Information Quality - Evaluation Initiative (WIQ-EI) European Commission project 269180.

References

1. Pang, B., Lee, L.: Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval 2, 1–135 (2008)
2. Liu, B.: Sentiment Analysis and Subjectivity. In: Indurkhya, N., Damerau, F.J. (eds.) Handbook of Natural Language Processing, 2nd edn. (2010)
3. Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the ACL 2004 (2004)
4. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL, pp. 79–86. Association for Computational Linguistics (2002)
5. Polanya, L., Zaenen, A.: Contextual Valence Shifters. Computing Attitude and Affect in Text: Theory and Applications. In: Computing Attitude and Affect in Text: Theory and Applications, vol. 20 (2006)
6. Wilson, T., Hoffmann, P., Somasundaran, S., Kessler, J., Wiebe, J., Choi, Y., Cardie, C., Riloff, E., Patwardhan, S.: Opinion Finder: a system for subjectivity analysis. In: Proceedings of HLT/EMNLP on Interactive Demonstrations, Vancouver, British Columbia, Canada, pp. 34–35 (2005)
7. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford University, Stanford, CA (2009)
8. Martínez Cámara, E., Martín Valdivia, M.T., Perea Ortega, J.M., Ureña López, L.A.: Técnicas de Clasificación de Opiniones Aplicadas a un Corpus en Español. Procesamiento del Lenguaje Natural, Revista 47, 163–170 (2011)
9. Aiala, R., Wonsever, D., Jean-Luc, M.: Opinion Identification in Spanish Texts. In: Proceedings of the NAACL HLT (2010)
10. Padró, L., Collado, M., Reese, S., Lloberes, M., Castellón, I.: FreeLing 2.1: Five Years of Open-Source Language Processing Tools. In: Proceedings of 7th Language Resources and Evaluation Conference, La Valletta, Malta (2010)
11. Das, S., Chen, M.: Yahoo! For Amazon: Extracting market sentiment from stock message boards. In: Proceedings of the 8th Asia Pacific Finance Association Annual Conference (2001)
12. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations 11(1) (2009)
13. EAGLES. Recommendations for the morphosyntactic annotation of corpora, Eag-tcwg-mac/r, ILC-CNR, Pisa (1996)
14. Esuli, A., Sebastiani, F.: SentiWN: A Publicly Available Lexical Resource for Opinion Mining. In: Fifth International Conference on Language Resources and Evaluation (LREC 2006), pp. 417–422 (2006)
15. Díaz-Rangel, I., Sidorov, G., Suárez-Guerra, S.: Weighted Spanish Emotion Lexicon (submitted 2012)
16. Cohen, J.: A coefficient of agreement for nominal scales. Educational and Psychological Measurement 20, 37–46 (1960)

An Empirical Evaluation of Different Initializations on the Number of K-Means Iterations

Renato Cordeiro de Amorim

Department of Computer Science and Information Systems
Birkbeck University of London, Malet Street WC1E 7HX, UK
renato@dcs.bbk.ac.uk

Abstract. This paper presents an analysis of the number of iterations K-Means takes to converge under different initializations. We have experimented with seven initialization algorithms in a total of 37 real and synthetic datasets. We have found that hierarchical-based initializations tend to be most effective at reducing the number of iterations, especially a divisive algorithm using the Ward criterion when applied to real datasets.

Keywords: K-Means, K-Means++, iK-Means, Build, Hierarchical Clustering.

1 Introduction

The K-Means algorithm originates from the independent works of MacQueen [1] and Ball and Hall [2]. It is arguably the most popular partitional clustering algorithm. It aims to partition a dataset into K clusters by minimising the sum of distances between all entities in a cluster $S_k \in S$ and the respective centroid $c_k \in C$ for k=1, ..., K.

$$W(S,C) = \sum_{k=1}^{K} \sum_{i \in S_k} d(i, c_k) \tag{1}$$

where i is an entity in the dataset I, and $d(i, c_k)$ the distance between i and its cluster centroid c_k. This criterion is open to different distance measures, the squared Euclidian distance being the most frequently used. The K-Means algorithm is formally defined below:

1. Set the value of K initial centroids c_1, c_2, ..., c_K.
2. Set all entities to their closest centroid c_k, creating clusters S_1, S_2, ..., S_K.
3. If Step 2 produces no changes in the clustering, the clusters are final.
4. Update each centroid c_k to the mean of S_k.

K-Means has been in use for over 50 years; its longevity is due to the number of benefits it presents, such as efficiency and empirical success [3]. However,

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 15–26, 2013.

this algorithm is not faultless; its flaws include the lack of a universally agreed definition for the term 'cluster' [18], whose number the algorithm assumes to be known beforehand. This raises a complex question: how can one quantify something that is undefined? Plausible answers are subjectively related to the data being clustered.

K-Means is a greedy algorithm; as such, its outcome is highly dependent on its initial centroids, which are normally chosen at random. There have been a number of suggested solutions for this problem [4,9,12,14,15] and related comparisons [8,13,17,23].

In this paper, we take a different focus, aiming to answer the question: how do different initializations impact the number of iterations K-Means takes to converge?

It is intuitive to us that different initializations of K-Means will change the number of iterations the algorithm takes to converge. We are unaware of any publication that has attempted to quantify this change and we find the assertion that 'the required number of iterations tends to be less than the number of entities' [19] to be vague.

The number of K-Means iterations is intimately related to the convergence speed of K-Means, making its minimisation desirable. The number of iterations is much less dependent on hardware or programming ability than on the processing time of K-Means; thus, the experiments in this paper are easier to reproduce.

To quantify the change in K-Means iterations, we have experimented with a number of datasets, synthetic and otherwise, utilising seven different K-Means initializations. For each experiment, we present the number of iterations K-Means needed to converge and the amount of time the initialization alone took to complete. We also present the accuracy of each dataset as the number of correctly clustered entities, obtained using a confusion matrix, divided by the total number of entities.

In our experiments, we have utilized the standard implementation of K-Means in the statistics toolbox of Matlab. In this implementation, after the algorithm converged and we took the number of iterations, a further approximation was made by re-running K-Means from its current state, moving only one entity at a time. By following this approach, if a single entity moved its cluster in the final set of clusters, it would increase the output of Equation (1).

2 Initialization Algorithms

We have experimented with seven popular initializations for K-Means. Evidence of their success can be found in the appointed references.

Random. We ran K-Means 100 times. In each run, we randomly selected the values of K different entities in the dataset as the initial centroids. The main idea behind this initialization was that cluster areas have a high density; in consequence, entities in such areas have a higher chance of being picked than outliers.

This initialization on its own (i.e., without taking into account the next step: K-Means) takes a negligible amount of time to complete, hence the 'neg' in the tables.

K-Means++. This algorithm was introduced by Arthur and Vissilvitskii [15]. After picking the first centroid from Dataset I at random, the probability of an entity i being chosen as the following centroid is proportional to its distance from the nearest previously chosen centroid. Formally, the algorithm has five steps:

1. Choose one centroid, c_1, at random from Dataset I. Initialise $C = \{c_1\}$.
2. Set $D(x)$ as the shortest Euclidean distance from an entity $i \in I$ to the closest centroid.
3. Take a new centroid, c_n, with probability $D(x)^2 / \sum_{i \in I} D(x)^2$. Add c_n to C
4. If the cardinality of C is smaller than K, go back to Step 2.
5. Run K-Means, initialising it with all of the centroids in C.

The algorithm is not deterministic; hence we ran it 100 times per dataset, as for the Random initialization.

iK-Means. The intelligent K-Means algorithm (iK-Means) introduced by Mirkin [4], finds initial centroids for K-Means based on the concept of anomalous patterns (AP). The algorithm iteratively finds the centroids by picking the entity most distant from the dataset centre and applying K-Means with two initial centroids: the found entity and the data centre. After convergence, the cluster initiated with the found entity is removed from the dataset and the process restarts until all of the data is clustered. Clusters whose cardinality is below a user-specified threshold are discarded.

1. Pick the entity most distant from the data centre, c_d, as a tentative centroid c_t.
2. Apply K-Means, using c_d and c_t as initial centroids.
3. After convergence, remove the cluster S_t from the dataset; c_d remains unchanged.
4. If not all data is clustered, go to Step 1.
5. Remove any cluster whose cardinality is below a user-specified threshold.
6. Using the found centroids, apply K-Means.

In our experiments, we were not interested in finding the quantity of clusters in the datasets, which is a complex problem on its own; hence when applying iK-Means, we set its threshold to zero and chose the K largest clusters, where K is the given number of clusters. Where two clusters had the same cardinality, preference was given to the one that was found first, as this was the the most anomalous.

We have successfully taken a similar approach before, when incorporating constraints [21] and feature weighting in the Minkowski space to iK-Means [22].

Build. The Build algorithm [12] is normally utilised to initialise the Partition Around Medoids (PAM). The algorithm is as follows:

1. Select c_m, the entity with the smallest sum of distances from all other entities.
2. The next centroid, c_{m+1}, is defined as the most remote from c_m. An entity, i, for which E_i is maximum over $i \in I - c_m, E_i = \sum_{i \in S_i} d(j, c_m) - d(i, j) > 0$, j being a non-selected entity.

To some extent, it does resemble the iK-Means algorithm; the similarities and differences are discussed in [4].

Hartigan and Wong (H&W). In their paper [9], Hartigan and Wong suggest an algorithm to find initial centroids for K-Means. In this, one should first order all N entities of a dataset according to its centre of gravity. Then, for each cluster in $S = \{S_1, S_2, ..., S_K\}$, its centroid is the $1 + (k-1) * [N/K]th$ entity. Hartigan and Wong state that, using this method, no cluster will be empty after the first assignment of entities to centroids. We have formalized the algorithm as follows:

1. Order all N entities in the dataset in respect to the gravity centre.
2. For each cluster in S, set its centroid equal to $1 + (k-1) * [N/K]th$ entity.

Hierarchical Agglomerative Ward. Ward's hierarchical method [7] allows the creation of a dendrogram describing the whole dataset. Milligan [14] suggests using it to determine the initial centroids for K-Means. The method can be formalised as:

1. Set every entity as the centroid of its own cluster.
2. Merge the closest clusters S_{w1} and S_{w2}, following the Ward distance.
3. Remove the references to S_{w1} and S_{w2}, leaving only the new cluster, $S_{w1 \cup w2}$.
4. If the number of clusters is bigger than the desired number, go to Step 2.

Hierarchical Divisive Ward. Unlike the agglomerative approach, divisive algorithms begin with a cluster containing the whole dataset and iteratively split any cluster into two child clusters, until it reaches a stop condition. Mirkin [11,4] demonstrates that the Ward criteria can be utilised under this approach by introducing a Ward-like divisive clustering algorithm. The formalisation of the algorithm is as follows:

1. Put $S_w \leftarrow S$, where S is the cluster containing all of the entities in the dataset.
2. Split S_w into S_{w1} and S_{w2} in such a way that it maximises the Ward distance $wd(S_{w1}, S_{w2})$.
3. Set $S_w \leftarrow S_{w'}$, where $S_{w'}$ is the cluster with the maximum sum of distances between its entities and respective centroid.
4. If the algorithm reaches the aspired quantity of clusters, stop; otherwise go to Step 2. There are other possible stop conditions [11,4].

In order to implement the above, we need an algorithm to perform the splitting of step 2. One of Mirkin's suggestions is the 2-Means splitting procedure[11,4], as follows:

1. Given S_w, find the two entities with the maximum distance to be the initial centroids c_{w1} and c_{w2}.
2. Apply the K-Means algorithm to S_w, utilising c_{w1} and c_{w2} as the initial centroids.
3. Output S_{w1} and S_{w2}.

3 Datasets and Experiments

This paper presents experiments with 37 datasets, of which seven are real and 30 are synthetic. The real datasets were obtained from the Irvine machine learning repository [10]. All of the datasets were standardised, using $y_j = \frac{i_j - \bar{I}_j}{0.5range(I_j)}$,for a given feature, j, whose average over the whole dataset was $\bar{I}_j$. Using the range in the standardisation of data, rather than for instance the standard deviation, is empirically supported [20]. The categorical features followed a strategy described by Mirkin [4]: a categorical feature with Categories 1 to p is transformed into p binary features, each of which is assigned 1 if a given entity falls within the category, and zero otherwise. Each binary feature is standardised in a quantitative manner by subtracting its grand mean.

We measured the accuracy of an initialization algorithm by computing a confusion matrix. Specifically, we mapped each cluster $S_k = \{S_1, ..., S_K\}$ produced by K-Means under a given initialization to the pre-labelled K clusters with the largest overlap. The final accuracy was then the number of correctly clustered entities divided by the total number of entities.

Each table shows the initialization elapsed time, which is independent of K-Means, and the number of iterations it took K-Means to reach convergence.

3.1 Experiments with Real-World Datasets

We used the iris dataset for our first experiments. This dataset contains 150 flower specimens over four numerical features and three clusters of equal size. The results of the experiments for this dataset can be found in Table 1.

Table 1. The outcome of the experiments with the iris dataset; 150 entities over four features partitioned into three clusters. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy			K-Means iterations			Initialisation duration		
	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max
Random	0.82	0.12	0.89	8.43	3.36	19	*neg*	*neg*	*neg*
K-Means++	0.84	0.11	0.89	7.29	3.32	18	0.0007	0.00016	0.0022
iK-Means	-	-	0.89	-	-	11	-	-	0.0034
Build(PAM)	-	-	0.89	-	-	8	-	-	0.4355
H&W	-	-	0.89	-	-	5	-	-	0.0004
Agg. W.	-	-	0.89	-	-	3	-	-	0.0036
Div. W.	-	-	0.89	-	-	2	-	-	0.0256

We also experimented with the Australian credit card approval dataset. This dataset has 690 entities and two clusters over 13 features. After standardisation, this dataset contained 42 features. There were 383 and 307 entities in the partitions regarding approval and denial of credit respectively. With this dataset we obtained the results in Table 2.

For our third experiment, we used the heart disease dataset which has 270 entities and two clusters over 13 features. After standardisation, we obtained

Table 2. The outcome of our experiments with the Australian credit card approval dataset; 690 entities over 42 features and two clusters. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy			K-Means iterations			Initialisation duration		
	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max
Random	0.75	0.13	0.85	9.61	3.98	19	*neg*	*neg*	*neg*
K-Means++	0.72	0.14	0.85	9.62	4.51	21	0.0019	0.00182	0.0199
iK-Means	-	-	0.85	-	-	10	-	-	0.4606
Build(PAM)	-	-	0.85	-	-	13	-	-	4.2794
H&W	-	-	0.73	-	-	10	-	-	0.0049
Agg. W.	-	-	0.85	-	-	6	-	-	0.2076
Div. W.	-	-	0.73	-	-	2	-	-	0.8182

32 features. The clusters represent the presence and absence of heart disease in human subjects, and contained 150 and 120 entities respectively. The results for this dataset are in Table 3.

Table 3. The outcome of the experiments with the heart disease dataset; 270 entities, 32 features and two clusters. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy			K-Means iterations			Initialisation duration		
	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max
Random	0.81	0.03	0.82	7.47	2.04	14	*neg*	*neg*	*neg*
K-Means++	0.81	0.03	0.82	7.55	1.88	14	0.00073	0.00003	0.0009
iK-Means	-	-	0.82	-	-	4	-	-	0.0134
Build (PAM)	-	-	0.82	-	-	6	-	-	0.6412
H&W	-	-	0.82	-	-	9	-	-	0.0006
Agg. W.	-	-	0.82	-	-	6	-	-	0.0160
Div. W.	-	-	0.82	-	-	2	-	-	0.0613

The hepatitis dataset contains 155 entities partitioned into two clusters of 32 and 123 entities respectively, over 19 features; after standardisation, this increased to 80. The results can be found in Table 4.

Table 4. The outcome of the experiments with the hepatitis dataset; 155 entities over 80 features partitioned into two clusters. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy			K-Means iterations			Initialisation duration		
	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max
Random	0.71	0.02	0.72	8.04	2.44	15	*neg*	*neg*	*neg*
K-Means++	0.72	0.01	0.72	9.05	3.04	21	0.0009	0.00003	0.0010
iK-Means	-	-	0.72	-	-	4	-	-	0.0155
Build (PAM)	-	-	0.72	-	-	8	-	-	0.2156
H&W	-	-	0.72	-	-	6	-	-	0.0007
Agg. W.	-	-	0.72	-	-	7	-	-	0.0070
Div. W.	-	-	0.72	-	-	2	-	-	0.0385

The Pima Indian diabetes dataset consists of 768 entities, eight numerical features and two clusters. In terms of cardinality, the clusters have 500 and 268 entities. An update recently (28/02/2011) posted on the web page regarding this dataset [10], stating that some of its values are biologically implausible; however, we decided to keep the dataset unchanged for easy comparison with previously published papers. The results of this dataset can be found in Table 5.

Table 5. The outcome of the experiments with the Pima Indian diabetes dataset; 768 entities, eight features and two clusters. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy			K-Means iterations			Initialisation duration		
	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max
Random	0.67	0.004	0.67	10.04	3.31	20	*neg*	*neg*	*neg*
K-Means++	0.67	0.005	0.67	10.21	3.83	24	0.0007	0.00005	0.0010
iK-Means	-	-	0.67	-	-	4	-	-	0.0207
Build (PAM)	-	-	0.67	-	-	8	-	-	6.2721
H&W	-	-	0.67	-	-	13	-	-	0.0006
Agg. W.	-	-	0.67	-	-	9	-	-	0.0540
Div. W.	-	-	0.67	-	-	2	-	-	0.1169

We also experimented with the wine dataset. This contains 178 entities, over 13 numerical features and three clusters. The cardinalities of the clusters are 59, 71 and 48. The results can be found in Table 6.

Table 6. The outcome of the experiments with the wine dataset; 178 entities over 13 features partitioned into three clusters. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy			K-Means iterations			Initialisation duration		
	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max
Random	0.95	0.004	0.97	6.64	2.23	15	*neg*	*neg*	*neg*
K-Means++	0.95	0.005	0.97	6.84	2.35	14	0.0008	0.00005	0.0011
iK-Means	-	-	0.95	-	-	2	-	-	0.0059
Build (PAM)	-	-	0.95	-	-	3	-	-	0.6163
H&W	-	-	0.95	-	-	4	-	-	0.0004
Agg. W.	-	-	0.97	-	-	3	-	-	0.0045
Div. W.	-	-	0.95	-	-	3	-	-	0.0340

Finally, we experimented with the MAGIC Gamma Telescope dataset. This is a considerably larger real-world dataset than the other six discussed here, comprising 19,020 entities over 11 features partitioned into two clusters. The results for this dataset are contained in Table 7.

Table 7. The outcome of the experiments with the MAGIC Gamma Telescope dataset; 19,020 entities over 11 features partitioned into two clusters. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy			K-Means iterations			Initialisation duration		
	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max
Random	0.59	0.0004	0.59	24.31	6.74	41	*neg*	*neg*	*neg*
K-Means++	0.59	0.0004	0.59	24.66	6.85	39	0.0239	0.0016	0.0360
iK-Means	-	-	0.59	-	-	14	-	-	0.9182
Build (PAM)	-	-	0.59	-	-	19	-	-	7853.8
H&W	-	-	0.59	-	-	20	-	-	0.4018
Agg. W.	-	-	0.59	-	-	17	-	-	82.10
Div. W.	-	-	0.59	-	-	2	-	-	145.81

In this initial set of experiments, we were surprised by the good performance of the divisive ward algorithm in regard to the number of iterations. This kept the number of K-Means iterations from two to three and was outperformed only once by iK-Means, by a single iteration in the wine dataset(possibly the most well-separated of them). The agglomerative ward algorithm also presented a good average outcome, always providing a lower number of iterations than those obtained by random initialization. Nonetheless, it was outperformed by iK-Means in five of the seven datasets.

Regarding the initializations' overhead time, the worst case was presented by the Build algorithm; it seems clear that the number of entities in a dataset can have a drastic impact on the amount of time it takes to complete. Such an impact was also seen in both the agglomerative and divisive Wards. In our experiments, the fastest initialisation algorithm was the Hartigan and Wong [9], but unfortunately in three datasets (Australian credit card approval, Pima Indian diabetes, and heart disease), the amount of K-Means iterations given by using this initialization was higher than that obtained on average by using random initialization.

Although K-Means++ was the second fastest algorithm, it did not seem to provide an enhancement to K-Means under the specific conditions of our experiments.

Experiments with Synthetic Data

Using Netlab software [16], we experimented with Gaussian clusters. The clusters were spherical, of variance *0.1*, and had mixture coefficients of $1/K$. Their centres were independently generated from a Gaussian distribution *N(0,1)* with zero mean and unit variance.

The first set of data comprises 10 Gaussian mixtures, each with 500 entities, six features and 5 clusters. The results can be found in Table 8. We then experimented with an increased number of features and generated five other Gaussian mixtures, yet with 500 entities partitioned into five clusters, this time over 15 features. The results can be found in Table 9. In the next set of experiments, we generated a new set of five Gaussian mixtures, this time with 1,000 entities,

Table 8. The outcome of the experiments with 10 Gaussian mixtures; 500 entities, six features and five clusters, each. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy				K-Means iterations				Initialisation duration			
	Mean	Std	Max	Std	Mean	Std	Max	Std	Mean	Std	Max	Std
Random	0.80	0.07	0.84	0.05	17.39	3.19	39.7	10.13	*neg*	*neg*	*neg*	*neg*
K-Means++	0.80	0.07	0.84	0.05	17.03	3.11	37.5	8.76	0.002	0.00006	0.0027	0.0012
iK-Means	-	-	0.82	0.08	-	-	11.6	3.72	-	-	0.0133	0.0017
Build (PAM)	-	-	0.83	0.06	-	-	13.8	5.92	-	-	9.2859	0.0919
H&W	-	-	0.79	0.09	-	-	19.6	6.27	-	-	0.0006	0.00006
Agg. W.	-	-	0.83	0.07	-	-	7.6	2.32	-	-	0.0182	0.0004
Div. W.	-	-	0.83	0.06	-	-	12.1	4.38	-	-	0.1271	0.0097

Table 9. The outcome of the experiments with five Gaussian mixtures; 500 entities, 15 features and five clusters, each. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy				K-Means iterations				Initialisation duration			
	Mean	Std	Max	Std	Mean	Std	Max	Std	Mean	Std	Max	Std
Random	0.89	0.04	0.95	0.03	11.51	2.86	25.8	4.32	*neg*	*neg*	*neg*	*neg*
K-Means++	0.89	0.04	0.95	0.03	11.56	3.18	27.2	5.36	0.002	0.00005	0.0031	0.0003
iK-Means	-	-	0.95	0.03	-	-	5.8	2.28	-	-	0.0168	0.0036
Build (PAM)	-	-	0.95	0.03	-	-	9.0	2.55	-	-	9.2523	0.0465
H&W	-	-	0.95	0.03	-	-	10.8	4.87	-	-	0.0006	0.00001
Agg. W.	-	-	0.95	0.03	-	-	5.2	1.79	-	-	0.0208	0.0006
Div. W.	-	-	0.95	0.03	-	-	5.2	2.18	-	-	0.169	0.0184

Table 10. The outcome of the experiments with five Gaussian mixtures; 1,000 entities, 25 features and 12 clusters, each. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy				K-Means iterations				Initialisation duration			
	Mean	Std	Max	Std	Mean	Std	Max	Std	Mean	Std	Max	Std
Random	0.88	0.01	0.99	0.003	13.91	0.95	28.4	3.13	*neg*	*neg*	*neg*	*neg*
K-Means++	0.88	0.02	0.99	0.003	13.80	0.81	29.2	5.49	0.019	0.002	0.1057	0.0714
iK-Means	-	-	0.93	0.06	-	-	7.8	2.68	-	-	0.048	0.006
Build (PAM)	-	-	0.97	0.05	-	-	7.2	4.49	-	-	100.09	0.1567
H&W	-	-	0.78	0.01	-	-	21.2	5.21	-	-	0.0011	0.00004
Agg. W.	-	-	0.99	0.003	-	-	2.4	0.55	-	-	0.1057	0.0010
Div. W.	-	-	0.99	0.003	-	-	3.8	0.84	-	-	0.9280	0.0263

25 features and 12 clusters. Table 10 presents the results. We also generated five Gaussian mixtures, each with 1,000 entities, 50 features, and 12 clusters. Table 11 presents the results. In our final set of experiments, we decided to considerably increase the number of entities and generated five Gaussian mixtures, each comprising of 5,000 entities over 25 features, partitioned into 12 clusters. Table 12 presents the results. Regarding the number of iterations, there appear to be a small change in the pattern; the agglomerative Ward seemed to reduce the number of K-Means iterations the most, closely followed by the divisive Ward. The Build and iK-Means algorithm also showed considerably better results than random initialization.

Table 11. The outcome of the experiments with five Gaussian mixtures; 1,000 entities over 50 features partitioned into 12 clusters, each. The non-deterministic initializations (Random and K-Means++) were run 100 times.

Initializations	Accuracy				K-Means iterations				Initialisation duration			
	Mean	Std	Max	Std	Mean	Std	Max	Std	Mean	Std	Max	Std
Random	0.83	0.01	1	0	10.65	0.39	20.4	1.52	*neg*	*neg*	*neg*	*neg*
K-Means++	0.84	0.01	1	0	10.61	0.23	19.2	1.79	0.032	0.002	0.1083	0.0382
iK-Means	-	-	0.98	0.05	-	-	4.2	2.95	-	-	0.0452	0.0056
Build (PAM)	-	-	1	0	-	-	3.2	0.45 -		-	100.92	0.8027
H&W	-	-	0.73	0.09	-	-	11.6	1.14	-	-	0.0016	0.0001
Agg. W.	-	-	1	0	-	-	2	0	-	-	0.1363	0.0032
Div. W.	-	-	1	0	-	-	2.2	0.45	-	-	2.0670	0.0008

Regarding time, the Build algorithm was the worst performer. The experiments with the datasets containing 1,000 entities took on average over 10 times

Table 12. The outcome of the experiments with five Gaussian mixtures; each with 5,000 entities over 25 features partitioned into 12 clusters, each. The non-deterministic initializations (i.e. Random and K-Means++) were run 100 times.

Initializations	Accuracy				K-Means iterations				Initialisation duration			
	Mean	Std	Max	Std	Mean	Std	Max	Std	Mean	Std	Max	Std
Random	0.88	0.02	0.99	0.01	23.38	1.38	57.6	8.02	*neg*	*neg*	*neg*	*neg*
K-Means++	0.88	0.02	0.99	0.01	22.31	1.26	51.4	7.09	0.146	0.027	0.3503	0.0831
iK-Means	-	-	0.97	0.05	-	-	13.2	11.61	-	-	0.4822	0.0897
Build (PAM)	-	-	0.99	0.01	-	-	6.2	1.79	-	-	5001	426.69
H&W	-	-	0.85	0.05	-	-	32	13.47	-	-	0.7846	0.0024
Agg. Ward.	-	-	0.99	0.01	-	-	4	1.87	-	-	5.465	0.2128
Div. Ward.	-	-	0.99	0.01	-	-	5.4	2.07	-	-	38.48	3.7288

longer than the experiments with the datasets containing 500 entities. Although faster than Build, K-Means++ also had a similar increase in time between the experiments of 1,000 entities against the experiments with 500 entities. The smallest increase in time between these two groups of experiments was of about two times, with the Hartigan and Wong method, followed by iK-Means with an increase of just over three times; of course, the difference in the number of features between the two groups of experiments should not be disregarded.

Again, the Hartigan and Wong algorithm was the fastest, but unfortunately, it did not reduce the number of K-means iterations. In this regard, it always performed better than the worst scenario using random initialization, but not as well as its average in three of the five groups of experiments.

4 Conclusion and Future Research

The inability to find an initialization algorithm that works well in every scenario, in terms of accuracy, has already been established [13], and it seems this also holds true in terms of optimizing the number of K-Means iterations. Nonetheless, we observed a few patterns that may prove helpful.

Our experiments have demonstrated that initializations based on hierarchical clustering tend to find initial centroids that minimise the amount of iterations K-Means takes to converge. We can observe this in our experiments with the Agglomerative Ward and Divisive Ward criterion, but due to poor scalability these may not be the best options when dealing with large datasets. It seems that iK-Means and the Hartigan and Wong algorithm are the most scalable of the algorithms we experimented with, but the latter seems unable to minimise the number of K-Means iterations. The Build algorithm was able to produce a substantial reduction in some experiments, but it proved to be the slowest initialization algorithm in the experiments.

Regarding the assertion that the number of iterations K-Means takes to converge tends to be less than the number of entities [19], this seems true in all of the experiments. The number of entities was between 150 and 19,020 entities, 17 to 782 times bigger than the average number of K-Means iterations under a Random initialization. Other initializations, such as the Agglomerative and Divisive Ward, provided a more consistent number of iterations per dataset.

In the experiments with synthetic data, we observed that increasing only the number of features does not necessarily lead to a higher number of iterations. It produced the opposite effect in K-Means under all of the initializations we experimented with. It appears that with a large quantity of features, the clusters in these datasets are better formed and easier to cluster.

In this paper we presented a comparison regarding the number of iterations K-Means takes to converge under different initializations. In future research we intend to use real-world datasets with a larger number of clusters as well as address the amount of time the algorithm takes to converge.

References

1. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, pp. 281–297 (1967)
2. Ball, G., Hall, D.: A clustering technique for summarizing multivariate data. Behav. Sci. 12, 153–155 (1967)
3. Jain, A.K.: Data Clustering: 50 Years Beyond K-Means. Pattern Recognition Letters 31, 651–666 (2010)
4. Mirkin, B.: Clustering for Data Mining: A Data Discovery Approach. Chapman and Hall/CRC, Boca Raton (2005)
5. Pal, S.K., Mitra, P.: Pattern Recognition Algorithms for Data Mining. CRC Press, Boca Raton (2004)
6. Steinley, D., Brusco, M.: Initializing K-Means Batch Clustering: A Critical Evaluation of Several Techniques. Journal of Classification 22, 221–250 (2007)
7. Ward Jr., J.H.: Hierarchical grouping to optimize an objective function. Journal of the American Statistical Association 58, 236–244 (1963)
8. Chiang, M.M., Mirkin, B.: Intelligent choice of the number of clusters in k-means clustering: An experimental study with different cluster spreads. Journal of Classification 27(1), 1–38 (2010)
9. Hartigan, J.A., Wong, M.A.: Algorithm AS 136: A K-Means Clustering Algorithm. Journal of the Royal Statistical Society 28(1), 100–108 (1979)
10. Irvine machine learning repository (accessed September 05, 2011)
11. Mirkin, B.: Mathematical classification and clustering. Kluwer Academic Press, Dordrecht (1996)
12. Kaufman, L., Rousseeuw, P.: Finding groups in data: An introduction to cluster analysis. J. Wiley and Son (1990)
13. Maitra, R., Peterson, A.D., Ghosh, A.P.: A systematic evaluation of different methods for initializing the K-means clustering algorithm. Transactions on Knowledge and Data Engineering (2010)
14. Milligan, G.W.: The validation of four ultrametric clustering algorithms. Pattern Recognition 12, 41–50 (1980)
15. Arthur, D., Vissilvitskii, S.: K-Means++: The advantages of careful seeding. In: ACM-SIAM Symposiom on Discrete Algorithms, Astor Crowne Plaza, New Orlans, Lousiana, pp. 1–11 (2007)
16. Netlab Neural Network software, `http://www1.aston.ac.uk/eas/research/groups/ncrg/resources/netlab/` (accessed on September 01, 2011)

17. Lozano, J.A., Pena, J.M., Larranaga, P.: An empirical comparison of four initialization methods for the k-means algorithm. Pattern Recognition Lett. 20, 1027–1040 (1999)
18. Xu, R., Wunsch II, D.C.: Clustering. John Wiley and Sons (2010)
19. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. John Wiley and Sons, Inc., New York (2001)
20. Milligan, G.W., Cooper, M.C.: A study of standardization of the variables in cluster analysis. Journal of Classification 5, 181–204 (1988)
21. De Amorim, R.C.: Constrained Intelligent K-Means: Improving results with limited previous knowledge. In: Proceedings of Advanced Engineering Computing and Applications in Science, pp. 176–180. IEEE Computer Society (2008)
22. De Amorim, R.C., Mirkin, B.: Minkowski Metric, Feature Weighting and Anomalous Cluster Initializing in K-Means Clustering. Pattern Recognition (2011), doi:doi:10.1016/j.patcog.2011.08.12
23. de Amorim, R.C., Komisarczuk, P.: On Initializations for the Minkowski Weighted K-Means. In: Hollmén, J., Klawonn, F., Tucker, A. (eds.) IDA 2012. LNCS, vol. 7619, pp. 45–55. Springer, Heidelberg (2012)

Intelligent Feature and Instance Selection to Improve Nearest Neighbor Classifiers

Yenny Villuendas-Rey[1,3], Yailé Caballero-Mota[2], and María Matilde García-Lorenzo[3]

[1] Computer Science Department, University of Ciego de Ávila, Carr. a Morón km 9 ½, Cuba
yennyv@informatica.unica.cu
[2] Computer Science Department, University of Camagüey, Circunv. Norte km 3 ½, Cuba
yaile.caballero@reduc.edu.cu
[3] Computer Science Department, University of Las Villas, Carr. a Camajuaní, km 5 ½, Cuba
mmgarcia@uclv.edu.cu

Abstract. Feature and instance selection before classification is a very important task, which can lead to big improvements in both classifier accuracy and classifier speed. However, few papers consider the simultaneous or combined instance and feature selection for Nearest Neighbor classifiers in a deterministic way. This paper proposes a novel deterministic feature and instance selection algorithm, which uses the recently introduced Minimum Neighborhood Rough Sets as basis for the selection process. The algorithm relies on a metadata computation to guide instance selection. The proposed algorithm deals with mixed and incomplete data and arbitrarily dissimilarity functions. Numerical experiments over repository databases were carried out to compare the proposal with respect to previous methods and to the classifier using the original sample. These experiments show the proposal has a good performance according to classifier accuracy and instance and feature reduction.

Keywords: instance selection, object selection, rough sets, nearest neighbor.

1 Introduction

Increasing the efficiency of Case Based Reasoning techniques constitutes a significant research area in Artificial Intelligence. One of the key topics in this research is case base preprocessing. It may include instance selection or generation, feature selection, and simultaneous or combined feature and instance selection. In the latter, the algorithms select both features and instances, obtaining a highly reduced and accurate case base. Previous work done by Kuncheva and Jain [1] show that simultaneous selection of features and instances leads to better results than sequential selection. The quality of the case base is important to every supervised classifier, and Nearest Neighbor (NN) is one of the most affected by it; because it stores the case base and compares every new case with those stored, having a time and memory costs increasing with the dimensions of the case base. There are several methods to improve NN classifiers through simultaneous or combined feature and instance selection, having some drawbacks such as a stochastic nature, high computational

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 27–38, 2013.

cost, insufficient noise filtering, and inability to deal with imbalanced case bases. This paper introduces a novel deterministic method to improve NN classifier by selecting features and instances, which makes this process better than other combined methods. The main contributions of the proposal are:

1. It has strong theoretic basis, because it uses extended Rough Set Theory [2] and structuralizations of the Logical Combinatorial Approach to Pattern Recognition [3].
2. It is deterministic, and deals with mixed as well as imbalanced data.
3. It uses metadata to determine the condensing or editing strategy to follow in instance selection procedure.
4. It obtains high data reduction, maintaining the original classifier error.

The organization of the contribution is as follows: the next section covers some general concepts about extended Rough Set Theory and structuralizations of the Logical Combinatorial Approach to Pattern Recognition. Section 3 explains the proposed approach to feature and instance selection, and Section 4 contains several numerical experiments to determine the performance of the proposal with respect to other feature and object selection methods. Section 5 gives the conclusions and future works.

2 Maximum Similarity Graphs and Rough Sets

2.1 Maximum Similarity Graphs

The Logical Combinatorial approach to Pattern Recognition has several data structuralization procedures [3], which have their basis on Maximum Similarity Graph computation. A Maximum Similarity Graph (MSG) is a directed graph such that it connects each instance with all of its most similar instances. More properly, let be $G = (X, \theta)$ a MSG for a set of instances X, with arcs θ. In this graph, two instances $x_i, x_j \in X$ form an arc $(x_i, x_j) \in \theta$ if and only if $\max_{x \in X}\{sim(x_i, x)\} = sim(x_i, x_j)$, where $sim(x_i, x_j)$ is a similarity function. If there are several instances with maximum similarity with respect to an instance x (ties), the MSG establishes an arc between x and each of its more similar instances. Each connected component of a MSG is a Compact Set (CS). Compact sets guaranteed that the most similar example of each instance belong to the same compact of the instance. It is usual to construct a MSG using $sim(x_i, x_j) = 1 - \Delta(x_i, x_j)$, where $\Delta(x_i, x_j)$ is a dissimilarity function.

Maximum Similarity Graphs are the basis for several prototype selection methods, such as [4-6], and offers several advantages to data analysis. They do not assume any properties of data and do not need any parameter for their construction, except the similarity function to compare two instances. They also handle mixed as well as incomplete data.

2.2 Minimum Rough Sets as Extended Rough Sets

Rough Set Theory (RST) was proposed by Pawlak in 1982 [7]. It assumes that each object x of an universe U has related a certain amount of information, and the attributes or features that describe the object express it. In RST, the basic structure of information is the Information System. An Information System is a pair S= (U, F), where U is a non-empty finite set of objects called the Universe and $F = \{f_1, f_2, \dots, f_n\}$ is a non-empty finite set of features. The classification data form a Decision System, which is any Information System such that $DS = F \cup \{d\}$, where $d \notin F$ is the decision feature. The decision feature d induces a partition of the universe U. Let be the sets $Y_i = \{x \in U: x(d) = i\}$, $\{Y_1, \dots, Y_b\}$ is a collection of equivalence classes, named decision classes, where the objects belong to the same class if and only if they have the same value at the decision attribute d. Each subset B of F, B⊆F, has associated a binary indiscernible relation denoted by R, which is the set of object pairs which are indiscernible according to the relation [2]. An equivalence relation is an indiscernible relation defined by forming subsets of objects of U having the same values of a subset of features B of F, B⊆F.

When dealing with continuous attributes, an equivalence relation as defined previously is not appropriate, since some close values may be similar, but discernible. An extension of the classical RST is to modify the concept of indiscernible objects, such that similar objects according to a similarity relation R are together in the same class. The similarity relations generate similarity classes, for each object $x \in U$. The recently introduced Minimum Neighborhood Rough Sets [8] defines the similarity relation using Maximum Similarity Graph concepts. Two objects are similar (neighbors) if they form an arc in a Maximum Similarity Graph, that is, the Neighborhood of an object is $N_B(x_i) = \{x_j \mid (x_i, x_j) \in \theta\}$. Let be $\mathrm{Y_i} \in \mathrm{Y}$ a decision class, its positive region is as following:

$$POS_B(\mathrm{Y_i}) = x_i \left| x_i \in X, \forall_{x_j \in N_B(x_i)}, x_i(\mathrm{d}) = x_j(\mathrm{d}) = \mathrm{i} \right. \qquad (1)$$

Therefore, objects with pure neighborhood will form the positive region of the decision classes. The limit region of the decision contains objects with neighbors of different classes (equation 2). This generalization allows handling mixed data, and using specific similarity functions, without any threshold definition.

$$LIM_B(\mathrm{Y_i}) = x_i \left| x_i \in X, \exists_{x_j \in N_B(x_i)}, x_i(\mathrm{d}) \neq x_j(\mathrm{d}) \right. \qquad (2)$$

As shown, extended Rough Set Theory has several advantages to data analysis, such as it does not need any external information; no assumptions about data are necessary, and it is suitable for analyzing both quantitative and qualitative features.

3 Intelligent Feature and Instance Selection

As stated before, combined feature and instance selection (FIS) algorithms obtain better results than sequential selection [1]. This may be due to these algorithms use the information of the entire case base (CB) in order to obtain a reduced one, while in sequential selection the second method only has access to the results of the first one.

According to the nature of the selection process, FIS algorithms are stochastic or deterministic. Among stochastic algorithms, there has been and extensive use of Genetic Algorithms [9], Swarm Intelligence techniques [10], Cooperative Co-evolution [11] and Hybrid methods [12]; however, stochastic methods are beyond the scope of this paper. Much little work exists on deterministic FIS methods for mixed data. The first algorithm for this purpose is the proposed by Dasarathy in [13], which uses a wrapper selection strategy. Another deterministic algorithms are SOFSA (Simultaneous Object and Feature Selection Algorithm) [14] and TCCS (Testor and Compact set based Combined Selection) [15], which use a combined selection strategy. Although SOFSA and TCCS deal with mixed data, they do not filter noise and have a sub-matrixes fusion strategy based only on a sub-matrixes sort procedure. To overcome these drawbacks, this section introduces the proposed algorithm for feature and instance selection. The IFIS (Intelligent Feature and Instance Selection) algorithm has four steps (figure 1). Step one addresses initial noise filtering or data condensation, depending of a metadata designed to determine what action to take. Step 2 obtains several sub-matrixes using the support set (section 3.2) obtained using the entire training set and Step 3 sorts them according to classifier accuracy. Step 4 merges the sub-matrixes using a fusion procedure.

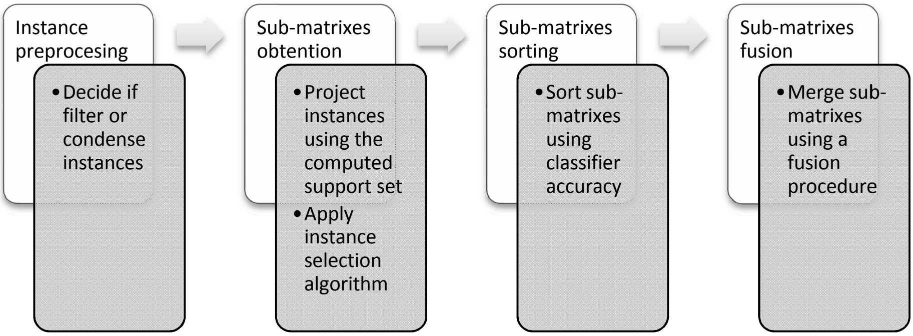

Instance preprocesing
• Decide if filter or condense instances
Sub-matrixes obtention
• Project instances using the computed support set
• Apply instance selection algorithm
Sub-matrixes sorting
• Sort sub-matrixes using classifier accuracy
Sub-matrixes fusion
• Merge sub-matrixes using a fusion procedure

Fig. 1. Proposed IFIS algorithm

3.1 Filtering Noise or Condensing Data

Among instance selection methods, there are error - based editing methods and condensing methods. Error – based editing typically delete noisy or mislabeled instances in class boundary regions, while condensing methods keep border instances and delete the redundant ones. Error – based editing tries to smooth class boundaries and decreasing Bayes error; but in some cases, they may delete an entire class. On the other hand, condensing may keep many instances, decreasing classifier accuracy. Pioneer work of Dasarathy in 2000 [16] tried to exploit the benefit of both error – based editing and condensing strategies, and minimize their weakness through a synergic exploitation. However, the question of when to apply each strategy is still open. In the authors' opinion, there are two key factors in deciding the performance of error – based editing and condensing methods: class overlapping and class imbalance. In case of imbalanced data, if the minority class has overlapping with another class, error – based editing may delete the entire minority class.

When dealing with well-separated classes, condensing methods obtain very good results, and error – based editing do not offer a good instance reduction. On the other hand, when classes have a certain degree of overlapping, and are balanced, error - based editing obtain very good performance. On the contrary, if exist a high degree of class overlapping, and balanced data, it is not clear which strategy perform better. Considering this, the IFIS algorithm uses a metadata to guide the initial instance-preprocessing step. The proposed metadata computes Class Overlapping (CO) as the maximum amount of instances of a class having a heterogeneous arc in a Maximum Similarity Graph (equation 3), and Imbalance Ratio (IR) as the ratio between the amount of instances of majority and minority class, respectively.

$$CO = \max_{i} \frac{|\{x \in T, x(d) = i | \exists \theta(x,y), x(d) \neq y(d)\}|}{|\{x \in T, x(d) = i\}|} \tag{3}$$

The metadata has the form of decision rules (figure 2) and plays a key role in given and "intelligent" decision in preprocessing stage of IFIS algorithm. If CO is "Low", IFIS applies first an editing method. On the contrary, if CO is "High", IFIS applies a condensing method. In addition, if CO is medium, IFIS analyzes IR. If IR is "Low", IFIS applies first an editing method, and if IR is "High", IFIS applies a condensing method. The values of "Low", "Medium" and "High" where obtained by discretizing CO and IR.

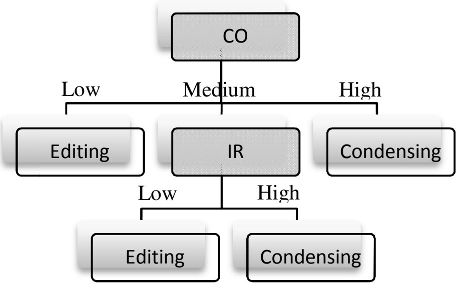


Fig. 2. Metadata rules to guide instance preprocessing. The leaves of the tree indicate the strategy to follow (editing or condensing).

The procedure for instance preprocessing using error - based editing is as follows: it computes the positive region of each decision class, according to the Minimum Neighborhood Rough Sets and deletes the object not present in any positive region. In a different way, the procedure for instance preprocessing using condensing computes the limit region of the decision, and deletes the objects not in the limit region.

Procedure for instance preprocessing

Inputs: Training set T, Attribute set B, Dissimilarity Δ.
Outputs: Preprocessed training set P

```
1. Obtain a Maximum Similarity Graph, $G = (T, \theta)$ of the objects in $T$
2. Compute Metadata and decide the strategy (editing or condensing) according to
   the rules.
3. If (strategy = editing)
   Compute the positive region of the Decision System as $POS_B(\mathrm{Y}) = \bigcup_{\mathrm{i}} POS_B(\mathrm{Y_i})$,
    where $POS_B(\mathrm{Y_i}) = x_i \,\Big|\, x_i \in X, \forall_{x_j \in N_B(x_i)}, x_i(\mathrm{d}) = x_j(\mathrm{d}) = \mathrm{Y_i}$
    Remove the objects not included in the positive region of the Decision System, as $P = T - POS_B(\mathrm{Y})$
    Else
    Compute the limit region of the Decision System as $LIM_B(\mathrm{Y}) = \bigcup_{\mathrm{i}} LIM_B(\mathrm{Y_i})$,
    where $LIM_B(\mathrm{Y_i}) = x_i \,\Big|\, x_i \in X, \exists_{x_j \in N_B(x_i)}, x_i(\mathrm{d}) \neq x_j(\mathrm{d})$
    Remove the objects not included in the limit region of the Decision System, as
    $P = T - LIM_B(\mathrm{Y})$
4.  Return $P$
```

3.2 Obtaining Sub-matrixes

In early 70's Zhuravlev and Nikiforov [17] introduce the idea of using a feature support set system to classification. A support set system is a set $S = \{F_1, \ldots, F_k\}$, where each set F_i is a feature set. Having a support set system allows using different subspaces to project instances, to improve overall classifier accuracy, as in ALVOT classifiers [3]. An example of a support set system is the set of all reducts (or typical testors) of a training data. The concept of reduct in Rough Set Theory attains to an irreducible set of features B such that using an equivalence relation R, the set of indiscernible objects using all features $IND(F)$ is equal to the set of indiscernible objects using only the features in B, $IND(B)$; that is, it preserves the partition of the universe [7]. In addition, the concept of typical testor in Logical Combinatorial approach to Pattern Recognition, first proposed by Zhuravlev in the past century [18], attains to a set of features such that it does not confuse instances of different classes and are irreducible. As shown, the concepts of reduct and typical testor, although described in different scenarios, reference the same feature sets.

To obtain sub-matrixes, IFIS needs a feature support set system and an instance selection algorithm. IFIS computes a support set system by using the entire training set, without any preprocessing. Then, the procedure projects the preprocessed instances using each feature set in the support set system, and then applies the instance selection method to each projection (figure 3).

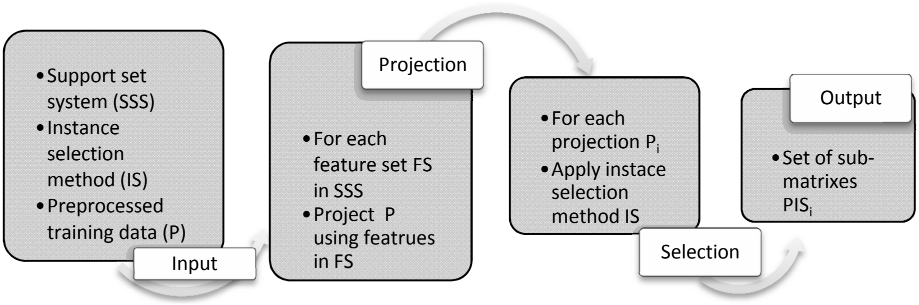

• Support set system (SSS)
• Instance selection method (IS)
• Preprocessed training data (P)
Input
Projection
• For each feature set FS in SSS
• Project P using featrues in FS
• For each projection P_i
• Apply instace selection method IS
Selection
Output
• Set of sub-matrixes PIS_i

Fig. 3. Procedure to obtain sub-matrixes

The procedure will obtain as many sub-matrixes as feature sets in the support set system. The authors consider that using an instance selection method that obtain a good representation of the entire training set will lead to better results than using error-based editing or condensing methods with low object retention rates. Section 4 discusses in detail the influence of the instance selection method in IFIS performance.

3.3 Sorting and Fusing Sub-matrixes

The IFIS algorithm obtains several sub-matrixes in the previous step. Then, it associates to each sub-matrix a fitness value that determines the quality of the sub-matrix. The fitness value may correspond to classifier accuracy, or to a Rough Set Theory measure such as Classification quality [2]. The sorting procedure sorts the sub-matrixes descendant or ascendant, depending of the fitness function. Therefore, the procedure guarantees best sub-matrixes being first in the resulting list. In this paper, IFIS uses classifier error of the training set as fitness function. Usually, the best sub-matrix obtained by the sorting procedure is worse than the original training set. Therefore, the fusion procedure (figure 4) merges it with other sub-matrixes to improve classifier accuracy.

The procedure uses a greedy approach; each iteration finds the available sub-matrix that decreases the most the classifier error. The process continues until no sub-matrixes are available or the classifier error is lower than original. The fusing procedure does not resembles the original training set, because sub-matrixes are obtained using only the instances in the preprocessed training set (section 3.1) and the feature set of the support set system (section 3.2).

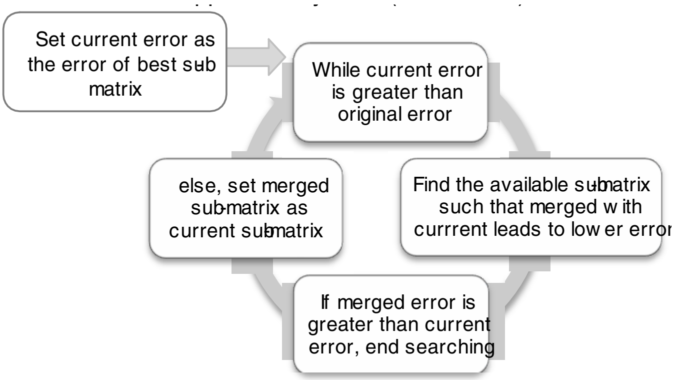

Set current error as the error of best sub matrix
While current error is greater than original error
Find the available submatrix such that merged with currrent leads to lower error
If merged error is greater than current error, end searching
else, set merged submatrix as current submatrix

Fig. 4. Procedure to fuse sub-matrixes

Due to its greedy approach, it is reasonable that the fusion strategy of IFIS will obtain better results in object retention that the one of TCCS. Section 4 addresses this topic in detail.

4 Experimental Results

4.1 Experimental Setup

This section addresses some numerical experiments to test the performance of the IFIS algorithm. The selected twenty six databases are from the Machine Learning Repository of University of California at Irvine [19]. Table 1 gives the description of them. Marked with * databases having missing values.

Table 1. Description of the databases used in numerical experiments

Databases	*Attributes (Categorical -Numerical)*	*Obj.*	*IR*	*Databases*	*Attributes (Categorical -Numerical)*	*Obj.*	*IR*
anneal*	29-9	798	86.51	heart-h*	7-6	294	1.77
autos*	10-16	205	23.13	hepatitis*	13-6	155	3.87
breast-c*	9-0	289	2.37	iris	0-4	150	1.09
breast-w	0-9	699	1.90	labor	6-8	57	1.86
car	6-0	1728	18.69	lymph	15-3	148	47.55
colic*	15-7	368	1.73	new-thyroid			5.01
credit-a*	9-6	690	1.25	tae	2-3	151	1.09
credit-g	13-7	1000	2.35	tic-tac-toe			1.89
diabetes	0-8	768	1.87	vehicle	0-18	946	1.10
ecoli	2-5	336	93.05	vote*	16-0	435	1.59
glass	0-8	214	8.48	vowel	3-9	990	1.12
hayes-roth			2.14	wine*	0-13	178	1.47
heart-c*	7-6	303	1.20	zoo	16-1	101	10.46

The fist experiment studies the influence of using positive and limit region in IFIS preprocessing. Then, subsection 4.2 explores different instance selection methods in IFIS performance. Next subsection compares the fusion strategies of TCC and IFIS and subsection 4.4 studies the performance of IFIS using different dissimilarity functions. The first three experiments use as object dissimilarity the HEOM (equation 4), and the later also uses the HVDM (equation 5), both proposed by Wilson and Martínez [20]. max_a and min_a are the maximum and minimum values of attribute a, respectively. C is the amount of classes, $N_{a,x}$ is the amount of objects having value x at attribute a, $N_{a,x,c}$ is the amount of objects of class c having value x at attribute a, and q is a constant, usually 1 or 2.

$$HEOM(x,y) = \sqrt{\sum_{a=1}^{m} d_a(x_a,y_a)}\,,\ d_a = \begin{cases} 1 \\ overlap(x_a,y_a), \\ diff(x_a,y_a) \end{cases} \tag{4}$$

$$overlap(x_a,y_a) = \begin{cases} 0 & if\ x_a = y_a \\ 1 & elsewhere \end{cases},\ diff(x_a,y_a) = |x_a - y_a|/(max_a - min_a)$$

$$HVDM(x,y) = \sqrt{\sum_{a=1}^{m} vdm_a(x_a, y_a)}, vdm_a = \sum_{c=1}^{C} \left|\frac{N_{a,x,c}}{N_{a,x}} - \frac{N_{a,y,c}}{N_{a,y}}\right|^q \tag{5}$$

4.2 Influence of the Positive and Limit Regions in IFIS

Among deterministic feature and instance selection algorithms, TCCS [15] has very good performance. This section compares the usefulness of the positive and limit region on IFIS algorithm, using TCCS and original classifier as base algorithms. To test only the influence of the preprocessing stage, in this experiment IFIS use the same sorting and fusion strategy of TCCS. Also use the same support set system and instance selection method (typical testors and CSE [4], respectively).

Cross validation is a standard procedure to compare the performance of supervised classification algorithms; therefore, all experiments use 10-fold cross validation and average results. To statistically determine if the differences in performance were significant, Demsar [21] recommends using Wilcoxon test, also employed in all experiments with a 95% of confidence. Table 2 gives the results of the Wilcoxon's test comparing IFIS using the preprocessing step (IFIS-P) with respect to Original classifier (Orig.) and TCCS. In this experiment, IFIS uses the same parameters and procedures than TCCS. Each column show the probability of the Wilcoxon's test, and the times the proposal wins, losses or ties with respect other. In bold the times the test found significant differences. Feature retention of both algorithms has no significant differences, because they use the same support set system.

Table 2. Results of the Wilcoxon test comparing IFIS preprocessing

Pair	*Error*		*Instance Retention*	
	wins-losses-ties	prob.	wins-losses-ties	prob.
Orig. vs TCCS	22-4-0	**0.000**	0-26-0	**0.000**
Orig. vs IFIS-P	15-11-0	0.082	0-26-0	**0.000**
TCCS vs IFIS-P	12-14-0	0.675	1-25-0	**0.000**

The results show that the preprocessing procedure of IFIS maintains original classifier accuracy, having no significant differences with respect to the original classifier. On the contrary, TCCS loses 22 times with respect to the original classifier error. However, IFIS-P has no differences in error with respect TCCS. According to object retention, both TCCS and IFIS-P obtain a reduced set of instances, but IFIS-P achieves much reduction than TCCS, being better on 25 databases.

4.3 Influence of the Instance Selection Method in IFIS

As stated before, the instance selection algorithm may influence the results of IFIS. This experiment compares the performance of IFIS using CSE [4] (IFIS-CSE) and CSESupport [5] (IFIS-CS) as instance selection algorithms. Both CSE and CSESupport rely on Maximum Similarity Graph computation. CSE tries to preserve the structure of data, using the sub-class consistency property [4], while CSESupport looks for a minimum consistent set. Table 3 gives the results according to classifier error and instance retention, using Wilcoxon test. Although the results show no difference in classifier error using CSE and CSESupport on IFIS (probability value greater than 0.05), the results of both methods with respect to classifier accuracy differ.

Table 3. Results of the Wicoxon test comparing different instance selection methods in IFIS

Pair	*Error*		*Instance Retention*	
	wins-loses-ties	prob.	wins-loses-ties	prob.
Orig. vs IFIS-CSE	16-10-0	0.082	0-26-0	**0.000**
Orig. vs IFIS-CS	17-9-0	**0.015**	0-26-0	**0.000**
IFIS-CSE vs IFIS-CS	14-10-2	0.189	0-23-3	**0.000**

IFIS-CSE has lower error than IFIS-CS, having a higher probability value compared with respect to the original classifier. On the other hand, IFIS-CS keeps much less objects than IFIS-CSE (due to the significant difference of both algorithms, that favors IFIS-CS). The experiment shows that IFIS is dependant of the instance selection method. The authors recommend using structure-preserving algorithms such as CSE to preserve original accuracy, and using high-condensing methods such as CSESupport to obtain as much instance reduction as possible.

4.4 Influence of the Fusion Strategy in IFIS

IFIS introduces a novel fusion strategy using a greedy approach. This subsection compares the utility of the novel strategy (IFIS-N) by comparing it with the fusion strategy of TCCS (IFIS-TC). Both algorithms use the same preprocessing step, as well as support set systems, sorting and instance selection algorithm. Table 4 gives the results according to classifier accuracy and instance retention, by means of Wilcoxon test.

Table 4. Results of the Wicoxon test comparing IFIS fusion

Pair	*Error*		*Instance Retention*	
	wins-loses-ties	prob.	wins-loses-ties	prob.
Orig. vs IFIS-N	16-10-0	**0.055**	0-26-0	**0.000**
Orig. vs IFIS-TC	16-10-0	0.082	0-26-0	**0.000**
IFIS-TC vs IFIS-N	7-3-16	0.139	0-13-13	**0.001**

The above results show the novel fusion strategy maintains classifier accuracy, tying with original classifier and IFIS-TC. In addition, it leads to a much-reduced training set, being significantly better than the fusion strategy of TCCS.

4.5 Influence of the Dissimilarity Function in IFIS

Finally, this section compares the performance of IFIS using HEOM and HVDM dissimilarities. Table 5 shows the results. Our proposal does not closely depend of the dissimilarity function used. It obtains the best results according to object reduction, being significantly better than TCCS with HEOM and HVDM dissimilarities. According to classifier error, IFIS obtains very good results. It ties with the original classifier and with TCCS.

Table 5. Results of the Wicoxon test comparing IFIS with different disimilarities

Pair	*Error*		*Instance Retention*	
	wins-loses-ties	prob.	wins-loses-ties	prob.
Orig. vs IFIS-HEOM	15-11-0	0.082	0-26-0	0.000
TCCS vs IFIS-HEOM	12-14-0	0.675	1-25-0	0.000
Orig. vs IFIS-HVDM	17-9-0	0.218	0-26-0	0.000
TCCS vs IFIS-HVDM	11-15-0	0.603	0-26-0	0.000

It is important to mention that IFIS maintains classifier accuracy using a very reduced training set. The above results show that using positive or limit regions of a Minimum Neighborhood Rough Set, leads to better results than directly use training instances. Also, the sorting and fusion strategies introduced by IFIS, obtain better results in instance retention and classifier accuracy than previuos methods.

5 Conclusions

Nearest Prototype Classification offers several advantages to Nearest Neighbor classifiers. However, it suffers dealing with mixed data is still a challenge for prototype selection algorithms. The proposed IFIS algorithm for combined feature and instance selection uses extended Rough Set Theory and structuralizations of the Logical Combinatorial Approach to Pattern Recognition to instance preprocessing, deciding the best editing or condensing strategy through a metadata computation. IFIS is deterministic, and deals with mixed as well as imbalanced data. The experimental results show the sorting and fusion strategies introduced by IFIS, obtain better results in instance retention and classifier accuracy than previuos methods, with high data reduction and maintaining the original classifier error.

References

1. Kuncheva, L.I., Jain, L.C.: Nearest neighbor classifier: Simultaneous editing and feature selection. Pattern Recognition Letters 20, 1149–1156 (1999)
2. Pawlak, Z., Skowron, A.: Rough sets: Some extensions. Information Sciences 177, 28–40 (2007)
3. Ruiz-Shulcloper, J., Abidi, M.A.: Logical combinatorial pattern recognition: A Review. In: Pandalai, S.G. (ed.) Recent Research Developments in Pattern Recognition. Transword Research Networks, USA, pp. 133–176 (2002)

4. García-Borroto, M., Ruiz-Shulcloper, J.: Selecting Prototypes in Mixed Incomplete Data. In: Sanfeliu, A., Cortés, M.L. (eds.) CIARP 2005. LNCS, vol. 3773, pp. 450–459. Springer, Heidelberg (2005)
5. García-Borroto, M., Villuendas-Rey, Y., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F.: Finding Small Consistent Subset for the Nearest Neighbor Classifier Based on Support Graphs. In: Bayro-Corrochano, E., Eklundh, J.-O. (eds.) CIARP 2009. LNCS, vol. 5856, pp. 465–472. Springer, Heidelberg (2009)
6. García-Borroto, M., Villuendas-Rey, Y., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F.: Using Maximum Similarity Graphs to Edit Nearest Neighbor Classifiers. In: Bayro-Corrochano, E., Eklundh, J.-O. (eds.) CIARP 2009. LNCS, vol. 5856, pp. 489–496. Springer, Heidelberg (2009)
7. Pawlak, Z.: Rough Sets. International Journal of Information & Computer Sciences 11, 341–356 (1982)
8. Villuendas-Rey, Y., Caballero-Mota, Y., García-Lorenzo, M.M.: Using Rough Sets and Maximum Similarity Graphs for Nearest Prototype Classification. In: Alvarez, L., Mejail, M., Gomez, L., Jacobo, J. (eds.) CIARP 2012. LNCS, vol. 7441, pp. 300–307. Springer, Heidelberg (2012)
9. Ahn, H., Kim, K.J., Han, I.: A case-based reasoning system with the two-dimensional reduction technique for customer classification. Expert Systems with Applications: An International Journal 32, 1011–1019 (2007)
10. Sakinah, S., Ahmad, S., Pedrycz, W.: Feature and Instance selection via cooperative PSO. In: IEEE International Conference on Systems, Man and Cybernetic, pp. 2127–2132. IEEE Publishing (2011)
11. Derrac, J., García, S., Herrera, F.: IFS-CoCo in the Landscape Contest: Description and Results. In: Ünay, D., Çataltepe, Z., Aksoy, S. (eds.) ICPR 2010. LNCS, vol. 6388, pp. 56–65. Springer, Heidelberg (2010)
12. Derrac, J., Cornelis, C., Gaecía, S., Herrera, F.: Enhancing evolutionary instance selection algorithms by means of fuzzy rough set based feature selection. Information Sciences 186, 73–92 (2012)
13. Dasarathy, B.V.: Concurrent Feature and Prototype Selection in the Nearest Neighbor Decision Process. In: 4th World Multiconference on Systemics, Cybernetics and Informatics, Orlando, USA, vol. VII, pp. 628–633 (2000)
14. Villuendas-Rey, Y., García-Borroto, M., Medina-Pérez, M.A., Ruiz-Shulcloper, J.: Simultaneous Features and Objects Selection for Mixed and Incomplete Data. In: Martínez-Trinidad, J.F., Carrasco Ochoa, J.A., Kittler, J. (eds.) CIARP 2006. LNCS, vol. 4225, pp. 597–605. Springer, Heidelberg (2006)
15. Villuendas-Rey, Y., García-Borroto, M., Ruiz-Shulcloper, J.: Selecting Features and Objects for Mixed and Incomplete Data. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) CIARP 2008. LNCS, vol. 5197, pp. 381–388. Springer, Heidelberg (2008)
16. Dasarathy, B.V., Sanchez, J.S., Townsend, S.: Nearest Neighbour Editing and Condensing Tools - Synergy Exploitation. Pattern Analysis & Applications 3, 19–30 (2000)
17. Zhuravlev, Y.I., Nikiforov, V.V.: Recognition algorithms based on voting calculation. Journal Kibernetika 3, 1–11 (1971)
18. Lazo-Cortés, M., Ruiz-Shulcloper, J., Alba-Cabrera, E.: An overview of the evolution of the concept of testor. Pattern Recognition 34, 753–762 (2001)
19. Merz, C.J., Murphy, P.M.: UCI Repository of Machine Learning Databases. University of California at Irvine, Department of Information and Computer Science, Irvine (1998)
20. Wilson, R.D., Martinez, T.R.: Improved Heterogeneous Distance Functions. Journal of Artificial Intelligence Research 6, 1–34 (1997)
21. Demsar, J.: Statistical comparison of classifiers over multiple datasets. The Journal of Machine Learning Research 7, 1–30 (2006)

A Method for Building Prototypes in the Nearest Prototype Approach Based on Similarity Relations for Problems of Function Approximation

Marilyn Bello-García, María Matilde García-Lorenzo, and Rafael Bello

Deparment of Computer Sciences, Universidad Central Marta Abreu de las Villas, Santa Clara,
Villa Clara, Cuba
mbgarcia@uclv.edu, {mmgarcia,rbellop}@uclv.edu.cu

Abstract. In this article, the problem of function approximation is studied using the paradigm of the nearest prototypes. A method is proposed to construct prototypes using similarity relations; the relations are constructed using the measurement quality of similarity and the metaheuristic UMDA. For every class of similarity, a prototype is constructed. The experimental results show that the proposed method achieves a significant reduction of the quantity of instances to consider, while significant differences do not exist with regard to the performance reached with all the instances.

Keywords: nearest prototype, similarity relations, function approximation.

1 Introduction

Most of the automatic learning methods construct a model during the learning process, but there are some approaches where the algorithm does not need a model, these algorithms are known as methods of lazy learning; the paradigms based on instances are inside this type of machine learning technique. In supervised learning, every instance describes itself as a pair X= (**x**, y), where **x** is a vector of predictor (condition) features and y is a decision (goal) feature. Learning algorithms based on instances present problems of scalability when the size of the training set grows, because its time of classification is proportional to the size of the classifier; that is, the quantity of training instances affects the computational cost of the method [1]; according to [2], the rule Nearest Neighbor is an example of this for its high computational cost when the quantity of instances grows.

This problem has been faced by means of the modification of the set of instances [2] and [3]. According to [4], these methods can qualify in Edition or Reduction. The first one is a step in the learning process entrusted to increase the precision of the learning method, when a substantial quantity of noise exists in the information of training sets. The step Reduction is directed to determine a small set of original instances or prototypes without a significant degradation in the precision of the classification.

There is an alternative known as Nearest Prototype (NP) [5]. The idea is to determine the value of the decision feature of a new object analyzing his similarity

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 39–50, 2013.

with regard to a set of prototypes, selected or generated from an initial set of instances. The prototypes represent the typical characteristics of a set of instances instead of necessary or sufficient conditions; the prototypes can be abstractions of the same instances previously observed, or they can be the directly observed examples [6]. To learn prototypes is to represent the information of the training sets as a set of points in the space of the application domain, called prototypes, the decision value of a new point is calculated using the decision value of one or more prototypes.

The intention of the NP approach is to decrease the costs of storage and processing of the learning techniques based on instances. According to [7], when there is an enormous quantity of information, a possible solution is to reduce the quantity of vectors “examples”, while the efficacy in the solution of problems based on the reduced data is supported so well or almost so well as when the original set of information is used. Therefore, a good set of prototypes have two desirable properties: minimum cardinality and maximum precision in the solving problem process (minimal error in the case of the functions approach) [5]. Strategies are needed to reduce the quantity of examples of the input data to a representative small quantity, and its performance must be evaluated in terms of the classification (or function approximation) precision and the reduction coefficient [7].

A small set of prototypes has the advantage of a low computational cost and small requests of memory, while it achieves a similar efficacy, and even better than all the instances. There are two strategies to construct the set of prototypes in order to reduce the quantity of instances: Selection and Replacement [8]. In the first case, a limited quantity of points of the original set of data X is preserved. In the second alternative, the set of original data is replaced for a number of prototypes that not necessarily coincides with some original instances. The algorithms to find prototypes can also qualify as determining or non-determining, depending on if they can or cannot control the quantity of prototypes generated by the algorithms [7].

Different strategies have been followed to solve the problem of finding a set of prototypes, based on clustering techniques (the data set X is processed by any algorithm of clustering, which generates centers of clusters, and takes the centroids as the set of prototypes) [9], [10]; learning vector quantization (LVQ) [11]; evolutionary approach [12], [13], [14], [15]; etc.

In [7] appears a study of several families of schemes for the prototypes reduction (prototype reduction schemes, PRS), based on the strategy of replacement; they suggest that it does not seem to be clear that any scheme is clearly superior to others, on the contrary, it seems that every method has its advantages. The experimental results in [7] show that any specific method produces the best results for all the applications in terms of reduction coefficient and classification precision; the best method for a data set is not the best for another. Nevertheless, the authors suggest that the family of methods which create prototypes is generally superior to the family of methods based on the selection of prototypes.

An important component in the NP learning is a similarity function which calculates the similarity between the inputs and the prototypes. The Euclidean distance is usually selected. If the new case **x** and the prototypes are represented by vectors of real values, the similarity almost always is based on some function of distance between **x** and the prototypes. Nevertheless, in case of mixed information this approach is not applicable. To overcome this problem there have been proposed several algorithms that replace the Euclidean standard metrics with a function that allows executing the comparisons. So, the selection of a correct function to calculate the similarity between the inputs and the prototypes is an important topic in the approaches based on prototypes [16].

The approach of the nearest prototypes has been used traditionally in problems of classification, in which the decision value is a discrete value that represents a class. Using the set of prototypes is possible to construct a classifier of the following form: Let x be a non-labeled object which is meant to be classified. The rule of classification, standard NP (1-np), assigns to x the class of the "most similar" prototype in the set of labeled prototypes [10].

The case of the regression problems is studied in [17]. The application of the NP paradigm to the problems of function approximation has its peculiarities; since, how the concept of class does not exist, the construction of the prototypes is different.

The function approximation is the problem of finding the existing relations in a finite data set, where the decision feature has a real value. The problem is defined in the following way: Let be a set of M vectors of information input - output in the form ($\mathbf{x}_i$, f($\mathbf{x}_i$)). Let be $\mathbf{x}_i \in \Re^n$ a real numbers´ vector and f($\mathbf{x}_i$)=di $\in \Re^1$, where (i=1, …, M). The problem is to find the unknown function f($\mathbf{x}$): $\Re^n \rightarrow \Re^1$ that satisfies the interpolation where f($\mathbf{x}_i$)=di, i=1,…,M. The quality of the approximation of f is usually calculated by a measurement of the error produced by the function, according to the expression (1):

$$E(f)=\frac{1}{2}*\sum_{i=1}^{M}(d_i-f(x_i))^2 \ . \tag{1}$$

The target is to find the function f which minimizes the expression (1); this means that, to improve the precision of the estimation is the principal target in the problem of the function approximation. The function approximation can turn of a more general form, thinking that xi can be a vector of mixed values (real values or symbolic values can be the domain of the dimensions), also called mixed data, instead of a vector of real values.

In this work, the problem of the function approximation is studied using the NP approach. The idea is building prototypes using a subset of similar instances. A method is proposed for the construction of prototypes using relations of similarity. For every class of similarity a prototype is constructed. The key problem is how to determine the similarity between instances; in this case, the paper proposes a similarity relation based on a weighted sum, where the weights of the features are calculated by an optimization process based on the heuristic method UMDA (Algorithm of Unvaried Marginal Distribution) and the measure quality of similarity as a fitness function. The proposed method can be qualified as of replacement, reduction, and non-determining.

2 A Method for the Function Approximation Based on Prototypes

Let be the X = {X_i: i=1, … m_1} a data set, every instance X_i is a pair ($\mathbf{x}_i$, y_i), where $\mathbf{x}_i$ is usually an input vector (which dimensions correspond to predictor features) and $y_i \in \Re$ (target feature), and the vector $\mathbf{x}_i \in \Re^n$; but, the method proposed here allows to work with mixed data. The target is to construct a set of m_2 prototypes (m_2 is significantly lower than m_1). The Algorithm NP-FA (Nearest Prototype - Function

Approximation), figure 1, uses this set of prototypes P to infer the output value for a new problem $\mathbf{x}_{new}$.

Input: A set of prototypes P; the problem $\mathbf{x}_{new}$.
Output: The output value y_{new} for the problem $\mathbf{x}_{new}$.
F1: To calculate the similarity between the problem $\mathbf{x}_{new}$ and every prototype P_i.
F2: To find the set of prototypes most similar to the problem $\mathbf{x}_{new}$, denoted for $NP(x_{new})$.
F3: To calculate the output value, denoted for y_{new}, using expression (2).

Fig. 1. Algorithm NP-FA (Nearest Prototype - Function Approximation)

$$y_{new} = \frac{\sum_{Pj \in NP(x_{new})} y_j}{|NP(x_{new})|} \quad (2)$$

Expression (3), denominated as global similarity, is used to calculate the similarity between the problem xnew and each prototype Pj:

$$F(x_{new}, P_j) = \sum_{i=1}^{n} w_i * sim_i(x_i, y_i) \quad (3)$$

The weights w_i, are usually normalized such that $\Sigma w_i=1$ are used to strengthen or to debilitate the relevancy of every dimension. The function sim_i $(\mathbf{x}_i, \mathbf{y}_i)$ measures the grade of similarity between the features of the vectors **x** and **y** in accordance with the dimension i (named local similarity). The use of weighted similarity functions and numerous metric and non-metric functions have been studied by different authors as [16] and [4]. The employment of local similarity measurements for every feature allows working with mixed data.

The principal component for any algorithm based on NP is the procedure to construct the prototypes using the set of instances. In this work the algorithm NP-BASIR is proposed for the construction of prototypes. The similarity relation R_1 is used between two instances X_1 and X_2 defined by (4):

$$X_1 \ R_1 \ X_2 \text{ iff and only iff } F(\mathbf{x}_1,\mathbf{x}_2) \geq e_1 \text{ and } sim(y_1,y_2) \geq e_2 . \quad (4)$$

Where X_1, $X_2 \in X$, e_1 y e_2 are thresholds, and *sim* is the local similarity measurement between the values of the decision feature of the instances X_1 and X_2.

The algorithm NP-BASIR, see figure 2, builds a set of prototypes P using the instances in X. For instances X_i in X, the similarity class of X_i using (4) is built; that is, the subset of instances in X that are similar to X_i according to the relation defined by (4). A prototype is generated using the instances in this similarity class. An aggregation operator Γ is used to build the prototype.

A data structure called Used[] is used to control which instances have been considered by the algorithm, in which Used[i] has a value of 1 if the instance i has been employed by the algorithm NP-BASIR, or 0 in other case.

```
Input: A set of instances X.
Output: A set of prototypes P.
NP1: Initialize the data structure Used.
     Used[j] ← 0, for j=1… $m_1$
     P ← $\phi$
     i ← 0
NP2: Starts the processing of the instance Ii
     i ← index of the first non-used instance in X (that is, Used[i]=0)
     If i=0 so, Finish the process of prototypes´ construction,
     else Used [i] ← 1.
NP3:  Construct the similarity class of the instance Ii according to $R_1$.
      Construct $[X_i]_{R1}$, $[X]_R$ denotes the similarity class of the instance X
      according to the similarity relation $R_1$.
NP4: Generation of a new prototype $P_j$.
      $P_j(i) \leftarrow \Gamma(V_i)$, where $V_i$ is the set of values of the feature i in the instances in
                          $[X_i]_{R1}$.
NP5: Add a new prototype
       P ← P ∪ Pj
NP6: Go to NP2.
```

Fig. 2. Algorithm NP-BASIR (Nearest Prototype Based on Similarity Relations)

In the step NP4, Γ denotes an aggregation operator. The intention is to construct a prototype or centroid for a set of similar objects. Here, a similar approach to those employed in other methods like [18] and [19] has been used to construct the centroid.

The performance of the algorithms NP-FA and NP-BASIR has been studied using the method of construction of similarity relations proposed in [20], [21] and [22]. In this case, the target is to find the relation R_1 which maximizes the measure quality of similarity defined by (5):

$$\theta(X) = \left\{ \frac{\sum_{i=1}^{m_1} \varphi(X_i)}{m_1} \right\} \tag{5}$$

Where $\varphi(X_i)$ is calculated for every $X_i \in X$ instance and it indicates the amount of objects in the intersection of the sets $N_1(X_i)$ and $N_2(X_i)$ which respectively denote the objects that are similar to X_i according to the predictor features ($\mathbf{x}_i$) and those that are similar to X_i according to the decision feature (y_i); to construct N_1 the similarity function F defined by (3) is used and also a threshold e_1, whereas to construct N_2 a local similarity measure and a threshold e_2 are used. Using the weighted sum defined by (3), and considering the functions of comparison for every feature, the problem is reduced to finding the set of weights W = {w_1, w_2..., w_n}, to construct F. To find the set of weights W associated with the predictor features, a method based on heuristic search is used, as is described in the section 3.

3 Experimental Results

The performance of the algorithms NP-FP and NP-BASIR has been studied using several datasets of UCI Repository[1]; these sets are described in the Table 1.

Table 1. Data set of UCI Repository used in the experimental study

Data sets	Number of instances	Number of features
baskball	96	5
bodyfat	252	15
detroit	13	14
diabetes_numeric	43	3
elusage	55	3
fishcatch	158	8
pollution	60	16
pwlinear	200	11
pyrim	74	28
schlvote	37	6
sleep	51	8
veteran	137	8
vineyard	52	4
ipi	92	20
cpu_act	1000	22
elevators	1000	19
fried	1000	11
cpu_act	2000	22
mv	2000	11

In the experiments, the local similarity measurement defined by (6) is used in the expression (3), which can be used for both, continuous and discrete domains, where Di denotes the domain of the feature i:

$$sim_i(x,y)=\begin{cases}1-\dfrac{|x_i-y_i|}{Max(D_i)-Min(D_i)} & if\ \ i\ is\ \ continuous\\ 1 & if\ \ i\ is\ \ discrete\ \ \ and\ \ \ x_i=y_i\\ 0 & if\ \ i\ is\ \ discrete\ \ \ and\ \ \ x_i\neq y_i\end{cases} \tag{6}$$

In the step NP4 of the NP-BASIR algorithm, the following operator of aggregation is used, Γ: If the values in v_i are real numbers the average is calculated; if they are discrete, the most common value is determined.

[1] http://www.ics.uci.edu/mlearn/MLRepository.html

The performance of the method for the function approximation proposed in this work formed by the algorithm NP-FP and the algorithm NP-BASIR, which was named NpBasirRegression, is compared with the method of k-Nearest Neighbors (k-NN). Different values are used for k (1 and 3), both for k-NN and for the proposed method. For the thresholds e_1 and e_2, which determine the similar instances with regard to the predictor and decision features respectively, there are used values as 0.85 and 0.95. Also the expression (6) is used as local similarity measurement for the decision feature.

The reduction coefficient Re is used (·), [7], defined by the expression (7) to evaluate the reduction of the amount of instances obtained by NP-BASIR:

$$\mathbf{Re(.)} = \frac{|\boldsymbol{X}| - |\boldsymbol{P}|}{|\boldsymbol{X}|} * \mathbf{100} \quad (7)$$

Taking into account the main parameters of the method are the thresholds e_1 and e_2 and k, six alternatives were studied defined according to the parameters e_1, e_2 and k, to accomplish the comparative study between the method NpBasirRegression and the method k-NN:

e_1=e_2=0.85, k=1 for NpBasirRegression and K-NN
e_1=e_2=0.95, k=1 for NpBasirRegression and K-NN
e_1=e_2=0.85, k=1 for NpBasirRegression, and k=3 for K-NN
e_1=e_2=0.95, k=1 for NpBasirRegression, and k=3 for K-NN
e_1=e_2=0.85, k=3 for NpBasirRegression and K-NN
e_1=e_2=0.95, k=3 for NpBasirRegression and K-NN

In this experimental study the cross-validation and different statistical tests to evaluate the performance of the proposed method were used. The method cross-validation [23] divides the sets of information in k subsets of equal size. Every subset is used to evaluate the given aproximator, while it trains with k-1 sets remaining. Then, the errors of the approximation are obtained of the average of k-1 results. The value used is k=10. To measure the error committed on approximating, the average of the absolute error (PMD) was used, that is, the average of the differences between the expected value and the real value produced by the method [24].

SPSS was used in the statistical processing of all the experimental results as the statistical tool. For the statistical analysis of results, non-parametrical tests were included, such as Friedman and Wilcoxon's tests, which allow studying the existence or not of significant differences between the samples.

The method UMDA (Algorithm of Unvaried Marginal Distribution) is used to find the weights used in the similarity function defined by (3) (which is included in the similarity relation used in NP-FA and NP-BASIR, as well as in the recovery of the similar cases of the method k-NN). UMDA method belongs to the class of algorithms with Estimation of Distributions (EDA) [25] and [26], in them neither crossing nor mutation is used, the new population is generated from the distribution of the estimated probability of the chosen set. The measure quality of similarity defined by (5) is used as function of heuristic evaluation (fitness function). In this experimentation the following values were used as the parameters: number of individuals 40, quantity of generations 100 and per cent of selection for truncation 0.5.

In tables 2 and 3, the averages of the absolute error (PMD) are shown. The application of the Friedman's test showed that significant differences do not exist since the significance is greater than 0.05.

Table 2. Results of average of the absolute error with NpBasirRegression

Data sets	Results for each alternative					
	1	2	3	4	5	6
baskball	0.09	0.09	0.08	0.08	0.08	0.70
detroit	28.29	30.89	25.60	29.73	37.82	33.80
diabetes_numeric	0.60	0.68	0.53	0.60	0.61	0.52
elusage	10.01	11.57	9.37	11.57	7.92	10.56
pollution	43.40	42.88	42.68	41.35	41.59	38.07
pwlinear	2.31	2.33	2.07	2.48	1.83	2.16
pyrim	0.08	0.06	0.07	0.06	0.07	0.06
schlvote	76.13	76.13	90.69	76.13	91.48	95.51
sleep	2.65	2.18	2.51	1.95	2.65	2.64
veteran	100.4	101.56	94.19	105.95	99.91	94.49
vineyard	2.39	2.23	2.11	2.20	2.17	1.94
ipi	8.38	5.29	7.61	5.16	8.08	5.64
cpu_act	3.94	-	4.11	-	6.55	-
fried	3.27	-	2.45	-	2.63	-
mv	7.71	-	5.12	-	4.70	-

Table 3. Results of average of the absolute error with k-NN

Data sets	Results for each alternative					
	1	2	3	4	5	6
baskball	0.10	0.09	0.08	0.08	0.09	0.70
detroit	28.35	30.89	32.15	37.60	34.80	33.80
diabetes_numeric	0.64	0.69	0.54	0.52	0.67	0.52
elusage	10.38	16.04	8.11	17.11	8.19	18.02
pollution	49.62	44.22	42.54	38.77	43.69	38.12
pwlinear	2.36	2.33	1.66	2.23	1.90	2.16
pyrim	0.08	0.06	0.06	0.06	0.06	0.06
schlvote	53.52	72.98	53.10	63.27	53.43	70.86
sleep	2.65	2.18	2.48	2.72	2.63	2.81
veteran	102.58	101.56	101.57	104.40	100.33	94.49
vineyard	2.41	2.23	2.18	1.92	2.07	1.95
ipi	5.25	5.25	4.88	5.80	4.38	5.51
cpu_act	2.64	-	2.33	-	2.26	-
fried	3.41	-	2.22	-	2.55	-
mv	5.12	-	1.46	-	1.84	-

Table 4 shows the effects of the reduction, this includes the reduction coefficient according to maximum and minimal quantity of prototypes; for instances, the database mv has 2000 instances, this quantity is reduced in the first alternative to a maximum of 14 (reduction of 99. 3%) and a minimum of 12 (99.4). In the case of the biggest size databases (1000 instances or more) only the alternatives (1), (3) and (5)) with e_1=0.85 and e_2=0.85 are shown, because they produce the greatest reduction.

Table 4. Results of the reduction of the set of instances when approximating with NpBasirRegression

Data set (number of instances)	Coefficient of reduction Maximum/Minimum					
	1	2	3	4	5	6
baskball (96)	70.8/76.0	9.38/10.4	75.0/80.2	9.38/11.5	77.1/83.3	9.38/11.5
detroit (13)	30.8/46.2	7.69/15.4	30.8/46.2	7.69/15.4	30.8/46.2	7.69/15.4
diabetes_numeric (43)	58.1/67.4	20.9/27.9	67.4/74.4	9.30/11.6	69.8/72.1	9.30/14.0
elusage (55)	80.0/85.5	12.7/14.5	67.3/70.9	10.9/12.7	65.5/70.9	10.9/14.5
pollution (60)	80.0/86.7	18.3/21.7	85.0/88.3	10.0/10.0	81.7/85.0	15.0/20.0
pwlinear (200)	27.5/29.0	10.0/10.0	39.5/43.5	10.0/10.0	34.5/39.5	10.0/10.5
pyrim (74)	85.1/91.9	54.1/62.2	85.1/91.9	51.6/54.1	83.8/87.8	54.1/58.1
schlvote (37)	8.11/10.8	8.11/10.8	32.4/43.2	8.11/10.8	13.5/21.6	8.11/10.8
sleep (51)	58.8/62.7	19.6/25.5	58.8/64.7	17.6/23.5	56.9/58.8	15.7/21.6
veteran (137)	21.9/26.3	9.49/10.2	14.6/19.7	9.49/10.2	16.1/21.2	9.49/10.2
vineyard (52)	71.2/75.0	9.62/11.5	73.1/78.8	26.9/36.5	69.2/76.9	9.62/
ipi (92)	80.4/83.7	47.8/52.2	81.5/84.8	47.8/51.1	80.4/82.6	51.1/53.3
cpu_act (1000)	98.6/99.0	-	98.5/98.8	-	98.5/98.9	-
fried (1000)	89.2/90.3	-	93.5/94.5	-	71.2/72.7	-
mv (2000)	99.3/99.4	-	98.5/98.7	-	99.1/99.2	-

It is possible to appreciate that the proposed method achieves a substantial reduction of the quantity of instances (in the majority of the cases a reduction about to 80 per cent of the number of instances), preserving the precision. Figure 3 shows a the reduction achieved by NpBasirRegression. This result is very important due to the computational complexity of the lazy methods (such as k-NN and nearest prototype) depends on the quantity of instances; when the number of instances decreases, the computational cost is reduced.

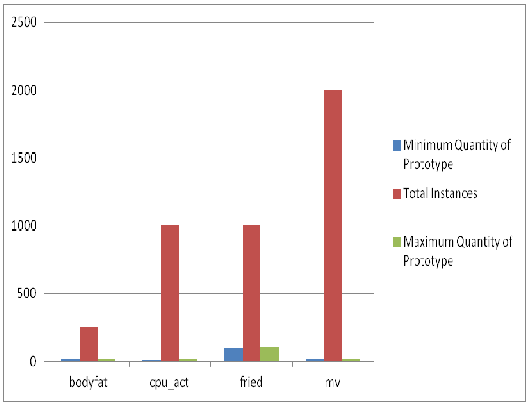


Fig. 3. Examples of the reduction of quantity of instances

Also, there was compared the average absolute error committed on approximating with NpBasirRegression with other aproximators of functions (Multilayer Perceptron, Linear Regression and Regression Tree) that appear in WEKA [27], using the 16 data sets previously described, see Table 5. The application of the Friedman´s test proved that significant differences do not exist since the significance is greater than 0.05.

Table 5. Results of average of the absolute error with Multilayer Perceptron, Linear Regression and Regression Tree N

Base de casos	Resultados		
	Multilayer Perceptron	Linear Regression	REPTree
baskball	0.08	0.07	0.08
bodyfat	0.60	0.52	0.78
detroit	33.06	44.92	55.91
diabetes_numeric	0.55	0.50	0.57
elusage	10.76	8.71	10.92
pollution	47.39	32.44	50.16
pwlinear	1.68	1.80	1.65
pyrim	0.08	0.07	0.10
schlvote	111.53	78.38	77.14
sleep	3.16	2.99	2.76
veteran	164.24	92.31	104.49
vineyard	2.09	2.30	2.45
ipi	9.57	10.80	5.41
cpu_act	2.46	6.05	2.75
fried	1.20	2.08	2.30

4 Conclusions

The method for the construction of prototypes based on similarity relations allows obtaining sets of prototypes with a significant decrease of the quantity of instances. The experimental results showed that there are not significant differences between the accomplished efficacies in the function approximation using the prototypes, with regard to the one that is reached using all the instances when there is applied the rule of the most similar neighbors. Also, that there are not significant differences between the efficacy of the approximation of functions obtained using the set of prototypes and the rule of the most similar neighbors and other aproximators like Multilayer Perceptron, Linear Regression and Regression Tree.

References

[1] Garcia-Duran, R., Fernando Fernandez, F., Borrajo, D.: A prototype-based method for classification with time constraints: a case study on automated planning. Pattern Anal. Applic. (2010), doi:10.1007/s10044-010-0194-6

[2] Barandela, R., et al.: The nearest neighbor rule and the reduction of the training sample size. In: Proceedings 9th Symposium on Pattern Recognition and Image Analysis, Castellon, España, vol. 1, pp. 103–108 (2001)

[3] Barandela, R., Gasca, E., Alejo, R.: Correcting the Training Data. In: Chen, D., Cheng, X. (eds.) Pattern Recognition and String Matching. Kluwer (2002)

[4] Lozano, M., Sotoca, J.M., Sanchez, J.S., Pla, F., PeRkalska, E., Duin, R.P.W.: Experimental study on prototype optimisation algorithms for prototype-based classification in vector spaces. Pattern Recognition 39, 1827–1838 (2006)

[5] James, C., Bezdek, J., Kuncheva, L.: Nearest Prototype Classifier Designs: An Experimental Study. International Journal of Intelligent Systems 16, 1445–1473 (2001)

[6] Gagliardi, F.: The Necessity of Machine Learning and Epistemology in the Development of Categorization Theories: A Case Study in Prototype-Exemplar Debate. In: Serra, R., Cucchiara, R. (eds.) AI*IA 2009. LNCS(LNAI), vol. 5883, pp. 182–191. Springer, Heidelberg (2009)

[7] Kim, S.-W., Oommen, B.J.: A brief taxonomy and ranking of creative prototype reduction schemes. Pattern Anal Applications 6, 232–244 (2003)

[8] Jin, X.-B., Liu, C.-L., Hou, X.: Regularized margin-based conditional log-likelihood loss for prototype learning. Pattern Recognition 43, 2428–2438 (2010)

[9] Kuncheva, L.I., Bezdek, J.C.: Nearest Prototype Classification: Clustering, Genetic Algorithms, or Random Search? IEEE Transactions on Systems, Man and Cybernetics 28(1), 160–164 (1998)

[10] Kuncheva, L.I., Bezdek, J.C.: An integrated framework for generalized Nearest prototype classifier design. International Journal of Uncertainty, Fuzziness, and Knowledge Based Systems 6(5), 437–457 (1998)

[11] Kim, S.-W., Oommen, B.J.: Enhancing prototype reduction schemes with LVQ3-type algorithms. Pattern Recognit. 36(5), 1083–1093 (2003)

[12] Fernando, F., Isasi, P.: Evolutionary Design of Nearest Prototype Classifiers. Journal of Heuristics 10, 431–454 (2004)

[13] Gustavo Recio, G., Saez, Y., Isasi, P.: Improved Accuracy Rates of a Prototype Based Classifier Using Evolutionary Computation. In: Ninth International Conference on Intelligent Systems Design and Applications, pp. 491–496 (2009)

[14] Triguero, I., García, S., Herrera, F.: IPADE: Iterative Prototype Adjustment for Nearest Neighbor Classification. IEEE Transactions on Neural Networks 21(12), 1984–1990 (2010)

[15] Triguero, I., García, S., Herrera, F.: Differential evolution for optimizing the positioning of prototypes in nearest neighbor classification. Pattern Recognition 44, 901–916 (2011)

[16] Hammer, B., Strickert, M., Villmann, T.: On the Generalization Ability of GRLVQ Networks. Neural Processing Letters 21, 109–120 (2005)

[17] Grbovic, M., Vucetic, S.: Regression Learning Vector Quantization. In: 2009 Ninth IEEE International Conference on Data Mining (2009), doi:10.1109/ICDM.2009.145

[18] Wu, X., Kumar, V., et al.: Top 10 algorithms in data mining. Knowledge Information System 14, 1–37 (2008)

[19] Mitra, S., et al.: Shadowed c-means: Integrating fuzzy and rough clustering. Pattern Recognition 43, 1282–1291 (2010)

[20] Filiberto, Y., Bello, R., et al.: Using PSO and RST to Predict the Resistant Capacity of Connections in Composite Structures. In: International Workshop on Nature Inspired Cooperative Strategies for Optimization, NICSO 2010, pp. 359–370. Springer (2010)

[21] Filiberto, Y., Bello, R., et al.: A method to built similarity relations into extended Rough set theory. In: Proceedings of the 10th International Conference on Intelligent Systems Design and Applications, ISDA 2010, Cairo, Egipto (2010)

[22] Filiberto Cabrera, Y., Bello Pérez, R., Mota, Y.C., Jimenez, G.R.: Improving the MLP Learning by Using a Method to Calculate the Initial Weights of the Network Based on the Quality of Similarity Measure. In: Batyrshin, I., Sidorov, G. (eds.) MICAI 2011, Part II. LNCS(LNAI), vol. 7095, pp. 351–362. Springer, Heidelberg (2011)

[23] Demsar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research, 1–30 (2006)

[24] Weber, R., Foix, C.: Pronóstico del precio del cobre mediante redes neuronales. Revista Ingenieria de Sistemas XXI, 75–76 (2007)

[25] Mühlenbein, H.: The equation for the response to selection and its use for prediction. Evolutionary Computation 5(3), 303–346 (1998)

[26] Mühlenbein, H., Mahnig, T., Ochoa, A.: Schemata, distributions and graphical models on evolutionary optimization. Journal of Heuristics 5(2), 215–247 (1999)

[27] Herramienta de código abierto escrita en Java. Disponible bajo licencia pública GNU en, `http://www.cs.waikato.ac.nz/ml/weka/`

A Diversity Production Approach in Ensemble of Base Classifiers

Hamid Parvin, Sara Ansari, and Sajad Parvin

Nourabad Mamasani Branch, Islamic Azad University, Nourabad Mamasani, Iran
{hamidparvin,s.ansari}@mamasaniiau.ac.ir

Abstract. One of crucial issue in the design of combinational classifier systems is to keep diversity in the results of classifiers to reach the appropriate final result. It's obvious that the more diverse the results of the classifiers, the more suitable final result. In this paper a new approach for generating diversity during creation of an ensemble together with a new combining classifier system is proposed. The main idea in this novel system is heuristic retraining of some base classifiers. At first, a basic classifier is run, after that, regards to the drawbacks of this classifier, other base classifiers are retrained heuristically. Each of these classifiers looks at the data with its own attitude. The main attempts in the retrained classifiers are to leverage the error-prone data. The retrained classifiers usually have different votes about the sample points which are close to boundaries and may be likely erroneous. Like all ensemble learning approaches, our ensemble meta-learner approach can be developed based on any base classifiers. The main contributions are to keep some advantages of these classifiers and resolve some of their drawbacks, and consequently to enhance the performance of classification. This study investigates how by focusing on some crucial data points the performance of any base classifier can be reinforced. The paper also proves that adding the number of all "difficult" data points just as boosting method does, does not always make a better training set. Experiments show significant improvements in terms of accuracies of consensus classification. The performance of the proposed algorithm outperforms some of the best methods in the literature. Finally, the authors according to experimental results claim that forcing crucial data points to the training set as well as eliminating them from the training set can lead to the more accurate results, conditionally.

Keywords: Classifier Fusion, Heuristic Retraining, Meta-Learner.

1 Introduction

The increasing importance of recognition systems arising in a wide range of advanced applications has led to the extensive study of classification [2], [4], [14] and [17]. So, a huge amount of research has been done around of it. While most of these researches have provided good performance for specific problem, they have not enough robustness for other problems. Due to the difficulty that these researches are faced to, recent researches are directed to the combinational methods that have more potential,

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 51–60, 2013.

robustness, resistance, accuracy and generality. Although the accuracy of the classifier ensemble is not always better than the most accurate classifier in ensemble pool, its accuracy is never less than average accuracy of them [9]. Combination of multiple classifiers (CMC) can be considered as a general solution for pattern recognition problems. Inputs of CMC are result of separate classifiers and output of CMC is their final combined decisions. Roli and Kittler [19] discuss that the rationale behind the Great tendency to multiple classifier systems (MCS) is due to some serious drawbacks of classical approach in designing of a pattern recognition system which focuses on the search for the best individual classifier. Identifying the best individual classifier for the classification task without deep prior knowledge is very difficult even impossible [3]. In addition, Roli and Giacinto [9] express that it is not possible to exploit the complementary discriminatory information that other classifiers may encapsulate with only a single classifier. It is worth-noting that the motivations in favor of MCS strongly resemble those of a "hybrid" intelligent system. The obvious reason for this is that MCS can be regarded as a special-purpose hybrid intelligent system.

In General, it is an ever-true sentence that "combining the diverse classifiers any of which performs better than a random classifier, results in a better classification". Diversity is always considered as a very important issue in classifier ensemble methodology. It is also considered as the most effective factor in succeeding an ensemble. The diversity in an ensemble refers to the amount of differences in the outputs of its components (classifiers) in deciding for a given sample. Assume an example dataset with two classes. Indeed the diversity concept for an ensemble of two classifiers refers to the probability that they may produce two dissimilar results for an arbitrary input sample. The diversity concept for an ensemble of three classifiers refers to the probability that one of them produces dissimilar result from the two others for an arbitrary input sample. It is worthy to mention that the diversity can converge to 0.5 and 0.66 in the ensembles of two and three classifiers respectively. Although reaching the more diverse ensemble of classifiers is generally handful, it is harmful in boundary limit. It means that increasing diversity in an ensemble to reach 0.5 results in an unsuccessful ensemble. It is a very important dilemma in classifier ensemble field: the ensemble of accurate/diverse classifiers can be the best. Although it's mentioned before the more diverse classifiers, the better ensemble, it is provided that the classifiers are better than a random classifier.

The authors believe that Combinational methods usually enhance the quality of the result of classification, because classifiers with different features and methodologies can eliminate drawbacks of each other. Kuncheva used Condorcet theorem to show that combination of classifiers could be able to operate better than single classifier. It was illustrated that if more diverse classifiers are employed in the ensemble, then error of them can considerably be reduced. Different categorizations of combinational classifier systems are presented in [10], [19]. Valentini and Masouli divide methods of combining classifiers into two categories: generative methods, non-generative methods. In generative methods, a set of base classifiers are created by a set of base algorithms or by manipulating dataset [9]. This is done in order to reinforce diversity of base classifiers. Generally, all methods which aggregate the primary results of the fixed independent classifiers are non-generative.

Neural network ensembles as an example of combinational methods in classifiers are also becoming a hot spot in machine learning and data mining recently [18]. Many researchers have shown that simply combining the output of many neural networks can generate more accurate predictions than that of any of those individual neural networks. Theoretical and empirical works show that a good ensemble is one in that the individual base classifiers have both accuracy and diversity, i.e. the individual base classifiers make their errors on different parts of the input space [6], [8].

2 Background

In generative approaches, diversity is usually made using two groups of methods. The first group obtains diverse individuals by training classifiers on different training set, such as bagging [1], boosting [9], [21], cross validation [8] and using artificial training examples [13].

Bagging [1] is the first and easiest resampling method. This meta-learner uses bootstrap sampling to produce a number of sub-samplings by randomly drawing, with replacement, *N* data points out of the train set (that has *N* data points). The individual classifiers are often aggregated by simple majority vote mechanism.

AdaBoost [25] successively produces a sequence of classifiers, where the training data points that are incorrectly classified by prior primary classifiers are chosen more frequently than data points which were properly classified. AdaBoost tries to create new classifiers that are capable of better forecasting of the data points which are misclassified by the existing ensemble. So it attempts to minimize the average miscalculation. Arc-X4 [26] belongs to the category of Boosting ensemble approaches.

The second group employs different structures, initial weighing, parameters and base classifiers to obtain various ensemble individuals. For example, Rosen [20] adapted the training algorithm of the network by introducing a penalty term to encourage individual networks to be decorrelated. Liu and Yao [12] used negative correlation learning to generate negatively correlated individual neural network.

There is another approach that is named selective approach where the diverse components are selected from a number of trained accurate base classifiers. For example, Opitz and Shavlik [16] have proposed a generic algorithm to search for a highly diverse set of accurate base classifiers. Lazarevic and Obradoric [11] have described a pruning algorithm to eliminate redundant classifiers. Navone et al. [15] have discussed another selective algorithm based on bias/variance decomposition. GASEN by Zhou et al. [22] and PSO based approach by Fu et al. [5] also has been introduced for selection of the ensemble components.

DECORATE is a meta-learner for building diverse ensembles of classifiers by using specially constructed artificial training examples. Comprehensive experiments have demonstrated that this technique is consistently more accurate than the base classifier, Bagging and Random Forests. Decorate also obtains higher accuracy than Boosting on small training sets, and achieves comparable performance on larger training sets [23]-[24]. In the rest of this paper, a new method to obtain diverse classifiers is demonstrated which uses manipulation of dataset structures.

Inspired from boosting method, in this paper a new sort of generative approaches is presented which creates new training sets from the original one. The base classifiers are trained focusing on the crucial and error prone data of the training set. This new approach which is called "Combination of Classifiers using Heuristic Retraining, CCHR" is described in section 2 in detail. In fact, the question of "how to create a number of diverse classifiers?" is answered in section 2. Section 3 addresses the empirical studies in which we show the great accuracy and robustness of CCHR method for different datasets. Finally, section 4 discusses the concluding remarks.

3 Proposed Method

The main idea of the proposed method is heuristically retraining of any base classifier on different subsets of training data. In CCHR meta-learner, the base classifiers are trained on some possible permutations of 3 datasets named: TS, NS, and EPS. They are abbreviation for Train Set, Neighbour Set and Error-Prone Set, respectively. The set involving TS and NS results in a classification by complex boundaries with more concentration on crucial points and neighbour of errors. The set involving TS, EPS and NS results in a classification by complex boundaries with more concentration on error prone (EPS) and crucial (NS) data points. The set involving TS and NS except the data points in EPS results in a classification by simple boundaries with more concentration on crucial points. The classifier trained on a set involving both TS and EPS leads to a classification by complex boundaries with more concentration on error prone data points. Finally the set involving TS except data points in EPS results in a classification by very simple boundaries.

Designing an ensemble of classifiers trained in these different defined sets, leads to an ensemble with a high degree of diversity. In the next step, the results of all these base classifiers are combined using simple average method.

At first, a base classifier is trained on TS. Then, using the trained base classifier, the data points that may be misclassified are recognized. This work is done for different perspectives of training-test datasets. It means that it is tried to detect all error-prone data on TS. It can be implemented using either leave-one-out technique or cross-validation technique.

In cross-validation which is also called the rotation method, an integer K (preferably a factor of N) is chosen and the dataset is randomly divided into K subsets of size N/K. Then, a classifier is trained on dataset-{*i*th subset of the dataset} and evaluated using *i*th subset. This procedure is repeated K times, choosing a different part for testing each time. When N=K, the method is called the leave-one-out or U-method.

In this paper, the dataset is decomposed into three partitions: training, evaluation and test subsets. The leave-one-out technique is applied to train set for obtaining the Error-Prone Set, EPS. As it is mentioned, using leave-one-out technique a base classifier on TS-{one of its data} is trained and evaluate whether the base classifier misclassifies that eliminated data point or not. If it is misclassified we insert it into EPS. It's obvious that, we repeat this work as many as the number of data points in training set. If training dataset is huge, the cross-validation technique can be used instead of leave-one-out technique, too.

In this work, the cross-validation technique is applied to {train set + validation set} for deriving the neighbor set, NS. Since the cross-validation is an iterative technique, for each iteration K-1 subset is considered as the training set and the remaining subset as validation set. The error on each validation set is added to an error set denoted by EPS. In the next step, for each member of the error set, the nearest neighbor data point which belongs to the same label of that member is found. We insert the nearest neighbor data point to each data point in error set into a neighbor set. This neighbor set is named NS.

The EPS and NS are obtained from previous section. In this section some base classifiers are trained based on them. The more diverse and accurate base classifiers, the better results in final. So, some combinations as shown in Table 1 are used to create diversity in our ensemble. The used permutations and the reasons of their usage are shown in Table 1. Training of base classifiers, using the combinations in Table 1, results in the classifiers that each of them focuses on a special aspect of data. This can result in very good diversity in the ensemble.

Table 1. Different data combinations and reasons of their usages

Num	TS	Resultant Classifier
1	TS	Creation of base classifiers
2	TS+NS	Classification by complex boundaries with more concentration on crucial points and neighbor of errors (NS)
3	TS+EPS+NS	Classification by complex boundaries with more concentration on error prone(EPS) and crucial points(NS)
4	TS-EPS+NS	Classification by simple boundaries with more concentration on crucial points
5	TS+EPS	Classification by complex boundaries with more concentration on error prone points (EPS)
6	TS-EPS	Classification by very simple boundaries

In this paper, 6 homogeneous base classifiers are trained with use of different data according to Table 1. Their outcomes are used in CCHR ensemble. CCHR algorithm is depicted in Fig 1.

```
NS: Neighbor Set, NS={};
EPS: Error Prone Set, EPS={};
Program CCHR
        1. NS=FindNS();             //calculating NS
        2. EPS=FindEPS();           // calculating EPS
        3. Train 6 base_classifiers according to Table 1.
        4. Combine the results using         simple average.
End.
```

Fig. 1. The proposed CCHR algorithm

After creating diverse classifiers for our classifier ensemble, the next step is finding a method to fuse their results and make final decision. Final decision is made in combiner part. Based on their output, there are many different combiner methods. Some traditional models of classifier fusion based on soft/fuzzy outputs are as follows:

1. Majority vote: assume that we have k classifiers. Classifier ensemble vote to class j if a little more than half of base classifiers vote to class j.
2. Simple average: the average results of separate classifiers are calculated and then the class with the most average value is selected as final decision.
3. Weighted average: it is similar to simple average except that a weight for each classifier is used to calculate the average.

In this paper, the simple average method is used to combine their results.

4 Experimental Results

The usual metric for evaluating an output of a classifier is accuracy; so the accuracy is taken as the evaluation metric throughout all the paper for reporting performance of classifiers. In all experimental results reported for CCHR meta-learner, Multi-Layer Perceptron (MLP) is taken as base classifier.

Table 2. Brief information about the used datasets

#	*Dataset Name*	*# of Class*	*# of Features*	*# of Samples*	*Data distribution per classes*
1	Iris	3	4	150	50-50-50
2	Halfrings	2	2	400	300-100
3	Wine	3	13	178	59-71-48

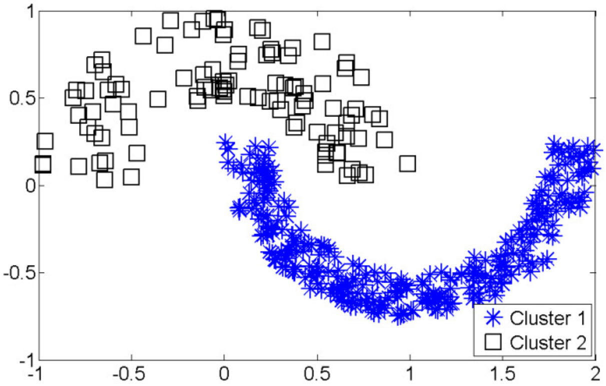


Fig. 2. Half Ring dataset

The proposed method is examined over 2 different standard datasets together with an artificial dataset. These real datasets are described in UCI repository [7]. Brief information about the used datasets is available in Table 2. The details of HalfRing dataset is available in [14]. The artificial HalfRing dataset is depicted in Fig. 2. The HalfRing dataset is considered as one of the most challenging dataset for the classification algorithms.

Table 3. Average results on Iris dataset

Train set	Classifier number as Table 1						CCHR
	1	2	3	4	5	6	
70%	95.01	95.20	95.20	94.97	**95.37**	95.07	**96.22**
50%	95.95	95.75	95.87	95.89	**96.24**	95.78	**96.64**
30%	93.57	93.26	93.17	93.64	**93.99**	93.48	**95.22**

This method is evaluated on two real standard datasets: Wine and Iris respectively in Table 3 and Table 4. All the presented results are reported over 10 independent runs. Result of each of classifiers is reported on 30%, 50%, 70% and 30%, 50% of Iris and Wine as training set, respectively. Table 3 and Table 4 show the base classifier that is trained on {TS+EPS} is relatively more robust than other base classifiers. This method is concentrated on error-prone data.

Table 4. Average results on Wine dataset

Train set	Classifier number as Table 1						CCHR
	1	2	3	4	5	6	
50%	91.58	91.64	92.66	91.98	**93.77**	91.29	**96.74**
30%	88.72	88.91	**89.31**	88.23	88.83	88.60	**93.76**

Table 5 shows the result of classification using CCHR meta-learner and two base classifiers comparatively. It is worthy to be mentioned that MLP is taken as the base classifier in CCHR meta-learner for reaching the results presented throughout all paper.

Table 5. CCHR vs. other methods

Classifier Type	Wine		Iris		
	50%	30%	70%	50%	30%
MLP	91.58	88.72	95.01	95.95	93.37
KNN	71.36	68.73	95.05	94.73	95.11
CCHR	**96.74**	**93.76**	**96.22**	**96.64**	**95.22**

As it is obvious from Table 5, recognition ratio is improved considerably by using CCHR meta-learner. Because of low number of features and records in Iris, the improvement is more significant on Wine dataset.

Table 6 shows the accuracy results of meta-learner CCHR and other traditional meta-learner methods comparatively. These results are reported averaging on the ten independent runs of the algorithms. In this comparison, the parameter K in K-Nearest Neighbor algorithm, KNN, is set to one. Also, the average accuracy of KNN method is reported over the 100 independent runs by randomly selecting a part of data as the training set, each time. To validate the meta-learner CCHR with harder benchmarks, an ensemble of the base MLPs is also implemented. These MLPs have the same structural parameters of the MLPs of meta-learner CCHR, i.e. two hidden layer with 10 and 5 neurons respectively in each of them. Like meta-learner CCHR, the voting method is chosen for combining their results.

Table 6. CCHR vs. other ensemble methods

	Wine			Iris			HalfRing		
	Train 30%	Train 50%	Train 70%	Train 30%	Train 50%	Train 70%	Train 30%	Train 50%	Train 70%
KNN	69.31	69.26	69.22	94.86	95.20	95.32	**100.00**	**100.00**	**100.00**
DT	84.80	90.26	92.34	92.03	93.47	95.23	94.49	97.42	99.41
MLP	88.72	91.58	93.09	93.37	95.95	95.01	94.27	95.26	95.05
Naïve Bayesian	**96.98**	96.42	**97.31**	95.01	95.51	95.57	94.11	94.85	94.53
Simple Ensemble	92.70	94.05	95.41	94.77	96.00	95.03	94.17	95.26	94.72
Random Forest	88.32	93.37	95.56	91.52	94.67	96.22	94.46	97.78	98.59
Decorate	96.31	95.43	95.47	93.88	95.82	**96.41**	95.57	95.71	97.27
AdaBoost	87.14	91.55	93.97	94.38	96.06	95.57	94.54	96.72	95.17
Arc-X4$_1$	96.4	96.13	96.42	94.86	96.07	95.33	97.08	98.02	98.32
Arc-X4$_2$	95.52	95.73	96.22	**95.33**	96.20	96.07	97.21	98.31	99.11
CCHR	93.76	**96.74**	96.56	95.22	**96.64**	96.22	99.19	**100.00**	**100.00**

The CCHR algorithm is compared with three state of the art meta-learners: decorate method, random forest method and boosting method. Here, the ensemble size of the random forest is 21. The ensemble size for Arc-X4$_1$ is 5 classifiers. While the ensemble size for Arc-X4$_2$ is 11 classifiers. The Simple Ensemble method is an ensemble of MLPs. The ensemble size in Decorate is 5. The classifiers used in Decorate are 5 MLPs. AdaBoost uses 11 MLPs with the same configuration of those used for CCHR meta-learner.

5 Conclusion and Future Work

In this paper, a new method for improving performance of multiple classifier systems named Combination of Classifiers using Heuristic Retraining, CCHR, is proposed. CCHR is based on heuristic retraining of base classifiers on different subsets of training data. Also, it is observed that different datasets result in different classifiers.

It is illustrated that the classifiers with complex boundaries and also those concentrating on error-prone data points have more generalization than others. It is shown that emphasizing on crucial data points causes improvement in results. With regard to the obtained results, we can conclude that the base classifier that is trained on {TS+EPS} is relatively more robust than other base classifiers. We can consider this base classifier as the best way for heuristically retraining of a base classifier. Also we have described that usage of different subsets of training set causes to a quite diverse ensemble.

Another interesting conclusion of the paper is the fact that emphasizing on the boundary data points, as boosting algorithm is not always very good. Although, boosting of the boundary data points in many cases is good, there are some problems where elimination of such data points is better. The Monk's problem is one of such cases which deleting error-prone data leads to a better result. Also, in data mining tasks which deal with huge data, the small size of ensemble is very interesting which is satisfied in CCHR as well. While CCHR only uses MLP as base classifier, it can use any other base classifier without losing its generality. So we can consider it as a meta-learner. For future work other base classifier can be used in CCHR meta-learner.

References

1. Breiman, L.: Bagging predictors. Machine Learning 24(2), 123–140 (1996)
2. Parvin, H., Minaei-Bidgoli, B., Ghatei, S., Alinejad-Rokny, H.: An Innovative Combination of Particle Swarm Optimization, Learning Automaton and Great Deluge Algorithms for Dynamic Environments. International Journal of the Physical Sciences 6(22), 5121–5127 (2011)
3. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. John Wiley & Sons, NY (2001)
4. Parvin, H., Minaei-Bidgoli, B., Alizadeh, H.: A New Clustering Algorithm with the Convergence Proof. In: König, A., Dengel, A., Hinkelmann, K., Kise, K., Howlett, R.J., Jain, L.C. (eds.) KES 2011, Part I. LNCS, vol. 6881, pp. 21–31. Springer, Heidelberg (2011)
5. Fu, Q., Hu, S.X., Zhao, S.Y.: A PSO-based approach for neural network ensemble. Journal of Zhejiang University (Engineering Science) 38(12), 1596–1600 (2004) (in Chinese)
6. Hansen, L.K., Salamon, P.: Neural network ensembles. IEEE Transaction on Pattern Analysis and Machine Intelligence 12(10), 993–1001 (1990)
7. Blake, C.L., Merz, C.J.: UCI Repository of machine learning databases (1998), http://www.ics.uci.edu/mlearn/MLRepository.html
8. Krogh, A., Vedelsdy, J.: Neural Network Ensembles Cross Validation, and Active Learning. Advances in Neural Information Processing Systems 7, 231–238 (1995)
9. Kuncheva, L.I.: Combining Pattern Classifiers, Methods and Algorithms. Wiley, New York (2005)
10. Lam, L.: Classifier Combinations: Implementations and Theoretical Issues. In: Kittler, J., Roli, F. (eds.) MCS 2000. LNCS, vol. 1857, pp. 77–86. Springer, Heidelberg (2000)
11. Lazarevic, A., Obradovic, Z.: Effective pruning of neural network classifier ensembles. In: Proc. International Joint Conference on Neural Networks, vol. 2, pp. 796–801 (2001)
12. Liu, Y., Yao, X.: Evolutionary ensembles with negative correlation learning. IEEE Trans. Evolutionary Computation 4(4), 380–387 (2000)

13. Melville, P., Mooney, R.: Constructing Diverse Classifier Ensembles Using Artificial Training Examples. In: Proc. of the IJCAI 2003, pp. 505–510 (2003)
14. Minaei-Bidgoli, B., Parvin, H., Alinejad-Rokny, H., Alizadeh, H., Punch, W.F.: Effects of resampling method and adaptation on clustering ensemble efficacy (2011)
15. Navone, H.D., Verdes, P.F., Granitto, P.M., Ceccatto, H.A.: Selecting Diverse Members of Neural Network Ensembles. In: Proc. 16th Brazilian Symposium on Neural Networks, pp. 255–260 (2000)
16. Opitz, D., Shavlik, J.: Actively searching for an effective neural network ensemble. Connection Science 8(3-4), 337–353 (1996)
17. Parvin, H., Helmi, H., Minaei-Bidgoli, B., Alinejad-Rokny, H., Shirgahi, H.: Linkage Learning Based on Differences in Local Optimums of Building Blocks with One Optima. International Journal of the Physical Sciences 6(14), 3419–3425 (2011)
18. Qiang, F., Shang-Xu, H., Sheng-Ying, Z.: Clustering-based selective neural network ensemble. Journal of Zhejiang University Science 6A(5), 387–392 (2005)
19. Roli, F., Kittler, J. (eds.): MCS 2002. LNCS, vol. 2364. Springer, Heidelberg (2002)
20. Rosen, B.E.: Ensemble learning using decorrelated neural network. Connection Science 8(3-4), 373–384 (1996)
21. Schapire, R.E.: The strength of weak learn ability. Machine Learning 5(2), 197–227 (1990)
22. Zhou, Z.H., Wu, J.X., Jiang, Y., Chen, S.F.: Genetic algorithm based selective neural network ensemble. In: Proc. 17th International Joint Conference on Artificial Intelligence, vol. 2, pp. 797–802 (2001)
23. Melville, P., Mooney, R.J.: Constructing Diverse Classifier Ensembles Using Artificial Training Examples. In: Conference on Artificial Intelligence, pp. 505–510 (2003)
24. Melville, P., Mooney, R.J.: Creating Diversity in Ensembles Using Artificial Data. Information Fusion: Special Issue on Diversity in Multiclassifier Systems 6(1), 99–111 (2004)
25. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. In: European Conference on Computational Learning Theory, pp. 23–37 (1995)
26. Breiman, L.: Arcing classifiers. The Annals of Statistics 26(3), 801–823 (1998)

A New Overlapping Clustering Algorithm Based on Graph Theory

Airel Pérez-Suárez[1,2], José Fco. Martínez-Trinidad[1],
Jesús A. Carrasco-Ochoa[1], and José E. Medina-Pagola[2]

[1] Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)
Luis E. Erro No. 1, Sta. María Tonantzintla, Puebla, CP:72840, Mexico
[2] Advanced Technologies Application Center (CENATAV)
7ma A ♯ 21406 e/ 214 and 216, Playa, C.P. 12200, La Habana, Cuba
{airel,ariel,fmartine}@ccc.inaoep.mx, jmedina@cenatav.co.cu

Abstract. Most of the clustering algorithms reported in the literature build disjoint clusters; however, there are several applications where overlapping clustering is useful and important. Although several overlapping clustering algorithms have been proposed, most of them have a high computational complexity or they have some limitations which reduce their usefulness in real problems. In this paper, we introduce a new overlapping clustering algorithm, which solves the limitations of previous algorithms, while it has an acceptable computational complexity. The experimentation, conducted over several standard collections, demonstrates the good performance of the proposed algorithm.

Keywords: Data Mining, Overlapping Clustering, Graph-based Algorithms.

1 Introduction

Clustering is a Data Mining technique that organizes a set of objects into a set of classes called *clusters*, such that objects belonging to the same cluster are enough similar to infer they are of the same type, and objects belonging to different clusters are enough different to infer they are of different types [1].

Most of the clustering algorithms reported in the literature build disjoint clusters; *i.e.*, they do not allow objects to belong to more than one cluster. Although this approach has been successful for unsupervised learning, there are some applications, like information retrieval [2], social network analysis [16, 19], Bioinformatics [14] and news stream analysis [4], among others, where objects could belong to more than one cluster. For this kind of applications, overlapping clustering is useful and important.

Several overlapping clustering algorithms have been proposed in the literature [2–21]; however, most of them have a high computational complexity or they have some limitations which reduce their usefulness in real problems. This is the main reason why we address the problem of overlapping clustering in this work.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 61–72, 2013.

In this paper, we introduce a new clustering algorithm for building overlapping clusterings. The proposed algorithm, called OCDC (Overlapping Clustering based on Density and Compactness), introduces a new graph-covering strategy and a new filtering strategy, which together allow obtaining a small set of overlapping clusters. Moreover, our algorithm reduces the limitations of previous algorithms and it has an acceptable computational complexity. An experimental evaluation, over several standard collections, showed that OCDC builds more accurate clusters, according to the FBcubed evaluation measure [22], than those built by previous overlapping clustering algorithms.

The remainder of this paper is as follows: in Section 2, the related work is analyzed. In Section 3, we introduce the OCDC algorithm. An experimental evaluation, for showing the performance of our proposed algorithm over several data collections, is presented in Section 4. Finally, the conclusions and future work are presented in Section 5.

2 Related Work

In the literature, there have been proposed several clustering algorithms that address the problem of overlapping clustering [2–21]. The DHS algorithm, reported in [23], was not included as related work because it addresses the problem of overlapping hierarchical clustering, which is out of the scope of this paper.

Star [2], ISC [4], Estar [5, 6], Gstar [11], ACONS [12], ICSD [17] and SimClus [21] are *graph-based* clustering algorithms, which build overlapping clusterings; the SimClus algorithm uses the same strategy and criterion for building the clustering as the one used in Estar algorithm, proposed by Gil-García *et al.* in [6]. The computational complexity of Star and ICSD is $O(n^2 \cdot \log^2 n)$ and $O(n^2)$, respectively. On the other hand, the computational complexity of Estar, Gstar, and ACONS is $O(n^3)$.

These algorithms have two main limitations. First of all, as we will show in our experiments, they build a large number of clusters. It is known that the correct number of clusters is unknown in real problems. However, when we use a clustering algorithm for discovering hidden relations among objects of a collection, the number of clusters obtained should be small wrt. the number of objects in the collection. Note that, if the number of clusters grows, then analyzing those clusters could be as difficult as analyzing the whole collection.

Another limitation, as we will show in our experiments, is that they build clusterings with high overlapping. As it was mentioned in the previous section, overlapping clustering is very useful for several applications; however, when the overlapping is high, it could be difficult to obtain something useful about the structure of the data. Moreover, there are applications like, for instance, document segmentation by topic, where a high overlapping could be a signal of a bad segmentation [24]. Another example can be found in social networks analysis [13, 15].

Another algorithm able to build overlapping clusters is STC [3]. As it was mentioned in [7], the main limitation of STC is related to its computational complexity, which could get to be exponential. Besides, STC was specifically designed to work with text strings which limits its applicability.

Other algorithms addressing the problem of overlapping clustering are SHC [7], LA-IS^2 [9], RaRe-IS [10], the algorithm proposed by Zhang *et al.* in [14], RRW [18], LA-CIS [19] and SSDE-Cluster [20]. The computational complexity of SHC, RaRe-IS, LA-IS^2, LA-CIS and RRW is $O(n^2)$. On the other hand, the computational complexity of SSDE-Cluster and the algorithm proposed by Zhang *et al.* is $O(n \cdot k \cdot d)$ and $O(n^2 \cdot k \cdot h)$, respectively.

From the user point of view, these algorithms have several limitations. First of all, SHC, RaRe-IS, RRW, SSDE-Cluster and the algorithm proposed by Zhang *et al.* need to tune values of several parameters, whose values depend on the collection to be clustered. In general, users do not have any a priori knowledge about the collection they want to cluster; therefore, to tune up several parameters could be a difficult task. Second, SHC, RaRe-IS and LA-CIS build clusterings with high overlapping. Finally, LA-IS^2 and LA-CIS make irrevocable assignment of objects to clusters; this is, once an object has been assigned to a cluster, it cannot be removed and added to another cluster even if doing it would improve the clustering quality.

Other overlapping clustering algorithms reported in the literature are the algorithm proposed by Palla *et al.* [8], CONGA [13], CONGO [15] and H-FOG [16]. The computational complexity of CONGA, CONGO and H-FOG is $O(n^6)$, $O(n^4)$ and $O(n^6)$, respectively. The computational complexity of the algorithm proposed by Palla *et al.* is exponential. The main limitation of these algorithms is their computational complexity, which makes them inapplicable in real problems involving large collections. Besides, CONGA, CONGO and H-FOG need to know in advance the number of clusters to build; this number is commonly unknown in real problems.

As it can be seen from the related work, in the literature there are many algorithms addressing the problem of overlapping clustering. However, as we pointed out in this section, they have some limitations. These limitations are mainly related to:

a) the production of a large number of clusters.
b) the production of clusterings with high overlapping.
c) the necessity of tuning several parameters whose values depend on the collection to be clustered.

Besides, there are several algorithms having a high computational complexity; these limitations make these algorithms worthless for many real problems.

The algorithm proposed in this work solves the aforementioned limitations, while it has an acceptable computational complexity. Moreover, as we will show in our experiments, our algorithm builds more accurate clusterings than those built by previous algorithms.

3 Building Overlapping Clusterings

In this section, a new overlapping clustering algorithm is introduced. Our algorithm, called OCDC (Overlapping Clustering based on Density and Compactness), represents a collection of objects as a *weighted thresholded similarity graph* $\widetilde{G}_\beta$ and it builds an overlapping clustering in two phases called *initialization* and *improvement*. In the initialization phase, a set of initial clusters is built through a vertex covering of $\widetilde{G}_\beta$. In the improvement phase, the initial clusters are processed for reducing both the number of clusters and their overlapping.

The presentation of the OCDC algorithm is divided into three parts. First of all, in subsection 3.1, we give some basic concepts needed for introducing our algorithm. After, in subsection 3.2, we explain the initialization phase and finally, in subsection 3.3, we explain the improvement phase.

3.1 Basic Concepts

Let $O = \{o_1, o_2, \ldots, o_n\}$ be a collection of objects, $\beta \in [0, 1]$ a given parameter and $S(o_i, o_j)$ a symmetric similarity function.

A *weighted thresholded similarity graph* is an undirected and weighted graph $\widetilde{G}_\beta = \langle V, \widetilde{E}_\beta, S \rangle$, such that $V = O$ and there is an edge $(v, u) \in \widetilde{E}_\beta$ iff $v \neq u$ and $S(v, u) \geq \beta$; each edge $(v, u) \in \widetilde{E}_\beta$ is labeled with the value of $S(v, u)$.

Let $\widetilde{G}_\beta = \langle V, \widetilde{E}_\beta, S \rangle$ be a weighted thresholded similarity graph. A *weighted star-shaped sub-graph* (*ws-graph*) in $\widetilde{G}_\beta$, denoted by $G^\star = \langle V^\star, E^\star, S \rangle$, is a subgraph of $\widetilde{G}_\beta$, having a vertex $c \in V^\star$, such that there is an edge between c and all the vertices in $V^\star \setminus \{c\}$. The vertex c is called the *center* of $G^\star$ and the other vertices of $V^\star$ are called *satellites*. All vertices having no adjacent vertices (*i.e.*, isolated vertices) are considered *degenerated* ws-graphs.

Let $\widetilde{G}_\beta = \langle V, \widetilde{E}_\beta, S \rangle$ be a weighted thresholded similarity graph and $W = \{G^\star_1, G^\star_2, \ldots, G^\star_k\}$ be a set of ws-graphs in $\widetilde{G}_\beta$, such that each $G^\star_i = \langle V^\star_i, E^\star_i, S \rangle, i = 1 \ldots k$. The set W is a *cover* of $\widetilde{G}_\beta$ iff it meets that $V = \bigcup_{i=1}^{k} V^\star_i$. In addition, we will say that a ws-graph $G^\star_i$ *covers* a vertex v iff $v \in V^\star_i$.

Let $G^\star_v = \langle V^\star_v, E^\star_v, S \rangle$ be a non degenerated ws-graph, having v as its *center*. The intra-cluster similarity of $G^\star_v$, denoted as $Intra\_sim(G^\star_v)$, is the average similarity between all pair of vertices belonging to $V^\star$ [25], and it is computed as:

$$Intra\_sim(G^\star_v) = \frac{\sum_{z,u \in V^\star_v, z \neq u} S(z, u)}{\frac{|V^\star_v| \cdot (|V^\star_v| - 1)}{2}}, \text{where } |V^\star_v| \neq 1 \quad (1)$$

Computing $Intra\_sim(G^\star_v)$ using (1) is $O(n^2)$. Thus, using this equation for computing the intra-cluster similarity of the ws-graph determined by each vertex in $\widetilde{G}_\beta$ becomes $O(n^3)$. Since we want to reduce the computational complexity of OCDC, we will approximate the intra-cluster similarity of a non degenerated ws-graph $G^\star_v$ by the average similarity between its center and its satellites, as follows:

$$Aprox\_Intra\_sim(G_v^\star) = \frac{\sum_{u \in V_v^\star, u \neq v} S(v,u)}{|V_v^\star| - 1}, \text{where } |V_v^\star| \neq 1 \tag{2}$$

Equation (2) is $O(n)$; thus, while we are building $\widetilde{G}_\beta$, the *Aprox_Intra_sim* of each ws-graph in $\widetilde{G}_\beta$ can be computed, without extra computational cost.

3.2 Initialization Phase

The main idea of this phase is to generate an initial set of clusters by covering $\widetilde{G}_\beta$ using ws-graphs; in this context, each ws-graph constitutes an initial cluster.

As it can be seen from subsection 3.1, a ws-graph is determined by its center; thus, the problem of building a set $W = \{G_{c_1}^\star, G_{c_2}^\star, \ldots, G_{c_k}^\star\}$ of ws-graphs, such that W is a cover of $\widetilde{G}_\beta$, can be transformed into the problem of finding a set $X = \{c_1, c_2, \ldots, c_k\}$ of vertices, such that $c_i \in X$ is the center of $G_{c_i}^\star \in W$, $\forall i = 1 \ldots k$. Each vertex of $\widetilde{G}_\beta$ forms a ws-graph; therefore, all vertices of $\widetilde{G}_\beta$ should be analyzed for building X. In order to prune the search space and for establishing a criterion for selecting those vertices that should be added to X, we introduce the concepts of *density* and *compactness* of a vertex v.

An idea for reducing the number of ws-graphs needed for covering $\widetilde{G}_\beta$, would be to iteratively add the vertex having the highest degree to X. Thus, we expect to maximize the number of vertices that are iteratively added to the cover of $\widetilde{G}_\beta$; nevertheless, the number of vertices that a vertex v could add to the cover of $\widetilde{G}_\beta$ depends on the previously selected vertices. Let $v_1, v_2, \ldots, v_k$ be those vertices which were selected in the first k iterations. Let u be a non previously selected vertex, having the highest degree in the iteration $k+1$. If d previously selected vertices ($d \leq k$) are adjacent to u, then u would add to the cover of $\widetilde{G}_\beta$ only $|u.Adj| - d$ vertices. However, if there is a vertex z in the same iteration, such that:

1. $|z.Adj| < |u.Adj|$.
2. u is adjacent to d_1 previously selected vertices ($d_1 \leq k$).
3. $|u.Adj| - d < |z.Adj| - d_1$.

then selecting the vertex z in the iteration $k+1$ would be a better choice than selecting u.

From the previous paragraph, we can conclude that if $v \in V$ is added to X, the number of adjacent vertices of v, having a degree non greater than the degree of v, is a good estimation about how much vertices could include v in the cover of $\widetilde{G}_\beta$. However, this number is bounded by the total number of adjacent vertices of v; *i.e.*, there could be a bias for vertices having a lot of adjacent vertices. Let v, u be two vertices, such that v and u could add 6 and 4 vertices to the cover of $\widetilde{G}_\beta$, respectively. Based on the previous paragraph, the best choice is adding v to X. But, if we know that $|v.Adj| = 10$ and $|u.Adj| = 5$, then u could be a better choice, because it could include in the cover of $\widetilde{G}_\beta$ a greater percent of its adjacent vertices than v (4/5 versus 6/10). Motivated by these ideas, we introduce the concept of *density* of a vertex.

The *density* of a vertex $v \in V$, denoted as $v.density$, is computed as follows:

$$v.density = \frac{v.pre\_dens}{|v.Adj|} \tag{3}$$

where $v.pre\_dens$ denotes the number of adjacent vertices of v, having a degree non greater than the degree of v. Equation (3) takes values in [0,1]. The higher the value of $v.density$, the greater the number of adjacent vertices that v could include in the cover of $\widetilde{G}_\beta$ and therefore, the better v is for covering $\widetilde{G}_\beta$.

There are some aspects we would like to highlight from the previous definition. First, a high value of $v.density$ also means that the ws-graph determined by v has low overlapping with the previously selected ws-graphs. Second, since this property is based on the degree of vertices, using it as the only clustering criterion could lead to select ws-graphs with a high number of satellites but with low average similarity among them. For solving this issue, we define the concept of *compactness* of a vertex, which takes into account the $Aprox\_Intra\_sim$ of a ws-graph, computed using equation (2).

The *compactness* of a vertex $v \in V$, denoted as $v.compactness$, is computed as follows:

$$v.compactness = \frac{v.pre\_compt}{|v.Adj|} \tag{4}$$

where $v.pre\_compt$ denotes the number of vertices $u \in v.Adj$, such that $Aprox\_Intra\_sim(G_v^\star) \geq Aprox\_Intra\_sim(G_u^\star)$, being $G_v^\star$ and $G_u^\star$ the ws-graphs determined by v and u, respectively. Like (3), equation (4) takes values in [0,1]. The higher the value of $v.compactness$, the greater the number of adjacent vertices that v could include in the cover of $\widetilde{G}_\beta$ and therefore, the better v is for covering the graph.

Finally, both aforementioned criteria are combined through the average between $v.density$ and $v.compactness$. A high average value corresponds with a high value of $v.density$ and/or $v.compactness$; thus, the higher this average, the better v is for covering $\widetilde{G}_\beta$. Therefore, in order to build the covering of $\widetilde{G}_\beta$, we must analyze the vertices in descending order, according to the average of $v.density$ and $v.compactness$. Moreover, we do not have to analyze those vertices having $v.density = 0$ and $v.compactness = 0$.

The strategy proposed for building a covering of $\widetilde{G}_\beta$ is comprised of three steps. First, all vertices having an average of density and compactness greater than zero are added to a *list of candidates* L; isolated vertices are included directly in X. After, the list L is sorted in descending order, according to the average of density and compactness of the vertices. Finally, L is iteratively processed and each vertex $v \in L$ is added to X if it satisfies at least one of the following conditions:

a) v is not covered yet.
b) v is already covered but it has at least one adjacent vertex which is not covered yet. This condition avoids the selection of ws-graphs having all their satellites covered by previously selected ws-graphs.

After this process, each selected ws-graph constitutes an initial cluster.

3.3 Improvement Phase

The main idea of this phase is to postprocess the initial clusters in order to reduce the number of clusters and their overlapping. For doing this, the set X is analyzed in order to remove those less *useful* ws-graphs. In this context, the usefulness of a ws-graph $G^\star$ will be determined based on its number of satellites and the number of satellites it shares with other ws-graphs.

The strategy proposed for building X is *greedy*; thus, it could happen that, once the set X has been built, some vertices in X could be removed and $\widetilde{G}_\beta$ would remain as covered. For example, if there is a vertex $v \in X$ such that:

(i) the ws-graph $G^\star_v$ determined by v shares all their satellites with other selected ws-graphs.
(ii) v itself belongs to at least another selected ws-graph.

then we can remove v from X and $\widetilde{G}_\beta$ would remain as covered. Nevertheless, if there is a vertex $u \in X$ such that u meets condition (ii) but it does not meet condition (i), then u cannot be removed from X. Notice that, even though most of the satellites of $G^\star_u$ were covered by other ws-graphs, removing u from X will leave the non-shared satellites of $G^\star_u$ as uncovered. For solving this issue, we will define the usefulness of a ws-graph based on how many satellites it shares.

Let $v \in X$ be a vertex determining the ws-graph $G^\star_v$ in the covering of $\widetilde{G}_\beta$. Let $v.Shared$ be the set of satellites that $G^\star_v$ shares with other selected ws-graphs and let $v.Non\_shared$ be the set of satellites belonging only to $G^\star_v$. In this work, we will understand that $G^\star_v$ is not useful for covering $\widetilde{G}_\beta$ iff the following conditions are meet:

(1) there is at least another selected ws-graph covering v.
(2) $|v.Shared| > |v.Non\_shared|$.

For removing a non-useful ws-graph, we must add all its non-shared satellites to other initial clusters. Since a non-useful ws-graph $G^\star_v$ meets condition (1), the non-shared satellites of $G^\star_v$ will be added to the ws-graph having the greatest number of satellites, among all ws-graphs covering vertex v; thus, we allow the creation of clusters with many objects. In this context, adding non-shared satellites of $G^\star_v$ to another ws-graph $G^\star_u$ means adding those satellites to a list named *u.Linked*; thus, the cluster determined by $G^\star_u$ now will include also the vertices in *u.Linked*.

The strategy proposed for removing the non-useful ws-graphs and building the final clustering consists of two steps. First, the vertices in X are marked as *not-analyzed* and they are sorted in descending order, according to their degree. In the second step, each vertex $v \in X$ is visited for removing from $v.Adj$ all the vertices forming non-useful ws-graphs. For this purpose, each vertex $u \in v.Adj$ is analyzed as follows: If u belongs to X and it is marked as *not-analyzed* we check if $G^\star_u$ is a non-useful ws-graph; the vertices belonging to X and marked as *analyzed* are those which were determined as useful in a previous iteration and therefore, they do not need to be verified again. For checking if $G^\star_u$ is a non-useful ws-graph we only need to check if $G_u^\star$ meets the above mentioned condition (2). Notice that, since u belongs to the ws-graph formed by v, $G_u^\star$ already meets condition (1). After, if $G_u^\star$ is non-useful, then u is removed from X and its non-shared satellites are added to $v.Linked$; otherwise, u is marked as *analyzed*. Once all the vertices of $v.Adj$ have been analyzed, the set determined by vertex v, together with the vertices in $v.Adj$ and $v.Linked$, constitutes a cluster in the final clustering. We would like to mention that, the way in which a cluster is built allows us to understand its center v as the *prototype* of the cluster. Noticed that, each vertex included in this cluster is related with v; that is, it is adjacent to v or it is adjacent to one adjacent vertex of v.

The OCDC algorithm has a computational complexity of $O(n^2)$; the proof was omitted due to space restrictions.

4 Experimental Results

In this section, the results of some experiments that show the performance of OCDC algorithm are presented. In these experiments, we contrasted our results against those obtained by the algorithms Star [2], ISC [4], SHC [7], Estar [5], Gstar [11], ACONS [12] and ICSD [17].

The collections used in the experiments were built from five benchmark text collections, commonly used in document clustering: AFP[1], CISI[2], CACM[2], Reuters-21578[3] and TDT2[4]. From these benchmarks, six document collections were built; the characteristics of these collections are shown in Table 1.

Table 1. Overview of collections

Name	#Documents	#Terms	#Classes	Overlapping
AFP	695	11 785	25	1.023
Reu-Te	3587	15 113	100	1.295
Reu-Tr	7780	21 901	115	1.241
TDT	16 006	68 019	193	1.188
cacm	433	3038	52	1.499
cisi	1162	6976	76	2.680

In our experiments, documents were represented using the Vector Space Model. The index terms of the documents represent the lemmas of the words occurring at least once in the whole collection; stop words were removed. The index terms of each document were statistically weighted using term frequency normalized by the *maximum term frequency*. The cosine measure was used to calculate the similarity between two documents.

[1] `http://trec.nist.gov`

[2] `ftp://ftp.cs.cornell.edu/pub/smart`

[3] `http://kdd.ics.uci.edu`

[4] `http://www.nist.gov/speech/tests/tdt.html`

There are three types of evaluation measures: *external*, *relative* and *internal* measures [1]. From these three kind of measures, the most widely used are the external measures. Several external evaluation measures have been proposed in the literature, for instance: Purity and Inverse Purity, F1-measure, Jaccard coefficient, Entropy, Class Entropy and V-measure, among others (see [22, 26]). However, none of the external measures reported so far have been developed, at least explicitly, for evaluating overlapping clustering; that is, these measures fail at reflecting the fact that, in a perfect overlapping clustering, objects sharing n classes should share n clusters.

In [22], Amigo *et al.* proposed a new external evaluation measure, called FBcubed, for evaluating overlapping clusterings. This measure meets four constraints which evaluate several desirable characteristics in an overlapping clustering solution. These constraints are intuitive and they express important characteristics that should be evaluated by an external evaluation measure. Moreover, in [22] the authors showed that none of the most used external evaluation measures satisfies all the four constraints. Based on the aforementioned analysis and in order to conduct a fair comparison among the algorithms, we will use the FBcubed measure for evaluating the quality of the clusterings built by each algorithm. A more detailed explanation about the FBcubed measure can be found in [22].

The first experiment was focused on comparing the algorithms, according to the quality of the clusterings they build for each collection; for evaluating a clustering solution, we used the aforementioned FBcubed measure [22]. In Table 2, we show the best performances attained by each algorithm over each collection, according to the FBcubed measure.

Table 2. Best performances of each algorithm over each collection. The highest values per collections appear bold-faced.

Collection	Star	ISC	Estar	Gstar	ACONS	ICSD	SHC	OCDC
AFP	0.69	0.20	0.63	0.63	0.62	0.61	0.27	**0.77**
Reu-Te	0.45	0.05	0.39	0.40	0.40	0.39	0.20	**0.51**
Reu-Tr	0.42	0.03	0.36	0.36	0.36	0.36	0.19	**0.43**
TDT	0.43	0.06	0.37	0.35	0.34	0.35	0.15	**0.48**
cacm	0.31	0.18	0.32	0.31	0.32	0.32	0.15	**0.33**
cisi	0.30	0.05	0.29	0.29	0.29	0.29	0.21	**0.32**

As it can be seen from Table 2, OCDC outperforms, according to the FBcubed measure, all the other tested algorithms.

The second experiment was focused on comparing the algorithms according to the number of clusters they build for each collection. Table 3 shows the results of this experiment.

As it can be noticed from Table 3, OCDC builds, in almost all collections, less clusters than the other algorithms. We would like to highlight that these results of OCDC are obtained without affecting the quality of the clusterings

Table 3. Number of clusters built by each algorithm for each collection. The smallest values per collection appear bold-faced.

Collection	Star	ISC	Estar	Gstar	ACONS	ICSD	SHC	OCDC
AFP	123	334	98	90	129	104	85	**52**
Reu-Te	507	1785	600	711	798	621	273	**102**
Reu-Tr	471	3936	904	849	857	853	561	**166**
TDT	2019	8250	1854	1653	1663	1657	1203	**769**
cacm	124	228	122	129	129	152	**29**	102
cisi	134	654	195	159	213	209	100	**52**

(see Table 2); in this way, OCDC builds clusterings that could be easier to analyze than those built by the other algorithms used in the comparison.

The third experiment was focused on comparing the algorithms according to the overlapping they produce for each collection; the overlapping of a clustering is computed as the average number of clusters in which an object is included [23]. In Table 4, we show the overlapping of the clusterings built by each algorithm for each collection.

Table 4. Overlapping of the clustering built by each algorithm for each collection. The lowest values per collection appear bold-faced.

Collection	Star	ISC	Estar	Gstar	ACONS	ICSD	SHC	OCDC
AFP	1.71	1.65	2.52	2.31	2.48	2.53	2.43	**1.18**
Reu-Te	3.41	1.79	7.40	6.73	6.66	7.64	13.13	**1.40**
Reu-Tr	5.54	1.84	12.14	13.08	12.65	13.32	29.33	**1.56**
TDT	4.81	1.88	59.41	69.43	66.38	70.97	80.74	**1.50**
cacm	2.31	1.82	3.46	3.20	3.19	2.72	1.99	**1.26**
cisi	4.12	2.12	7.85	7.49	7.77	7.54	6.98	**1.58**

From Table 4, it can be seen that OCDC builds clusterings with less overlapping than the clusterings built by the other algorithms. This way, OCDC allows overlapping among the clusters but it controls the overlapping, in order to avoid building clusters with high overlapping, that could be interpreted as the same cluster. This characteristic could be useful for applications where low overlapping is desirable, for instance, document segmentation by topic [24] or web document organization [3, 7].

5 Conclusions

In this paper, we introduced OCDC, a new overlapping clustering algorithm. OCDC introduces a new graph-covering strategy and a new filtering strategy, which together allow obtaining a small set of overlapping clusters with low overlapping. Moreover, OCDC reduces the limitations of previous algorithms and it has an acceptable computational complexity.

The OCDC algorithm was compared against other overlapping clustering algorithms of the state-of-the-art in terms of: i) clustering quality, ii) number of clusters, and iii) overlapping of the clusters. The experimental evaluation showed that OCDC builds more accurate clusterings, according to the FBcubed measure, than the algorithms used in the comparison. Moreover, OCDC builds clusterings having less clusters and less overlapping than those clusterings built by most of the tested algorithms. From the above commented, we can conclude that OCDC is a better option for overlapping clustering than previously reported algorithms.

As future work, we are going to explore the use of the OCDC algorithm in agglomerative hierarchical overlapping clustering.

Acknowledgment. This work was supported by the National Council on Science and Technology of Mexico (CONACYT) under Grants 106366 and 106443.

References

1. Pfitzner, D., Leibbrandt, R., Powers, D.: Characterization and evaluation of similarity measures for pairs of clusterings. Knowledge and Information Systems 19(3), 361–394 (2009)
2. Aslam, J., Pelekhov, K., Rus, D.: Static and Dynamic Information Organization with Star Clusters. In: Proceedings of the Seventh International Conference on Information and Knowledge Management, pp. 208–217 (1998)
3. Zamir, O., Etziony, O.: Web document clustering: A feasibility demonstration. In: Proceedings of the 21st ACM SIGIR Conference, pp. 46–54 (1998)
4. Pons-Porrata, A., Ruiz-Shulcloper, J., Berlanga-Llavorí, R., Santiesteban-Alganza, Y.: Un algoritmo incremental para la obtención de cubrimientos con datos mezclados. In: Proceedings of CIARP 2002, pp. 405–416 (2002)
5. Gil-García, R.J., Badía-Contelles, J.M., Pons-Porrata, A.: Extended Star Clustering Algorithm. In: Sanfeliu, A., Ruiz-Shulcloper, J. (eds.) CIARP 2003. LNCS, vol. 2905, pp. 480–487. Springer, Heidelberg (2003)
6. Gil-García, R.J., Badía-Contelles, J.M., Pons-Porrata, A.: Parallel Algorithm for Extended Star Clustering. In: Sanfeliu, A., Martínez Trinidad, J.F., Carrasco Ochoa, J.A. (eds.) CIARP 2004. LNCS, vol. 3287, pp. 402–409. Springer, Heidelberg (2004)
7. Hammouda, K.M., Kamel, M.S.: Efficient Phrase-Based Document Indexing for Web Document Clustering. IEEE Transactions on Knowledge and Data Engineering 16(10), 1279–1296 (2004)
8. Palla, G., Derényi, I., Farkas, I., Vicsek, T.: Uncovering the overlapping community structure of complex networks in nature and society. Nature 435(7043), 814–824 (2005)
9. Baumes, J., Goldberg, M., Magdon-Ismail, M.: Efficient Identification of Overlapping Communities. In: Kantor, P., Muresan, G., Roberts, F., Zeng, D.D., Wang, F.-Y., Chen, H., Merkle, R.C. (eds.) ISI 2005. LNCS, vol. 3495, pp. 27–36. Springer, Heidelberg (2005)
10. Baumes, J., Goldberg, M., Krishnamoorty, M., Magdon-Ismail, M., Preston, N.: Finding communities by clustering a graph into overlapping subgraphs. In: Proceedings of IADIS Applied Computing, pp. 97–104 (2005)

11. Suárez, A.P., Pagola, J.E.M.: A Clustering Algorithm Based on Generalized Stars. In: Perner, P. (ed.) MLDM 2007. LNCS (LNAI), vol. 4571, pp. 248–262. Springer, Heidelberg (2007)
12. Alonso, A.G., Suárez, A.P., Pagola, J.E.M.: ACONS: A New Algorithm for Clustering Documents. In: Rueda, L., Mery, D., Kittler, J. (eds.) CIARP 2007. LNCS, vol. 4756, pp. 664–673. Springer, Heidelberg (2007)
13. Gregory, S.: An Algorithm to Find Overlapping Community Structure in Networks. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) PKDD 2007. LNCS (LNAI), vol. 4702, pp. 91–102. Springer, Heidelberg (2007)
14. Zhang, S., Wang, R.S., Zhang, X.S.: Identification of overlapping community structure in complex networks using fuzzy c-means clustering. Physica A: Statistical Mechanics and its Applications, Vol 374(1), 483–490 (2007)
15. Gregory, S.: A Fast Algorithm to Find Overlapping Communities in Networks. In: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008, Part I. LNCS (LNAI), vol. 5211, pp. 408–423. Springer, Heidelberg (2008)
16. Davis, G., Carley, K.: Clearing the FOG: Fuzzy, overlapping groups for social networks. Social Networks 30(3), 201–212 (2008)
17. Suárez, A.P., Trinidad, J.F. M., Ochoa, J.A.C., Medina Pagola, J.E.: A New Incremental Algorithm for Overlapped Clustering. In: Bayro-Corrochano, E., Eklundh, J.-O. (eds.) CIARP 2009. LNCS, vol. 5856, pp. 497–504. Springer, Heidelberg (2009)
18. Macropol, K., Can, T., Singh, A.K.: RRW: repeated random walks on genome-scale protein networks forlocal cluster discovery. BMC Bioinformatics 10(283) (2009)
19. Goldberg, M., Kelley, S., Magdon-Ismail, M., Mertsalov, K., Wallace, A.: Finding Overlapping Communities in Social Networks. In: Proceedings of SocialCom 2010, pp. 104–113 (2010)
20. Magdon-Ismail, M., Purnell, J.: SSDE-CLuster: Fast Overlapping Clustering of Networks Using Sampled Spectral Distance Embedding and GMMs. In: Proceedings of SocialCom 2011, pp. 756–759 (2011)
21. Al-Hasan, M., Salem, S., Zaki, M.J.: SimClus: an effective algorithm for clustering with a lower bound on similarity. Knowledge and Information Systems 28(3), 665–685 (2011)
22. Amigó, E., Gonzalo, J., Artiles, J., Verdejo, F.: A comparison of extrinsic clustering evaluation metrics based on formal constraints. Information Retrieval 12, 461–486 (2009)
23. Gil-García, R.J., Pons-Porrata, A.: Dynamic hierarchical algorithms for document clustering. Pattern Recognition Letters 31(6), 469–477 (2010)
24. Abella-Pérez, R., Medina-Pagola, J.E.: An Incremental Text Segmentation by Clustering Cohesion. In: Proceedings of HaCDAIS 2010, pp. 65–72 (2010)
25. Jo, T., Lee, M.R.: The Evaluation Measure of Text Clustering for the Variable Number of Clusters. In: Liu, D., Fei, S., Hou, Z., Zhang, H., Sun, C. (eds.) ISNN 2007, Part II. LNCS, vol. 4492, pp. 871–879. Springer, Heidelberg (2007)
26. Meilă, M.: Comparing Clusterings by the Variation of Information. In: Schölkopf, B., Warmuth, M.K. (eds.) COLT/Kernel 2003. LNCS (LNAI), vol. 2777, pp. 173–187. Springer, Heidelberg (2003)

Fuzzy Clustering for Semi-supervised Learning – Case Study: Construction of an Emotion Lexicon

Soujanya Poria[1], Alexander Gelbukh[2], Dipankar Das[3], and Sivaji Bandyopadhyay[1]

[1] Computer Science and Engineering Department,
Jadavpur University, Kolkata, India
[2] Center for Computing Research, National Polytechnic Institute,
Mexico City, Mexico
[3] Computer Science and Engineering Department, National Institute of Technology (NIT),
Meghalaya, India
{soujanya.poria,dipankar.dipnil2005}@gmail.com,
www.gelbukh.com, sbandyopadhyay@cse.jdvu.ac.in

Abstract. We consider the task of semi-supervised classification: extending category labels from a small dataset of labeled examples to a much larger set. We show that, at least on our case study task, unsupervised fuzzy clustering of the unlabeled examples helps in obtaining the hard clusters. Namely, we used the membership values obtained with fuzzy clustering as additional features for hard clustering. We also used these membership values to reduce the confusion set for the hard clustering. As a case study, we use applied the proposed method to the task of constructing a large emotion lexicon by extending the emotion labels from the WordNet Affect lexicon using various features of words. Some of the features were extracted from the emotional statements of the freely available ISEAR dataset; other features were WordNet distance and the similarity measured via the polarity scores in the SenticNet resource. The proposed method classified words by emotion labels with high accuracy.

1 Introduction

We consider the classification task, which consists in assigning to each object one label from a predefined inventory of labels. Typical supervised classification task consists in learning the classification rules from a test set of labeled examples, in order to later apply the learned rules to previously unseen examples and thus assign them a specific label.

In contrast, unsupervised classification task, called also clustering, consists in finding internal structure in a set of data items in order to group them together. Such grouping can be interpreted as assigning the items category labels, where belonging of two items to the same group is interpreted as assigning them the same label. Note that unlike in the supervised learning, the unlabeled data items, or at least a large amount of such items, are available to the classifier at the training stage and are used for training, i.e., fining regularities in the dataset. On the other hand, no specific predefined inventory of labels is used in unsupervised classification: the groups themselves are considered equivalent to the labels.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 73–86, 2013.

Semi-supervised learning lies in the middle between these two extremes. The task consists in using both a small number of labeled examples and a large number of unlabeled examples in order to learn to assign labels from a predefined inventory to unlabeled examples. The advantage of this task over unsupervised classification is in assigning specific labels, while advantage over supervised classification is in using much smaller training labeled dataset, with the lack of information from labeled examples being compensated by the information extracted from a large set of unlabeled examples.

In this paper we propose a method for semi-supervised learning, which consists in fuzzy clustering the large set of unlabeled examples, and then using the information on the found fuzzy clusters as additional features for supervised learning from labeled examples.

In order to evaluate our proposed method, we applied it to the task of building a large emotion lexicon by extending emotion labels from a small seed labeled lexicon to a larger set of words. Building emotion lexicons is currently a very important task. While emotions are not linguistic entities, the most convenient access that we have to them is through the language (Strapparava and Valitutti, 2004). Huge bodies of natural language texts in Internet contain not only informative contents but also such information as emotions, opinions, and attitudes. Analysis of emotions in natural language texts and other media is receiving considerable and rapidly growing interest from the research community under the umbrella of subjectivity analysis and affective computing.

The majority of subjectivity analysis methods related to emotions are based on text keywords spotting using lexical resources. Various techniques have been proposed for constructing dictionaries of sentiment-related words. For emotion detection, the Affective Lexicon (Strapparava and Valitutti, 2004), a small well-organized dictionary with affective annotation, is currently one of the most widely used resources.

The aspects that govern the lexical-level semantic orientation depend on natural language context (Pang *et al.*, 2002), language properties (Wiebe and Mihalcea, 2006), domain pragmatic knowledge (Aue and Gamon, 2005), time dimension (Read, 2005), colors and culture (Strapparava and Ozbal, 2010), as well as many other aspects. Combining all these aspects of emotion orientation is related with human psychology and is a multifaceted problem (Liu, 2010). Although a word may evoke different emotions in different contexts, an emotion lexicon is a useful component for any sophisticated emotion detection algorithm (Mohammad and Turney, 2010) and is one of the primary resources to start with.

The rest of the paper is organized as follows. In Section 2, we discuss related work. An overview of the algorithm is given in Section 3. Section 4 presents the fuzzy clustering step of the algorithm, and Section 5 the final hard clustering step. Section 6 outlines an application of the proposed algorithm to a problem of constructing of an emotion lexicon via semi-supervised learning and presents the experimental results. Finally, Section 7 concludes the paper.

2 Related Work

A number of research works aimed to create subjectivity and sentiment lexica (Hatzivassiloglou and McKeown, 1997; Wiebe, 2000; Riloff *et al.*, 2003; Baroni and Vegnaduzzo, 2004; Kamps *et al.*, 2004; Hu and Liu, 2004; Andreevskaia and Bergler, 2007; Voll and Taboada, 2007; Banea *et al.*, 2008; Baccianella *et al.*, 2010). Several researchers have contributed to the study of semantic orientation of words (Kobayashi *et al.*, 2001; Turney *et al.*, 2003; Takamura *et al.*, 2005).

However, most of these resources give coarse-grained classification (e.g., *positive*, *negative* or *neutral* sentiment). Other than WordNet Affect and General Inquirer,[1] we are not aware of widely used lexical resources for fine-grained emotion analysis. In particular, the SenticNet resource (Cambria *et al.*, 2010; Cambria and Hussain, 2012) currently provides only polarity information but not specific emotion labels.

A number of other related research attempts for detecting emotions are found in the literature. Elliott (1992) considered direct emotion denoting words. Read (2005) used term co-occurrences with emotion seeds. Sidorov and Castro-Sánchez (2006) used a linguistic-based approach. Neviarouskaya *et al.* (2009) hand-crafted rules for detecting emotions. The machine learning approach by Alm *et al.* (2005) used a large number of emotion-related features, including emotion words. Recently, the application of "Mechanical Turk" for generating emotion lexicon (Mohammad and Turney, 2010) was shown to be a promising research direction.

Some results (Yu *et al.*, 2003; Awad *et al.*, 2004; Boley and Cao, 2004) show that clustering technique can help to decrease the complexity of Support Vector Machine (SVM) training. However, building the hierarchical structure in these algorithms is computationally expensive. Cervantes *et al.* (2006) presented an SVM classification algorithm based on fuzzy clustering for large data sets. We follow a similar approach and show its effectiveness on the task of classifying emotion words.

3 Overview of the Algorithm

The main idea of the method consists in using unsupervised fuzzy clustering to provide additional features and reduce the confusion set for the supervised hard clustering algorithm.

Specifically, the input data of the algorithm consist in a large number of items, or data points (words in our test case) characterized by a number of features (that form feature vectors). A small amount of data items are labeled with the desired categories; the task consists in extending the labels to other data items.

Typically, supervised machine learning algorithms learn from the labeled examples only. For example, in Fig. 1(*a*) shown are data points characterized by one feature (more features would be difficult to show in the figure); four examples are labeled with the category "–" and four with the category "+". Apparently, the data point marked with a question mark should be labeled as "+", because it lays to the right of the line separating the two sets.

[1] http://www.wjh.harvard.edu/inquirer

However, analysis of the whole set of data items shown in Fig. 1(*b*), both labeled and unlabeled, suggests that there are two clusters, and the point in question rather belongs to the one with most items labeled as “–”. Indeed, fuzzy clustering assigns membership values in two clusters, characterized by the two centroids shown in Fig. 1(*c*), with the share of membership being larger in the left one than in the right one.

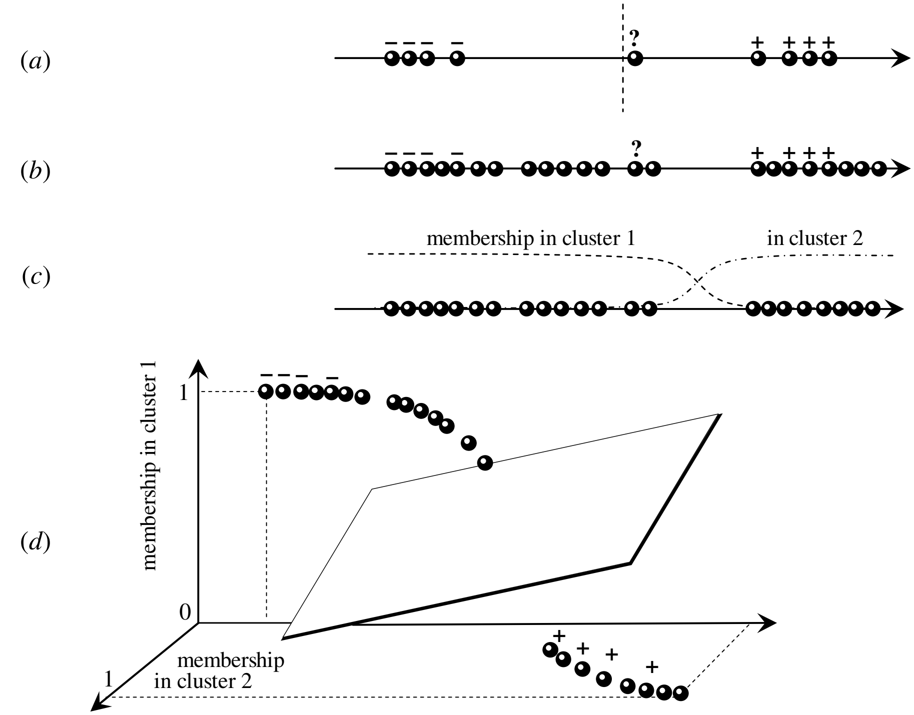


Fig. 1. The idea of the method

In our method, we add these membership values as additional features of the feature vectors, as shown in Fig. 1(*d*). If the fuzzy clustering reflects properties of the data relevant for the task, then in this higher-dimensional space the data items are better separated than in the original space, with the new coordinates providing stronger clues to the classifier and the gap being wider than in the original problem.

In addition, in case of multi-category classification, we restrict the confusion set for the classifier to the categories that correspond to the best and second-best membership value for each data point, as explained in Section 5. This requires an additional step of identifying the centroids with the categories; in Fig. 1 we would identify the left centroid with the category “–” and the right one with “+”.

With this, our algorithm consists of the following steps:

1. Fuzzy clustering of the whole set of data points, both labeled and unlabeled ones, into the number of clusters equal to the number of categories;
2. Identifying each cluster with a category label;
3. Restricting the confusion set to the best and second best category according to its membership values in the corresponding clusters;
4. For each pair of categories, training a binary classifier using only the labeled examples for which these two categories were predicted at the step 3 and for which the training label is one of the two (for training, we discard those items whose known label is neither of the two labels to which the confusion set for this item is restricted);
5. Finally, assigning the category to each unlabeled data point by applying the classifier trained for the pair of categories predicted for this data point at the step 3.

In the next sections we explain these steps in detail.

4 Fuzzy Clustering

The first step in our two-step process was unsupervised fuzzy clustering. Specifically, we used the fuzzy c-means clustering algorithm (Bezdek, 1981). Below we give the necessary background on this algorithm in the general case, and then introduce our modification to the general procedure.

4.1 General Procedure

Informally speaking, fuzzy clustering consists in optimal grouping of given data points together (that is, clustering) but in such a way that each data point can be shared by one or more clusters by belonging partially to one cluster and partially to another. In addition, each group is characterized by its location in the same space as the data points, that is, via a fictional data point representing the "center" of the cluster.

Formally, fuzzy clustering consists in finding in the feature space a set of points called centroids v_i, $i = 1, ..., c$ (where c is the desired number of clusters), and a set of membership values μ_{ik}, $k = 1, ..., N$ (where N is the number of available data points), that minimize a given objective function $J(\{\mu_{ik}\}, \{v_i\})$. The centroids characterize the constructed clusters, while the membership values μ_{ik} (often called membership functions, being considered as functions $\mu_i(x_k) = \mu_{ik}$) are interpreted as a degree with which a data point x_k belongs to the cluster characterized by the centroid v_i.

Depending on the task, such a degree can in turn be roughly interpreted as a probability of that the data point belongs to some "true" but unobservable hard cluster. In particular, the total membership of one data point in all clusters must be a unity:

$$\sum_{i=1}^{c} \mu_{ik} = 1 \tag{1}$$

for each $k = 1, ..., N$.

Often (though not in our particular case, see Section 4.2) the function J is taken as

$$J = \sum_{i=1}^{c} \sum_{k=1}^{N} \mu_{ik}^{p} \left\| x_k - v_i \right\|^2, \quad p > 1, \tag{2}$$

where the exponential constant p influences the degree of fuzziness of the obtained membership functions, and the notation

$$\left\| u \right\|^2 = \sum_{m=1}^{n} u_m^2 \tag{3}$$

denotes the square of the Euclidean length in the space of n features; u_m are the coordinates of a data point u in this space and n is the number of features given for the task—the dimensionality of the problem.

The standard procedure for fuzzy clustering consists in the following. The objective function (2) together with the constraints (1) is considered as a constraint optimization problem, which can be solved with the method of Lagrangian multipliers. This method consists in reducing a constraint optimization problem to an unconstrained optimization problem in a higher dimensional space: the problem of finding an optimum of the Lagrangian of the original system (1), (2). The Lagrangian is constructed as follows:

$$L(v, \mu, \lambda) = J + \sum_{k=1}^{N} \lambda_k g_k, \tag{4}$$

where λ_k are newly introduced auxiliary variables, v, μ, and λ are shortcuts for the sets of v_i, μ_{ik}, and λ_k, and the functions

$$g_k = 1 - \sum_{i=1}^{c} \mu_{ik} \tag{5}$$

are the penalties for violation the constraints (1).

Optimal solutions of the original system (1), (2) can be shown to be stationary points of its Lagrangian (4). The converse is not generally true, but in real-life applications it can be assumed to hold. The problem of the presence of maxima or local (but not global) minima of (4) is also conventionally ignored in real-life applications unless the experimental results suggest their presence.

With these assumptions, solving the original system (1), (2) is reduced to finding a stationary point of (4). This, in turn, is reduced to finding a point (v, μ, λ) at which

$$\frac{\partial L}{\partial v_{im}} = \frac{\partial L}{\partial \mu_{ik}} = \frac{\partial L}{\partial \lambda_k} = 0 \tag{6}$$

for all $i = 1, ..., c$; $m = 1, ..., n$; $k = 1, ..., N$, where v_{im} is the m-th coordinate of v_i in the feature space. Note that the last equality is equivalent to the constraints (1), so only the other two values are to be derived separately.

Computing-wise, the stationary point is usually found with a the following iterative procedure:

- An initial point (v, μ, λ) is chosen arbitrarily.
- At each iteration of the algorithm, each of the equations (6) is solved with respect to the corresponding variable (v_{im}, μ_{ik}, or λ_k) assuming the other values to be fixed, and the point is moved to the solution.
- The iterative process stops when the change of the objective function J between iterations becomes smaller than a predefined small constant ε.

The final position of the point (v, μ, λ) is taken as an approximate solution.

For faster convergence, when choosing the initial point, the constraints (1) are enforced by normalizing the initially random values for μ:

$$\mu_{ik} \leftarrow \frac{\mu_{ik}}{\sum_{j=1}^{c}\mu_{ij}}. \tag{7}$$

Thus, the algorithm is completely defined by the equations (6), which are to be solved with respect to v and μ (the auxiliary variables λ are of no interest).

4.2 Modified Objective Function

To obtain more compact clusters, we used the following modified objective function J; cf (2):

$$J = \sum_{i=1}^{c}\sum_{k=1}^{N}\mu_{ik}^{p}\left(\left\|x_k - v_i\right\|^2 + \rho\sum_{x_r\in N_k}\left\|x_r - v_i\right\|^2\right), \quad p > 1, \tag{8}$$

where the balancing constant ρ controls the effect of the additional member, and the sets N_k are obtained in a process of intermediate hard clustering. This process of hard clustering consisted in relating each point with the nearest centroid point: $c(x_k) = \arg\min_i \|x_k - v_i\|$; then the hard clusters were obtained as $C_i = \{x_k \mid c(x_k) = v_i\}$ and $N_k = C_i$ such that $x_k \in C_i$.

While the analytical solution for the standard objective function (2) is well-known, we had to derive the solution for our modified function (8).

Namely, for our modified objective function, the Lagrangian (4) is given by

$$L = \sum_{i=1}^{c}\sum_{k=1}^{N}\mu_{ik}^{p}\left(\left\|x_k - v_i\right\|^2 + \rho\sum_{x_r\in N_k}\left\|x_r - v_i\right\|^2\right) + \sum_{k=1}^{N}\lambda_k\left(1 - \sum_{i=1}^{c}\mu_{ik}\right).$$

Then the first part of (6) has the form

$$\frac{\partial L}{\partial v_i} = \sum_{k=1}^{N}\mu_{ik}^{p}\left(\left\|x_k - v_i\right\|^2 + \rho\sum_{x_r\in N_k}\left\|x_r - v_i\right\|^2\right) = 0$$

for all i = 1, ..., c, or, in coordinate notation:

$$\frac{\partial}{\partial v_{im}}\sum_{k=1}^{N}\mu_{ik}^{p}\left(\sum_{l=1}^{P}(x_{kl}-v_{il})^2+\rho\sum_{x_r\in N_k}\sum_{l=1}^{P}(x_{rl}-v_{il})^2\right)=$$

$$\frac{\partial}{\partial v_{im}}\sum_{k=1}^{N}\mu_{ik}^{p}\left((x_{km}-v_{im})^2+\rho\sum_{x_r\in N_k}(x_{rm}-v_{im})^2\right)=$$

$$-2\sum_{k=1}^{N}\mu_{ik}^{p}\left((x_{km}-v_{im})+\rho\sum_{x_r\in N_k}(x_{rm}-v_{im})\right)=0,$$

from which we have (back from coordinate notation to vector notation):

$$\sum_{k=1}^{N}\mu_{ik}^{p}x_k+\sum_{k=1}^{N}\rho\mu_{ik}^{p}\sum_{x_r\in N_k}x_r=\sum_{k=1}^{N}\mu_{ik}^{p}v_i+\sum_{k=1}^{N}\rho\mu_{ik}^{p}\sum_{x_r\in N_k}v_i\ ,$$

or, given that $\sum_{x_r\in N_k}v_i=v_i\sum_{x_r\in N_k}1=v_i\,|\,N_k\,|$, we have:

$$\sum_{k=1}^{N}\mu_{ik}^{p}x_k+\sum_{k=1}^{N}\rho\mu_{ik}^{p}\sum_{x_r\in N_k}x_r=v_i\left(\sum_{k=1}^{N}\mu_{ik}^{p}+\sum_{k=1}^{N}\rho\mu_{ik}^{p}\,|\,N_k\,|\right).$$

This gives

$$v_i=\frac{\sum_{k=1}^{N}\mu_{ik}^{p}\left(x_k+\rho\sum_{x_r\in N_k}x_r\right)}{\sum_{k=1}^{N}\mu_{ik}^{p}\left(1+\rho|N_k|\right)} \tag{9}$$

Similarly, the second part of (6) has the form

$$\frac{\partial L}{\partial\mu_{ik}}=p\mu_{ik}^{p-1}\left(\left\|x_k-v_i\right\|^2+\rho\sum_{x_r\in N_k}\left\|x_r-v_i\right\|^2\right)-\lambda_k=0$$

for all i = 1, ..., c; k = 1, ..., N, i.e.,

$$\mu_{ik}=\left(\frac{\lambda_k}{p\left(\left\|x_k-v_i\right\|^2+\rho\sum_{x_r\in N_k}\left\|x_r-v_i\right\|^2\right)}\right)^{\frac{1}{p-1}}. \tag{10}$$

Substituting this value in (1), we have

$$\lambda_k = \frac{p}{\left(\sum_{i=1}^{c} \frac{1}{\left(\|x_k - v_i\|^2 + \rho \sum_{x_r \in N_k} \|x_r - v_i\|^2 \right)^{\frac{1}{p-1}}} \right)^{p-1}}, \tag{11}$$

which turns (10) into

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{\|x_k - v_i\|^2 + \rho \sum_{x_r \in N_k} \|x_r - v_i\|^2}{\|x_k - v_j\|^2 + \rho \sum_{x_r \in N_k} \|x_r - v_j\|^2} \right)^{\frac{1}{p-1}}}. \tag{12}$$

It is the values (9) and (12) that we used for the iterative re-calculations at the second step of the algorithm mentioned at the end of Section 4.1 above (the values for λ_k are not really needed, though one can use (11) to calculate them).

5 Classification

Recall that our task was semi-supervised learning: we had a great amount of data items without known category for each item, and a small amount of data items for which the desired category has been manually assigned in the training set; our task consisted in extending this labeling to the whole data set.

The baseline classification method—supervised classification—consisted in using the feature vectors (the same vectors as those assumed in the previous section) of only those data points for which the category was known from the labeled training data set. With those points, a classifier was trained; then this classifier was applied to each data point for which the category was unknown, in order to relate it with some category. With this, each data point to be labeled was processed separately.

In contrast, our semi-supervised method used the internal structure learnt in an unsupervised manner from the raw data set, to help the supervised classifier in making its decisions.

For this, we first conducted fuzzy clustering of the whole available dataset (both labeled and unlabeled data points). Then we extended the N-dimensional feature vectors by c additional features: the membership values obtained in the fuzzy clustering step. The resulting $N + c$ features were used for hard classification in the usual way. The importance of the additional c features was in that they were likely strong predictors of the final class.

To take a further adventure of this fact, we restricted possible outputs of the classifier to only two variants—those that were predicted by the c features obtained from the fuzzy clustering: namely, to those two variants that corresponded to the highest and the second highest membership functions.

For example, given $c = 6$ clusters as the target, if a data point had the following membership functions in the clusters that corresponded to the following categories

cluster i	1	2	3	4	5	6
membership μ_i	0.1	**0.2**	0.1	**0.5**	0.0	0.1
category	APPLE	ORANGE	PEAR	BANANA	COCONUT	LEMON

then we forced the hard classifier to choose (using also other features not shown here) only between the categories ORANGE and BANANA for this data point, because these categories corresponded to the clusters to which the given data point was predicted to belong with the best and the second best degree.

However, for this we needed a mapping between fuzzy clusters (centroids) and categories. To find such a mapping, we used a simple majority voting. First, for each data point we selected only one cluster: the one in which it has the greatest membership (in case if several clusters tie, an arbitrary one was chosen). Next, for each cluster, the category was chosen to which the majority of the points associated with it at the previous step belonged; again, ties were resolved by a random choice. While this procedure can potentially result in not one-to-one correspondence between clusters and categories, this did not happen in our experiments.

Since the hard classifier thus needed to choose only between two possible labels, a binary classifier such as SVM was a natural choice. We trained a separate classifier for each pair of categories to choose from, that is, we trained $\binom{c}{2}$ separate binary classifiers: a classifier for two categories C_1, C_2 was trained on all training data points known to belong to C_1 or known to belong to C_2.

Finally, to classify each unlabeled data point, we determined the two categories that constituted the confusion set for it (those with the highest membership functions) and used the corresponding binary classifier.

6 Case Study: Semi-supervised Learning of an Emotion Lexicon

We applied our method to the task of semi-supervised learning of an emotion lexicon. A detailed account of the features used for classification and the obtained results can be found in (Poria *et al.*, 2012a, b, 2013).

An emotion lexicon is a dictionary that specifies for each word the main emotion typically communicated by the text where the word is used, for example:

Word	Emotion category	Word	Emotion category
offend	ANGER	*congratulate*	JOY
detestable	DISGUST	*cheerless*	SADNESS
cruelty	FEAR	*puzzle*	SURPRISE

(examples borrowed from WordNet Affect (Strapparava and Valitutti, 2004)). We assumed that words similar in some way, such as in their usage or with similar information associated with them in existing dictionaries, should be related with the same emotion.

With this, we applied the classification technique described in the previous sections to the task of extending the emotion labels from a small existing emotion lexicon to a much larger set of words for which we could collect sufficient information to form the feature vectors.

As a source of labeled examples, we used the mentioned WordNet Affect lexicon. It classifies words and some simple phrases (such as *scare away* or *the green-eyed monster*) into six categories: ANGER, DISGUST, FEAR, JOY, SADNESS, and SURPRISE.

For classification, we used two groups of features for words:

- A number of similarity measures. One set of measures were nine similarity measures based on WordNet (Miller, 1995) calculated with the WordNet::Similarity package were used. Another set were similarity measures based on co-occurrence (more specifically, the distance between occurrences) of the concepts in an emotion-related corpus, specifically, in the International Survey of Emotion Antecedents and Reactions (ISEAR) dataset (Scherer, 2005); see details in (Poria *et al.*, 2012a, b). To incorporate a similarity measure as a feature for the feature vectors, we considered each word or concept in our vocabulary as an independent dimension, and the corresponding coordinates for a given word were its similarity values with each word in the vocabulary.
- The data from the ISEAR dataset. This dataset consists of short texts (called statements) describing an emotional situation, each statement being annotated with 40 parameters, including the emotion that the statement describes (though the inventory of the basic emotions used in the ISEAR dataset slightly differs from that using in WordNet Affect). We considered each value of each parameter given in ISEAR as an independent dimension, and the corresponding coordinate value of a concept found in SenticNet was the number of times that this concept was found in the ISEAR statements annotated with this value of the parameter.

The rich set of features facilitated the unsupervised clustering of concepts in such a way that the concepts related to similar emotions we associated with the same fuzzy clusters.

We applied our method to the following sub-corpus and feature set combinations:

- $\boldsymbol{C}$: all words (after stemming) found in the ISEAR dataset. There were 449,060 distinct stemmed words in this dataset. No similarity measure was used.
- $\boldsymbol{C_{Co}}$: the same corpus, but features based on the co-occurrence similarity measure were used for this experiment.
- $\boldsymbol{C_{WA}}$: only words that co-occurred with those from WordNet Affect in an ISEAR statement. There were only 63,280 distinct stemmed words in this sub-corpus.
- $\boldsymbol{C'}$, $\boldsymbol{C'_{Co}}$, $\boldsymbol{C'_{WA}}$: the same sets, but WordNet-based and co-occurrence-based similarity measures were used in these experiments.

For evaluation, we used the membership value obtained at the step of fuzzy clustering for the class that corresponded to the label chosen at the step of final hard clustering as a confidence measure. In each corpus, we selected top 100 words with the best confidence measure, and calculated the accuracy of the final hard classification on this set. We compared the accuracy achieved by our method with the accuracy achieved for the same words by the baseline method: SVM without the fuzzy clustering step. The results are shown in Table 1.

Table 1. Accuracy (%) of the baseline (SVM only) and the proposed classifiers for top 100 confidence words on different subcorpora

Sub-corpus	SVM only	Fuzzy + SVM	Sub-corpus	SVM only	Fuzzy + SVM
C_{Co}	84.10	87.44	C'_{Co}	86.77	87.44
C	83.22	88.01	C'	85.19	90.78
C_{WA}	88.23	92.56	C'_{WA}	91.67	95.02

One can observe from the table that with each combination of a sub-corpus and the feature sets employed in the experiment, our method (denoted as Fuzzy + SVM in the table) significantly outperforms the baseline (SVM only) method.

7 Conclusions

Semi-supervised learning consists in using the inner structure of the set of unlabeled examples to aid supervised learning for classification basing on a small number of labeled examples.

We have proposed a two-step process for semi-supervised learning. At the first step, unsupervised fuzzy clustering of all available data is performed. The resulting membership functions are then used for two purposes: to reduce the confusion set for each data item and as additional features in the feature vectors. At the second step, a set of binary classifiers for the reduced confusion sets are trained in the extended feature space and are applied to assign the labels to the unlabeled data points.

We tested our method on an important task: construction of emotional lexicon. In this task, data items were words (we experimented with almost half million words) and a rich set of features were extracted from an emotion-related corpus. In addition, a number of similarity measures were used as features, namely, for each word and each similarity measure that we used, the similarity values between the given word and all words in the vocabulary were used as individual features. This gave a very large feature set suitable for unsupervised clustering.

Our experiments have shown that our suggested method outperforms the baseline classification technique, which was SVM without prior fuzzy clustering. In the future, we plan to conduct similar experiments on other classification tasks, in order to estimate the limitations and applicability of our method to a wider class of classification problems.

Acknowledgements. The work was partially supported by the Governments of India and Mexico under the CONACYT-DST India (CONACYT 122030) project "Answer Validation through Textual Entailment", the Government of Mexico under the CONACYT 50206-H project and SIP-IPN 20121823 project through Instituto Politécnico Nacional, as well as the Seventh Framework Programme of European Union, project 269180 "Web Information Quality Evaluation Initiative (WIQ-EI)".

References

1. Alm, O.C., Roth, D., Richard, S.: Emotions from text: Machine learning for text-based emotion prediction. In: Proceedings of HLT-EMNLP, pp. 579–586 (2005)
2. Andreevskaia, A., Bergler, S.: CLaC and CLaC-NB: Knowledge-based and corpus-based approaches to sentiment tagging. In: 4th International Workshop on SemEval, pp. 117–120 (2007)
3. Aue, A., Gamon, M.: Customizing sentiment classifiers to new domains: A case study. In: Proc. of RANLP (2005)
4. Awad, M., Khan, L., Bastani, F., Yen, I.L.: An Effective support vector machine (SVMs) Performance Using Hierarchical Clustering. In: Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2004), pp. 663–667 (2004)
5. Baccianella, S., Esuli, A., Sebastiani, F.: SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In: LRE, pp. 2200–2204 (2010)
6. Banea, C., Mihalcea, R., Wiebe, J.: A Bootstrapping Method for Building Subjectivity Lexicons for Languages with Scarce Resources. In: LREC (2008)
7. Baroni, M., Vegnaduzzo, S.: Identifying subjective adjectives through web-based mutual information. In: Proceedings of the German Conference on NLP (2004)
8. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York (1981)
9. Boley, D., Cao, D.: Training support vector machine Using Adaptive Clustering. In: Proc. of SIAM Int. Conf. on Data Mining, Lake Buena Vista, FL, USA (2004)
10. Cambria, E., Speer, R., Havasi, C., Hussain, A.: SenticNet: A publicly available semantic resource for opinion mining. In: Proc. of AAAI CSK, pp. 14–18 (2010)
11. Cambria, E., Hussain, A.: Sentic computing: Techniques, tools, and applications, p. 153. Springer, Dordrecht (2012)
12. Cervantes, J., Li, X., Yu, W.: Support Vector Machine Classification Based on Fuzzy Clustering for Large Data Sets. In: Gelbukh, A., Reyes-Garcia, C.A. (eds.) MICAI 2006. LNCS (LNAI), vol. 4293, pp. 572–582. Springer, Heidelberg (2006)
13. Elliott, C.: The affective reasoner: A process model of emotions in a multi-agent system. Ph.D. thesis, Institute for the Learning Sciences, Northwestern University (1992)
14. Hatzivassiloglou, V., McKeown, K.R.: Predicting the semantic orientation of adjectives. In: 35th Annual Meeting of the ACL and the 8th EACL, pp. 174–181 (1997)
15. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the ACM SIGKDD, pp. 168–177 (2004)
16. Kamps, J., Marx, M., Mokken, R.J., de Rijke, M.: Using wordnet to measure semantic orientation of adjectives. In: Proceedings of the 4th LREC 2004, IV, pp. 1115–1118 (2004)
17. Kobayashi, N., Inui, T., Inui, K.: Dictionary-based acquisition of the lexical knowledge for p/n analysis. In: Proceedings of Japanese Society for Artificial Intelligence, SLUD-33, pp. 45–50 (2001) (in Japanese)
18. Liu, B.: Sentiment Analysis: A Multi-Faceted Problem. IEEE Intelligent Systems (2010)
19. Miller, A.G.: WordNet: a lexical database for English. Communications of the ACM 38(11), 39–41 (1995)
20. Mohammad, S., Turney, P.D.: Emotions evoked by common words and phrases: Using Mechanical Turk to create an emotion lexicon. In: Proc. of NAACL-HLT, Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, pp. 26–34 (2010)
21. Neviarouskaya, A., Prendinger, H., Ishizuka, M.: SentiFul: Generating a Reliable Lexicon for Sentiment Analysis. In: ACII 2009, pp. 363–368. IEEE (2009)

22. Pang, B., Lillian, L., Shivakumar, V.: Thumbs up? Sentiment classification using machine learning techniques. In: The Proc. of EMNLP, pp. 79–86 (2002)
23. Poria, S., Gelbukh, A., Cambria, E., Das, D., Bandyopadhyay, S.: Enriching SenticNet Polarity Scores through Semi-Supervised Fuzzy Clustering. In: Proc. of the SENTIRE 2012 Workshop at IEEE ICDM 2012 (2012)
24. Poria, S., Gelbukh, A., Cambria, E., Yang, P., Hussain, A., Durrani, T.: Merging SenticNet and WordNet-Affect Emotion Lists for Sentiment Analysis. In: Proc. of the 11th International Conference on Signal Processing, IEEE ICSP 2012, Beijing (2012)
25. Poria, S., Gelbukh, A., Das, D., Bandyopadhyay, S.: Extending SenticNet with Affective Labels for Concept-based Opinion Mining. IEEE Intelligent Systems (submitted, 2013)
26. Read, J.: Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In: Proceedings of the ACL Student Research Workshop (2005)
27. Riloff, E., Wiebe, J., Wilson, T.: Learning subjective nouns using extraction pattern bootstrapping. In: Proceedings of the Seventh CoNLL 2003, pp. 25–32 (2003)
28. Scherer, K.R.: What are emotions? And how can they be measured? Social Science Information 44(4), 693–727 (2005)
29. Sidorov, G., Castro-Sánchez, N.A.: Automatic emotional personality description using linguistic data. Research in Computing Science 20, 89–94 (2006)
30. Strapparava, C., Ozbal, G.: The Color of Emotions in Texts. In: Proceedings of the 2nd Workshop on Cognitive Aspects of the Lexicon (CogALex 2010), Beijing, pp. 28–32 (2010)
31. Strapparava, C., Valitutti, A.: Wordnet affect: an affective extension of wordnet. Language Resource and Evaluation (2004)
32. Takamura, H., Inui, T., Okumura, M.: Extracting Semantic Orientations of Words using Spin Model. In: 43rd ACL, pp. 133–140 (2005)
33. Turney, P.D., Littman, M.L.: Measuring praise and criticism: Inference of semantic orientation from association. ACM TIS 21(4), 315–346 (2003)
34. Voll, K., Taboada, M.J.: Not All Words Are Created Equal: Extracting Semantic Orientation as a Function of Adjective Relevance. In: Orgun, M.A., Thornton, J. (eds.) AI 2007. LNCS (LNAI), vol. 4830, pp. 337–346. Springer, Heidelberg (2007)
35. Wiebe, J.M.: Learning subjective adjectives from corpora. In: Proceedings of the 17th National Conference on Artificial Intelligence (AAAI 2000), pp. 735–740 (2000)
36. Wiebe, J., Mihalcea, R.: Word sense and subjectivity. In: Proceedings of COLING/ACL, Sydney, Australia, pp. 1065–1072 (2006)
37. Yu, H., Yang, J., Han, J.: Classifying Large Data Sets Using SVMs with Hierarchical Clusters. In: Proc. of the 9th ACM SIGKDD (2003)

Statistical Framework for Facial Pose Classification

Ajay Jaiswal, Nitin Kumar, and R.K. Agrawal

School of Computer and Systems Sciences
Jawaharlal Nehru University, New Delhi, India
a_ajayjaiswal@yahoo.com, {nitin2689,rkajnu}@gmail.com

Abstract. Pose classification is one of the important steps in some pose invariant face recognition methods. In this paper, we propose to use: (i) Partial least square (PLS) and (ii) Linear regression for facial pose classification. The performance of these two approaches is compared with two edge based approaches and pose-eigenspace approach in terms of classification accuracy. Experimental results on two publicly available face databases (PIE and FERET) show that the regression based approach outperforms other approaches for both the databases.

Keywords: Pose classification, Partial least square, Regression, Eigenspace.

1 Introduction

Human head pose estimation/classification plays a vital role in pose invariant face recognition, human-computer interaction, human behaviour analysis, driver assistance systems and gaze estimation etc. Pose variation is one of the most challenging problems in face recognition [1, 2]. In recent methods [3-6] for face recognition across pose, first the pose of the probe face is estimated or assumed to be known and then transform it to an already learned reference pose. When multiple views of a facial image are available, pose classification can be used to reduce the search space by first estimating the pose and followed by matching the probe image with gallery images in the similar pose [6].

Various pose classification methods have been suggested in the literature [7]. Pose classification can be carried out in 3D image space or 2D image space. 3D based approaches are computationally expensive and need larger storage.

On the other hand, 2D image based methods for facial pose classification use two dimensional images for human pose classification.

In this paper, we propose to use (i) Partial least square (PLS) and (ii) Linear regression for facial pose classification, which can assist face recognition across pose. The performance of these two approaches is compared with pose-specific eigenspaces [8] and edge based pose classification [6, 9] in terms of classification accuracy on two well known publicly available face databases PIE and FERET. The rest of the paper is organized as follows. Section 2 includes the related work. Section 3 describes proposed approaches. Experimental setup and results are contained in section 4. Finally some concluding remarks are given in section 5.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 87–96, 2013.

2 Related Work

A comprehensive survey of head pose estimation methods is conducted recently by Chutorian et al. [7] and broadly classified these pose estimation methods into eight categories: (i) Appearance Template Methods (ii) Detector Array Methods (iii) Non-linear Regression Methods (iv) Manifold Embedding Methods (v) Flexible Models (vi) Geometric Methods (vii) Tracking Methods and (viii) Hybrid Methods. Each method has its own advantages and disadvantages. Appearance Template Methods [10, 11] compare probe image of a face to a set of gallery images (each labelled with a discrete pose) in order to find the most similar view. Appearance templates can handle both low and high resolution imagery and can adjust to changing conditions easily. These methods suffer from the drawback of estimating discrete poses only. Detector Array Methods [12, 13] train a series of head detectors each accustomed to a specific pose and a discrete pose is assigned to the detector with the greatest support. The main advantage of these approaches is that the steps for head localization are not needed. However, detectors' training for each discrete pose is computationally expensive. Nonlinear Regression Methods [14, 15] employ nonlinear regression tools (i.e. Support vector regression (SVR) and neural network) to obtain a functional mapping between the image or feature data and head pose. For regression tools, the high dimensionality of an image is a challenge which requires more computation and prone to error from poor head localization. These methods provide most accurate pose estimates and take less computation time when the dimensionality of images is reduced. Manifold Embedding Methods [8, 16-23] find low-dimensional manifolds that represent the continuous variation in head pose. Probe images can be embedded into these manifolds and then used for embedded template matching or regression. Manifold embedding techniques can be linear or non-linear. The linear techniques have the advantage of being simple and fast. The challenge here is to choose a dimensionality reduction algorithm that effectively recovers facial pose while ignoring other variations in the image. The manifold embedding methods for head pose estimation include Principal component analysis (PCA) [16], pose-specific eigenspaces [8], Local linear embedding [21], linear discriminant analysis (LDA) [18] Kernel LDA (KLDA) [19], Isometric feature mapping [20] and Laplacian eigenmaps [22] etc. In all these techniques, head pose of the test image is estimated by embedding into an existing manifold. With the exception of LDA and KLDA, all other manifold embedding approaches operate in an unsupervised fashion. Consequently, they have the affinity to build manifolds for identity as well as pose [23]. The limitation of PCA is that it is linear transform and cannot adequately represent nonlinear image variations caused by pose variation. Also it is unsupervised technique, therefore, it is not guaranteed that the dominant components will relate to pose variation rather than to appearance variation. To alleviate these problems, Srinivasan et al. [8] have suggested another approach called pose-eigenspaces. In this approach, appearance information is decoupled from the pose by dividing the training data according to poses. Then PCA is applied to generate a separate projection matrix for each pose. These pose-specific eigenspaces, or pose-eigenspaces, each representing the primary modes of appearance variation and provide a decomposition that does not depend on the pose variation.

Head pose is estimated by normalizing the test image and projecting it into each of the pose-eigenspaces, thus finding the pose with the highest projection energy [8]. In another approach proposed by Li et al. [17], the projected samples can be used as the input to a set of classifiers, such as multi-class support vector machines. The major drawback of using pose-eigenspaces is that only discrete poses can be estimated. Flexible Models [24-26] fit a non-rigid model to the facial structure of each person in the image space. Head pose is estimated from feature-level comparisons or from the instantiation of the model parameters. These methods provide good invariance to head localization error. The main drawback of these methods is that the training data with annotated facial features is required but comparisons can be done at feature level rather than the global image. Geometric Methods [27, 28] make use of the location of features such as the eyes, mouth, and nose tip to determine pose from their relative arrangement. Geometric methods have the advantage of being fast and simple, but need of precise locations of the facial features makes these methods vulnerable to bad illumination and low resolution. Tracking Methods [29, 30] obtain the pose changes from the observed variations between video frames. Hybrid Methods [31] combine one or more of these above mentioned approaches to conquer the drawbacks intrinsic in any single approach.

Recently, Choi et al. [6] have suggested another pose estimation method. In this approach, the images of different subjects are arranged according to poses in such a way that nearest poses are placed together and each pose represents one class. Histogram equalization is used to pre-process all the images. Then edges are obtained from images using Sobel operator and PCA+LDA is used to perform pose classification. This particular approach is also capable of handling illumination variation in the images. In [9], a comparative study of various combinations of edge operators and linear subspace methods for facial pose classification has been provided. The performance of these methods is evaluated under varying illumination. It is pointed out that the performance of eigenfeature regularization (Eigreg) [32] with Prewitt and Sobel edge operators is best among evaluated methods. We also investigate the performance of this approach on the images without illumination variation. Now we briefly describe the proposed approaches.

3 Proposed Approaches

3.1 Partial Least Square

As the dimension of sample face images is always high, there is need to reduce the dimension of feature vector. One of the most widely used methods for dimensionality reduction is PCA. It is used to obtain a few non-redundant feature components in the direction of maximum variances to represent the high dimensional feature vector. Also, the transformation may results into increase in intra-class scatter along with inter-class scatter. In contrast to PCA, partial least squares (PLS) determine components such that the covariance between the class label and a linear combination of the original feature vector is maximized. PLS, based on the principle of statistical learning, is widely used in chemo metrics and bioinformatics etc. [33].

In the proposed approach, we have used PLS [34] method for reducing the dimensionality before estimating the facial pose. PLS is a supervised learning and can be used to reduce p-dimensional image to K-dimensional space such that K<<p. Suppose X is N×p matrix which represents N images with p original face features. The features are normalized to have zero mean and unit variance. For C+1 pose classes (pose: 0, 1,...,C), we define C-dimensional random vector $\mathbf{y} = (y_1, y_2,..., y_C)'$ such that $y_i = 1$ and $y_j = 0$ for all $j \neq i$ when the face image is from class $i-1$; i=1, 2,..., C, and y_i =1 for all i=1, 2,...,C, when the image belongs to the class C. The observation of class vector is obtained $\{\mathbf{y}_1, \mathbf{y}_2,..., \mathbf{y}_N\}$ from the training sample of images to construct the N×C class matrix $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2,..., \mathbf{y}_N]'$.

The objective of PLS is to find weight vectors **b** and **c** such that

$$\mathbf{b}_k = \underset{\mathbf{b'b}=1}{\arg} \underset{\mathbf{c'c}=1}{\max} \mathrm{Cov}^2(\mathbf{Xb},\mathbf{Yc}) \qquad \text{for } k=1,2,...,K \tag{1}$$

subject to the orthogonality constraint $\mathbf{b}_k\mathbf{S}\mathbf{b}_j = 0$ for all $1 \le j < k$, Where $\mathbf{S} = \mathbf{X}'\mathbf{X}$ and **b** and **c** are unit vectors. The above procedure is called multivariate PLS. The kth PLS component is given by $\mathbf{Xb}_k$. Suppose $\mathbf{B}=[\mathbf{b}_1, \mathbf{b}_2, \ldots,\mathbf{b}_K]$ and $\mathbf{x}_p$ and $\mathbf{x}_g$ be the probe image and gallery image respectively.

$$\begin{aligned} \mathbf{t} &= \mathbf{x}_p\mathbf{B} \\ \mathbf{u} &= \mathbf{x}_g\mathbf{B} \end{aligned} \tag{2}$$

The similarity s between $\mathbf{x}_p$ and $\mathbf{x}_g$ can be simply measured by the correlation between **t** and **u** which is given by

$$\mathbf{s} = \frac{\langle \mathbf{t},\mathbf{u} \rangle}{\|\mathbf{t}\|.\|\mathbf{u}\|} \tag{3}$$

where t and u are the score vectors of probe and gallery image respectively and $\langle \mathbf{t},\mathbf{u} \rangle$ denotes the inner product of t and u. In pose classification, a probe face is simply classified to the pose class with highest value of s.

3.2 Linear Regression

Togneri et al. [35] have proposed LR for face recognition for handling expression variation and contiguous occlusion. They proposed a model to represent a probe image as a linear combination of class-specific images based on assumption that patterns from a single-object class lie on a linear subspace. The least-squares method is used to solve inverse problem and the verdict is made in favour of the class with the minimum reconstruction error. We propose to use LR for facial pose classification. In this approach for facial pose classification, training face images are arranged in such a way that nearest poses is placed together. Let $\mathbf{X_i} \in \Re^{\mathbf{d} \times \mathbf{n}_i}, i=1,2,...,p$ represent the set of training images in pose i. If test image is represented as $\mathbf{x_t} \in \Re^{\mathbf{d}}$, then output class (pose) of $\mathbf{x_t}$ is evaluated as follows:

$\boldsymbol{\beta}_{\mathbf{i}} \in \Re^{\mathbf{n}_i}$ is evaluated against each class mode which is given by

$$\boldsymbol{\beta}_{\mathrm{i}} = (\mathbf{X}_i^T \mathbf{X}_{\mathrm{i}})^{-1} \mathbf{X}_i^T \mathbf{x}_{\mathrm{t}} \tag{4}$$

$\mathrm{y_i}$ is computed for each $\boldsymbol{\beta}_{\mathrm{i}}$,

$$\mathbf{y}_{\mathrm{i}} = \mathbf{X}_{\mathrm{i}} \boldsymbol{\beta}_{\mathrm{i}}, \quad i = 1,2,\ldots,p \tag{5}$$

The distance between original and predicted variables is given by

$$\mathbf{d}_{\mathrm{i}}(\mathbf{x}_{\mathrm{t}}) = \left\| \mathbf{x}_{\mathrm{t}} - \mathbf{y}_{\mathrm{i}} \right\| \qquad i = 1,2,\ldots,p \tag{6}$$

Finally, the class with minimum distance $\mathbf{d}_{\mathrm{i}}(\mathbf{x}_{\mathrm{t}})$ is selected.

4 Experimental Setup and Results

The performance of the proposed approaches was evaluated in terms of average classification accuracy and compared with the following approaches: (i) pose-specific eigenspace (ii) Eigenfeature regularization using Sobel edge operator (iii) PCA+LDA using Sobel edge operator on two well known publicly available face databases PIE [36] and FERET [37]. In all the experiments, images were coarsely aligned according to the position of eyes and mouth corners in the pre-processing step. Then, images were cropped and finally resized to 50×50. The histogram equalization was performed on the images used in experiments.

In our experiments, images of seven poses from the PIE database were used. It covers the pose yawing over ±45 degree and the pitching variations in depth [36]. The poses used in our experiments are: P07 and P09 (pitching about ±20 degree), P05 and P29 (yawing about ±22.5 degree), pose set P37 and P11 (yawing about ±45 degree), and P27 (near frontal), respectively. Each pose class includes 68 subjects. Example faces of PIE database are shown in Table 1. The FERET database subset contains images of 196 subjects in seven poses (ba, bc, bd, be, bf, bg and bh) as shown in Table 2.

Table 1. Sample faces of a subject under different poses in PIE database

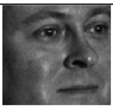
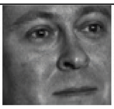
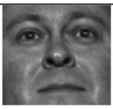
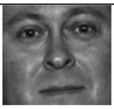
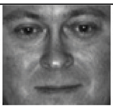
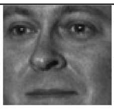
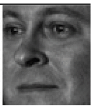
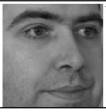
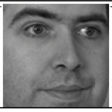
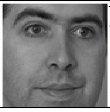
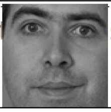
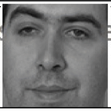
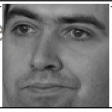
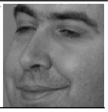
Pose Angle	P37 +45°	P05 +22.5°	P07 Pitch up	P27 Frontal	P09 Pitch Down	P29 -22.5°	P11 -45°
Examples							

Table 2. Sample faces of a subject under different poses in FERET database

Pose Angle	bc +40°	bd +25°	be +15°	ba Frontal	bf -15°	bg -25°	bh -40°
Examples							

For all the five approaches, we carried out 2-fold and 3-fold cross-validations to investigate the effect of number of training images on average classification accuracy. Each experiment is run 10 times to obtain average classification accuracy. The comparative results of different methods on PIE database using 2-fold and 3-fold cross-validations are shown in Table 3, Figure 1 and figure 2.

Table 3. Results on PIE database using 2-fold and 3-fold cross-validations

Pose	Eigreg using Sobel edge operator		PCA+LDA using Sobel edge operator		Pose-eigenspaces		**PLS based**		**Regression based**	
	2-fold	3-fold	2-fold	3-fold	2-fold	3-fold	2-fold	3-fold	2-fold	3-fold
P37 (+45°)	98.08	99.26	99.71	99.85	97.21	99.41	99.56	99.56	**100**	**100**
P05(+22.5°)	96.32	96.03	98.97	99.41	99.41	99.26	96.76	97.35	**100**	**100**
P07 (Pitch up)	73.82	74.26	85.88	87.65	91.91	94.26	89.26	89.56	**100**	**100**
P27 (frontal)	63.97	67.64	79.58	80.92	77.09	83.84	71.65	74.71	**96.89**	**97.09**
P09 (Pitch down)	87.65	89.56	87.5	87.21	87.94	92.79	78.38	80.44	**99.12**	**99.41**
P29 (-22.5°)	94.12	96.62	98.66	99.26	99.41	99.85	99.85	**100**	**100**	**100**
P11 (-45°)	**99.85**	**100**	**100**	99.85	99.71	**100**	**100**	**100**	**100**	**100**
Avg. all poses	87.69	89.05	92.9	93.45	93.24	95.63	90.78	91.66	**99.43**	**99.5**

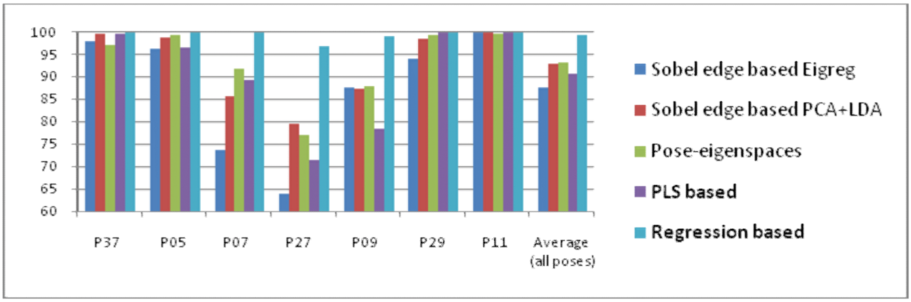


Fig. 1. Comparative results on PIE database using 2-fold cross validation

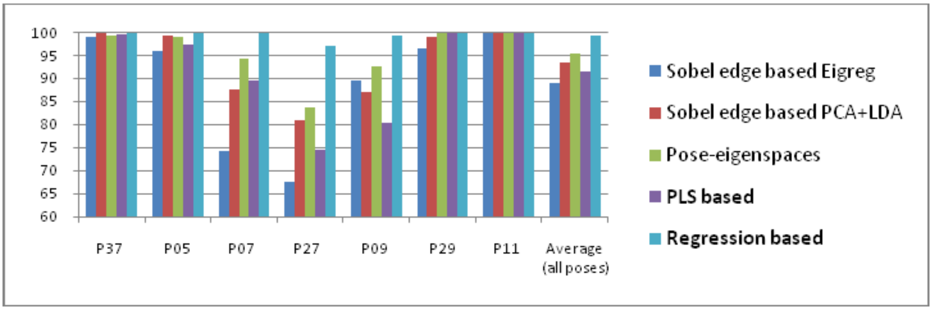


Fig. 2. Comparative results on PIE database using 3-fold cross validation

We observed the following from Table 3, Figure 1 and Figure 2:

- For nearly frontal poses (P07, P27 and P09), the regression based method outperformed other approaches.
- For right profile poses (P37 and P05), all the approaches performed well but the performance of regression based approach was best.
- For left profile poses (P29 and P11), the performance of all the methods was comparable except Eigreg using Sobel edge operator for pose P29.
- For pose P05, P07, P27 and P29, the performance of Eigreg using Sobel edge operator was worst.
- The performance of the proposed approaches either improved marginally or remained same as we moved from 2-fold to 3-fold cross validation.
- Regression based approach comparatively outperformed all other approaches in terms of average classification accuracy over all poses.

Table 4. Results on FERET database using 2-fold and 3-fold cross-validations

Pose	Eigreg using Sobel edge operator		PCA+LDA using Sobel edge operator		Pose-eigenspaces		**PLS**		**Regression**	
	2-fold	3-fold	2-fold	3-fold	2-fold	3-fold	2-fold	3-fold	2-fold	3-fold
bc (+40°)	96.43	97.09	98.27	97.04	99.74	99.95	99.74	99.95	**100**	**100**
bd (+20°)	86.48	86.68	90.05	89.08	96.84	98.88	90.61	92.3	**99.74**	**100**
be (+15°)	72.04	73.31	87.4	86.07	94.49	94.7	78.11	78.72	**97.3**	**97.3**
ba (frontal)	63.06	62.55	89.84	88.54	96.15	97.25	85.91	87.7	**96.66**	**97.27**
bf (-15°)	84.23	85.1	93.52	92.14	98.88	99.49	90.82	91.22	**99.03**	**99.29**
bg (-25°)	83.83	81.22	93.67	91.99	98.67	99.03	91.99	92.7	**99.29**	**99.49**
bh (-40°)	88.57	86.99	97.76	95.92	99.9	**100**	99.54	99.59	**100**	**100**
Avg. all poses	82.09	81.85	92.93	91.54	97.81	98.47	90.96	91.74	**98.86**	**99.05**

The comparative results of all the methods using 2-fold and 3-fold cross-validations on FERET database are shown in Table 4. It was observed from Table 4:

- For all the poses regression approach performed best and Eigreg using Sobel edge operator performed worst in both cross-validations.
- For poses bh and bc the performance of PLS based, Pose-eigenspace and regression based approaches was comparable.
- In terms of average classification accuracy over all poses, regression based approach comparatively outperformed all other approaches.
- Average classification accuracy over all poses increased in 3-fold cross validation in comparison to 2-fold cross validation for all the methods except edge based approaches.

5 Conclusion

In this paper we proposed to use PLS and regression for facial pose classification. We compared the performance of these two approaches with two edge based approaches and pose-eigenspace based approach in terms of average classification accuracy. Experimental results suggest that the regression based approach outperforms other approaches for both the databases. Also the regression based approach performs better than other approaches even when pose variation is very small. In future work, we would like to combine regression based approach with some pose invariant face recognition method to fully automate face recognition system.

References

1. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face recognition: a literature survey. ACM Comput. Surv. 35(4), 399–459 (2003)
2. Chen, S., Tan, X., Zhou, Z.H., Zhang, F.: Face recognition from a single image per person: a survey. Pattern Recognition 39(9), 1725–1745 (2006)
3. Sharma, A., Dubey, A., Tripathi, P., Kumar, V.: Pose invariant virtual classifiers from single training image using novel hybrid-eigenfaces. Neurocomputing 73, 1868–1880 (2010)
4. Li, A., Shan, S., Chen, X., Gao, W.: Cross-pose face recognition based on partial least squares. Pattern Recognition Letters 32(15), 1948–1955 (2011)
5. Chai, X., Shan, S., Chen, X., Gao, W.: Locally linear regression for pose-invariant face recognition. IEEE Trans. Image Process. 16(7), 1716–1725 (2007)
6. Choi, S., Choi, C., Kwak, N.: Face recognition based on 2D images under illumination and pose variations. PR Letters 32(4), 561–571 (2011)
7. Chutorian, E.M., Trivedi, M.M.: Head Pose Estimation in Computer Vision: A Survey. IEEE Trans. on PAMI 31(4), 607–626 (2009)
8. Srinivasan, S., Boyer, K.: Head pose estimation using view based eigenspaces. In: Proc. Int'l. Conf. Pattern Recognition, pp. 302–305 (2002)
9. Jaiswal, A., Kumar, N., Agrawal, R.K.: Illumination Invariant Facial Pose Classification. International Journal of Computer Application 37(1), 14–19 (2012)
10. Niyogi, S., Freeman, W.: Example-based head tracking. In: Proc. Int'l. Conf. Automatic Face and Gesture Recognition, pp. 374–378 (1996)

11. Beymer, D.: Face recognition under varying pose. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 756–761 (1994)
12. Huang, J., Shao, X., Wechsler, H.: Face pose discrimination using support vector machines (SVM). In: Proc. Int'l. Conf. Pattern Recognition, pp. 154–156 (1998)
13. Jones, M., Viola, P.: Fast multi-view face detection. Mitsubishi Electric Research Laboratories. Tech. Rep. 096 (2003)
14. Li, Y., Gong, S., Sherrah, J., Liddell, H.: Support vector machine based multi-view face detection and recognition. Image and Vision Computing 22(5), 413–427 (2004)
15. Seemann, E., Nickel, K., Stiefelhagen, R.: Head pose estimation using stereo vision for human-robot interaction. In: Proc. IEEE Int'l. Conf. Automatic Face and Gesture Recognition, pp. 626–631 (2004)
16. McKenna, S., Gong, S.: Real-time face pose estimation. Real-Time Imaging 4(5), 333–347 (1998)
17. Li, S., Fu, Q., Gu, L., Scholkopf, B., Cheng, Y., Zhang, H.: Kernel machine based learning for multi-view face detection and pose estimation. In: Proc. IEEE Int'l. Conf. Computer Vision, pp. 674–679 (2001)
18. Duda, R., Hart, P., Stork, D.: Pattern classification, 2nd edn. John wiley & sons, Inc. (2001)
19. Chen, I., Zhang, L., Hu, Y., Li, M., Zhang, H.: Head pose estimation using Fisher Manifold learning. In: Proc. IEEE Int'l. Workshop Analysis and Modeling of Faces and Gestures, pp. 203–207 (2003)
20. Raytchev, B., Yoda, I., Sakaue, K.: Head pose estimation by nonlinear manifold learning. In: Proc. Int'l. Conf. Pattern Recognition, pp. 462–466 (2004)
21. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500), 2323–2326 (2000)
22. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation 15(6), 1373–1396 (2003)
23. Balasubramanian, V., Ye, J., Panchanathan, S.: Biased manifold embedding: A framework for person-independent head pose estimation. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition (2007)
24. Krüger, V., Pötzsch, M., von der Malsburg, C.: Determination of face position and pose with a learned representation based on labeled graphs. Image and Vision Computing 15(8), 665–673 (1997)
25. Lanitis, A., Taylor, C., Cootes, T.: Automatic interpretation of human faces and hand gestures using flexible models. In: Proc. IEEE Int'l. Conf. Automatic Face and Gesture Recognition, pp. 98–103 (1995)
26. Cootes, T., Walker, K., Taylor, C.: View based active appearance model. In: Proc. Int'l. Conf. Automatic Face and Gesture Recognition, pp. 227–232 (2000)
27. Gee, A., Cipolla, R.: Determining the gaze of faces in images. Image and Vision Computing 12(10), 639–647 (1994)
28. Horprasert, T., Yacoob, Y., Davis, L.: Computing 3-d head orientation from a monocular image sequence. In: Proc. Int'l. Conf. Automatic Face and Gesture Recognition, pp. 242–247 (1996)
29. Yao, P., Evans, G., Calway, A.: Using affine correspondence to estimate 3-d facial pose. In: Proc. Int'l. Conf. Image Processing, pp. 919–922 (2001)
30. Schödl, A., Haro, A., Essa, I.: Gaze tracking using a textured polygonal model. In: Proc. Workshop Perceptual User Interfaces (1998)
31. Jebara, T., Pentland, A.: Parametrized structure from motion for 3d adaptive feedback tracking of faces. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 144–150 (1997)

32. Jiang, X., Mandal, B., Kot, A.: Eigenfeature Regularization and Extraction in Face Recognition. IEEE Trans. on PAMI 30(3), 383–394 (2008)
33. Rosipal, R., Kramer, N.: Overview and recent advances in partial least squares. Subspace Latent Struct. Feat. Select., 34–51 (2006)
34. Baek, J., Kim, M.: Face recognition using partial least squares components. Pattern Recognit. 37(6), 1303–1306 (2004)
35. Naseem, I., Togneri, R., Bennamoun, M.: Linear Regression for Face Recognition. IEEE Trans. PAMI 32(11), 2106–2112 (2010)
36. Sim, T., Baker, S., Bsat, M.: The CMU pose, illumination, and expression database. IEEE Trans.PAMI 25(12), 1615–1618 (2003)
37. Phillips, P.J., Wechsler, H., Huang, J., Rauss, P.: The FERET database and evaluation procedure for face recognition algorithms. Image Vision Comput. 16(5), 295–306 (1998)

Forecast of Air Quality Based on Ozone by Decision Trees and Neural Networks

Nahun Loya, Iván Olmos Pineda, David Pinto,
Helena Gómez-Adorno, and Yuridiana Alemán

Benemérita Universidad Autónoma de Puebla
Puebla, México
{nahun.loya,helena.adorno,yuridiana.aleman}@gmail.com,
{iolmos,dpinto}@cs.buap.mx

Abstract. In this paper we explore models based on decision trees and neural networks models for predicting levels of ozone. We worked with a data set of the Atmospheric Monitoring System of Mexico City (SIMAT), which includes measurements hour by hour, between 2010 to 2011. The data come from of three meteorological stations: Pedregal, Tlalnepantla and Xalostoc in Mexico city. The data set includes 8 parameters: four chemical variables and four meteorological variables. Based on our results, it's possible to predict ozone levels with these parameters, with an accuracy of 94.4%.

Keywords: Decision Trees, C4.5, Neural networks, Ozone.

1 Introduction

Big cities such as Los Angeles, Tokyo, Moscow, and Mexico City have serious problems of air pollution. These cities monitoring the air quality in the troposphere, with the aim to detect ozone concentration emitted by inhabitant [1], and record its progress. Globally this pollutant is one of the most important, which is a triatomic molecule of oxygen [2].

The study of gas concentration is crucial, because if some positive variations of levels of ozone are detected, should suggest that controls on emissions are having a positive effect in the environment, and the absence of such trends suggest the need to change that controls.

This work aims to build models for predict the air quality based on the ozone as primary pollutant. We consider a set of chemical and atmospheric attributes that authors and experts of air pollution have shown to be predictors for air quality. The prediction is done using models of multilayer neural networks and decision trees, using the Weka toolkit as data mining software [3].

The main objective of this work is to find a good model based on neural networks and decision trees, with the aim to predict air quality. We worked with the following attributes in our experiments: Ozone (O_3), Carbon monoxide (CO), Nitrogen Dioxide (NO_2), Sulfur dioxide (SO_2), Temperature (TMP), Relative humidity (RH), Speed wind (WSP), and Wind direction (WDR). The

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 97–106, 2013.

data considered in this work comes from of the Atmospheric Monitoring System of Mexico City (SIMAT, for its acronym in Spanish).

We proposed a methodology where a descriptive statistics for determining the general behavior of the data is initially performed. Also, we consider a preprocessing phase, including: a data cleaning, a data integration, and a data reduction. Then, a training phase is performed, taking as input the attributes selected in the previous phase. As result of this process, some classification models for predict the air quality were obtained. These models were tested with a cross validation.

This paper is organized as follows: section 2 shows several studies that have been conducted to predict levels of air quality using statistical tools, and other schemes such as neural networks. Section 3 describe the case of study, as well as the locations of weather stations considered in this study. Section 4 describes how the integration of data is done and the data clean is performed. Finally, section 5 presents the results obtained in our experiments.

2 Related Work

Several works have been conducted to predict the air quality with respect to the ozone. For example, Seinfeld *et al.*[4] shows how it is possible to measure tendencies of ozone levels based on estimators, such as: maximum daily, and average of the maximum daily in a period of 3 days.

In Mexico, there are attempts of many different scientific fields and institutions, which are trying to evaluate the ozone pollution [5], [6], [7]. In general, these works try to predict areas with high risk for inhabitants of the metropolitan area in Mexico City (ZMVM), using extreme values of pollutants.

Another works use data from only one monitoring site in Mexico City, where high levels of ozone are recorded and analyzed Garfias*et al.*[8]. In such investigations, authors proposed three different models for predict concentration of ozone based on 19 semi-annual observations.

On the other hand, there are different statistical techniques for predicting levels of ozone. For example, Aguirre *et al.*[9] shows the importance of neural networks in this task, where a multilayer perceptron model was used to predict maximum levels of ozone in the Pais Vasco, Spain.

Barai *et al.*[10] proposed neural networks to predict air quality, which works with a limited number of data, and this model is capable to work with noise. They proposed different models of neural networks for predicting air quality, with a very acceptable accuracy, using only two data sets: US Environmental Protection Agency (US. EPA) and Tata Energy Research Institute (TERI). These works are good examples of what it is possible to do with neural networks for predicting air pollutants.

In this paper, we explore important approaches with the aim to learn patterns that predict levels of ozone, such as multilayer perceptron neural networks, and classification trees models, including C4.5, and Random forest.

In the next section are explained details of the dataset used in this work.

3 Case of Study

In Mexico city several meteorological stations were created, which reports levels of pollutants hour by hour. In this study we focus in the dataset that come from of the SIMAT network, including data between the period of January 2010 to December 2011. This study includes data from three meteorological stations: Pedregal, Tlalnepantla and Xalostoc, which are strategically located in the metropolitan area of Mexico city. The size of the dataset is 420,480 records, including 8 attributes, recollected in 2 years of study. In this work we consider a set of attributes described by Seinfeld [4], which are predictors of ozone levels. These attributes are shown in Table 1. As we can see, the first column represents pollutants (Ozone O_3, Carbon monoxide CO, Nitrogen Dioxide NO_2, Sulfur dioxide SO_2) and atmospheric variables (Temperature TMP, Relative humidity RH, Speed wind WSP and Wind direction WDR). The highest values admitted by each pollutant according to the Official Mexican Norm (NOM-1993)[11] are present in the second column. The third column shows the measurement units for each variable [2]. Finally, last column represents the number of missing values by attribute.

Table 1. Set of abbreviations used in this paper

Pollutant/ Atmospheric variable	Value of the Standard on accordance with (NOM-1993)	Unit of Measure	Missing Values
Ozone (O_3)	0.11 ppm	Parts per million (ppm)	0
Carbon monoxide (CO)	11 ppm	Parts per million (ppm)	1088
Nitrogen Dioxide (NO2)	0.21 ppm	Parts per million (ppm)	1011
Sulfur Dioxide (SO2)	0.13 ppm	Parts per million (ppm)	792
Temperature (TMP)		Celsius Grades (C)	758
Relative Humidity (RH)		Percent (%)	369
Wind speed (WSP)		Meters over second (m/s)	925
Wind Direction (WDR)		North Grados	7925

Based on the data described above, we propose a methodology for buil–ding models with ANNs and decision trees. The next section describe the data preprocessing phase implemented in this work.

4 Data Preprocessing

In this section we present a data overview based on descriptive statistics, in order to give a general data perspective. This analysis let us to propose a data integration phase, and how missing values can be estimated.

4.1 Descriptive Statistics

Descriptive statistic is an important mathematical tool because allow us to understand the data behavior. With the aim to known trends of air pollutants, we compute statistical tendencies of each pollutant, which are shown in Table 2. In this table, “Min”, “Max” and “Mean” are the minimum, maximum and mean of each pollutant per each meteorological station.

Table 2. Data summary per each meteorological station

	Pedregal (DSN=189)			Tlalnepantla (DSN=121)			Xalostoc (DSN=86)		
Attribute.	Min.	Mean	Max.	Min.	Mean	Max.	Min.	Mean	Max.
O_3	0.000	0.033	0.182	0.000	0.026	0.183	0.000	0.024	0.150
CO	0.000	0.515	2.900	0.000	0.920	5.100	0.000	1.102	11.30
NO_2	0.001	0.025	0.115	0.004	0.033	0.161	0.000	0.034	0.138
SO_2	0.000	0.005	0.097	0.000	0.009	0.283	0.000	0.007	0.143
TMP	1.00	16.08	31.50	1.60	17.92	36.00	1.200	17.50	33.80
RH	0.000	45.83	95.00	1.00	45.46	100.00	1.00	46.80	100.00
WDR	0.000	-	360	0	-	360	0	-	360
WSP	0.000	1.80	7.60	0.00	2.00	8.20	0.00	2.0	10.90

In this table, we can see that Pedregal is the most contaminated meteorological station in this study. We can see that 189 times the NOM-1993 with respect of ozone levels was overcoming (DSN). In Tlalnepantla, the meteorological station recorded 121 times where the NOM-1993 was exceeding. In Xalostoc, located at the north of the Mexico City, the meteorological station recorded the lowest rate of contamination in the observation period, with only 86 events. Moreover, in this analysis is evident that most observations show that levels of air pollutants are acceptable. Note that these observations not include atmospheric variables such as the wind direction, because measures of central tendency such as mean and median are not good estimators.

One of the main objectives in this work is to predict with a good confidence level the air quality for the zone area surrounding each meteorological station considered in this work. In the following sections we present how it is possible to apply technics of data mining with the goal to discover such tendencies, including phases such as data cleaning and data integration.

4.2 Preprocessing Phase: Data Cleaning, Missing Values and Data Integration

The input database have missing values, non-relevant information (chemical variables), data from other stations not considered in this study, data in different unit measures, and noise. Therefore, all these problems were overcoming with different tools implemented in the R software [12]. Moreover, each database instance was categorized as good, regular, bad, very bad, and highly bad, according with the NOM-1993, and according with the ozone levels. In Table 3 are shown the ranges used in this work.

Table 3. Ozone level ranges used to categorize each instance

O_3	*Class*	Qualifier air quality
0.000-0.055	Green	Good
0.056-0.110	Yellow	Regular
0.111-0.165	Orange	Bad
0.166-0.220	Red	Very Bad
> 0.220	Purple	Highly bad

Although it is known that the selected classifiers can work with noise and missing values, we implement a phase where these problems are solved. Since data is continuous with respect of time, we implement a simple technique based on a linear interpolation, with the aim to detect outlier values and predict missing values. In both cases, if a value v_n is atypical or unknown, then it is computed based on the neighbors v_{n-1} and v_{n+1}.

4.3 Attribute Selection

Before to proceed with the training process, it is necessary to know the attributes that are more relevant for that process. In this phase, we propose to use an attribute evaluation technique based on the chi-square metric, where is computed the importance of each attribute considering the chi-square value with respect to the class attribute. The attribute with a value closer to 1 is the attribute that provides more information for the prediction class. In Table 4 are shown the values obtained in this process. With these results, we have a good idea of the importance of each attribute.

Table 4. Results using a χ^2 test for each meteorological station

Attribute	PEDREGAL	TLALNEPANTLA	XALOSTOC
HOUR	3.8+-0.4	3.2+-0.7	3.3+-0.4
CO	3.1+-0.5	2.9+-0.9	2.0+-0.0
NO_2	5.9+-0.3	3.7+-1.3	6.1+-0.8
SO_2	2.1+-0.3	4.8+-1.6	6.5+-2.2
TMP	1.0+-0.0	1.0+-0.0	1.0+-0.0
RH	5.1+-0.3	5.6+-0.4	5.0.+-0.6
WDR	7.6+-0.4	7.9+-0.3	7.3+-0.4
WSP	7.4+-0.4	6.9+-0.3	4.8+-0.7

On the other hand, in our analysis we can see that the dataset has a imbalance of classes. The imbalance produces poor results for predicting tasks. Then, we implement a phase where this problem was overcoming.

4.4 Removing the Class Imbalance

At this stage in our process, the original dataset has been preprocessed, removing noise, inconsistencies, etc. The next step consist of to perform a stratified sampling, where the strata are the seasons: spring, summer, fall and winter. This process try to preserve the distribution of the data with respect to the time. This process was implemented using a script in “awk”, where the program select an instance in the dataset with respect to a probability p (this value is defined by the user). Also, the balancing of the classes is preserved with respect to the total sample size: if an instance to be selected belongs to a class where have been achieved the maximum number of instances, then the instance is not selected. With this simple rule, we preserve the balancing of the classes.

For our experiments, three different samples per each meteorological station were generated, with the aim to perform cross validation. In the next section we show the results obtained in our experiments.

5 Experimental Results

In our experiments, we generate three balanced samples of the original data set from each meteorological station, each of one with 42000 records. In our experiments, we worked with two types of machine learning algorithms, such as neural networks (multilayer perceptron), and decision trees (C4.5 and the Random forest). We used the Weka tool kit in our experiments.

The training of the neural network was performed with different configurations, all of them with 10 fold cross validation. Some parameters considered in these experiments include a learning rate = 0.3, momentum = 0.2. Different topologies were tested, with 1, 2, and 3 hidden layers (HL), with 3, 4, 5, 8, and 9 neurons ($\#Neu$) per layer. Also, we consider a topology with a hidden layer, and a $= \frac{No.Att+No.Class}{2}$ neurons, where $No.Att$ is the number of attributes, and $No.Class$ represents the number of classes. Further, we experimented with different epochs: 500, 1000, and 2000. Our results are shown in Table 5, where each entry is the percentage accuracy.

On the other hand, we explored with two algorithms based on decision trees, such as: C4.5 and Random Forest. We also used the implementation available in Weka. Just as in neural networks, we perform a search of parameters for each algorithm. In the case of C4.5, we used the following parameters: $factor = 0.05$, $MinNumObj = 2$ and $Unpruned = FALSE$. Furthermore, in the case of the Random Forest algorithm, the parameters used in our experiments were: $MaxDepth = 10$, $Debug = False$, $NumTrees = 50$, $Seed = 1$. Results with these algorithms are shown in Table 6.

As we can see in Tables 5 and 6, the best results are obtained with the classification trees. According with the Table 5, the best result for the Pedregal station is with a multilayer perceptron neural network, with one hidden layer and 8 neurons, where the accuracy is 87.8%. For the case of Tlalnepantla station, the best result was obtained with a similar configuration: one hidden layer and 9 neurons, with an accuracy of 86.7%. Finally, with a identical topology, we obtain

Table 5. Results obtained with a Multilayer Perceptron Neural Network

		1 HL.			**2 HL.**			**3 HL.**		
#Neu.	**Epochs.**	Ped.	Tla.	Xal.	Ped.	Tla.	Xal.	Ped.	Tla.	Xal.
3	500	84.9	81.7	89.6	83.3	82.0	87.6	83.5	81.3	88.2
	1000	85.1	81.9	89.4	84.5	81.6	87.7	83.6	80.4	88.3
	2000	85.3	81.1	89.3	83.4	81.2	87.7	84.5	80.0	88.1
4	500	87.3	85.0	91.9	83.9	84.3	90.1	84.8	83.7	90.5
	1000	86.7	85.2	91.8	83.6	84.1	90.3	84.1	83.3	90.9
	2000	87.1	85.1	92.0	84.3	84.4	91.8	84.7	83.5	90.7
5	500	87.3	85.2	92.1	84.9	85.6	90.7	83.8	84.6	90.9
	1000	86.6	85.1	92.0	84.9	84.6	91.2	84.5	84.4	90.9
	2000	86.4	85.0	91.9	84.8	84.8	91.0	84.7	85.1	90.7
8	500	88.7	85.6	93.2	86.5	85.1	92.2	85.3	84.2	91.4
	1000	87.7	85.1	93.2	87.3	85.7	91.8	85.0	85.0	90.7
	2000	85.0	85.5	93.2	86.0	85.4	91.8	85.4	83.9	91.4
9	500	87.6	86.7	93.6	86.5	85.3	91.9	84.2	85.2	91.7
	1000	88.6	86.4	93.5	86.7	84.3	92.1	84.5	84.7	92.1
	2000	88.0	86.1	93.3	85.9	85.4	92.2	84.9	86.0	91.9
Avg.	500	86.4	85.0	92.6	86.2	85.1	91.5	82.9	84.5	91.1

Table 6. Results with the C4.5 and the Random Forest algorithms

Algorithm	Ped.	Tla.	Xal
C4.5	91.6	88.2	93.3
Random Forest	92.3	89.6	94.4

an accuracy of 85.3% for the case of Xalostoc station. It is evident that those results are consistent for all cases.

On the other hand, the best results obtained with decision trees were with the Random Forest algorithm, as is shown in Table 6. The accuracy achieved in these experiments is 92.3% for Pedregal station, 89.6% for Tlalnepantla, and 94.4% for Xalostoc. However, based on our experiments, the global accuracy of C4.5 was very similar to the Random Forest algorithm.

Remember that those results were obtained from data separated by seasons. These models can be used as a reference for predicting the behavior of air pollution with respect to seasons, but if we can predict in other timescales, then are necessary more experiments.

In this way, we proposed organize data per hour in order to have more precision at the moment to predict the pollution per day. Hence, we generate new balanced data samples from the preprocessed dataset. In those new experiments, we used again the decision trees algorithms, and the multilayer perceptron neural network, but with a topology with one hidden layer only. This decision is based on the previous results, because with more hidden layers did not improve the final accuracy.

Table 7. Results computed with the C4.5, the Random Forest and the Multilayer Perceptron algorithms, the data are obtained hour by hour

	C4.5.			Random Forest			MLP. 8 Neu.			MLP. 9 Neu.		
HOUR	PED	TLA	XAL	PED	TLA	XAL	PED	TLA	XAL	PED	TLA	XAL
1	99.73	99.86	99.73	99.59	99.86	99.59	99.73	99.86	99.73	99.73	99.86	99.73
2	99.86	100	99.86	99.86	100	99.86	99.86	100	99.86	99.86	100	99.86
3	99.86	100	99.86	99.86	100	99.86	99.86	100	99.86	99.86	100	99.86
4	100	100	100	100	100	100	100	100	100	100	100	100
5	100	100	100	100	100	100	100	100	100	100	100	100
6	100	100	100	100	100	100	100	100	100	100	100	100
7	100	100	100	100	100	100	100	100	100	100	100	100
8	100	100	100	100	100	100	100	100	100	100	100	100
9	100	100	100	100	100	100	100	100	100	100	100	100
10	98.35	99.45	98.08	97.81	99.45	97.81	97.81	99.45	97.67	97.94	99.45	97.53
11	89.44	90.12	87.52	88.75	90.26	88.75	90.12	91.08	89.85	89.44	90.95	89.30
12	75.31	80.52	75.58	76.95	80.11	76.95	76.95	82.58	75.31	76.13	82.17	75.99
13	66.80	70.51	65.98	70.92	70.10	70.92	69.68	72.02	72.57	70.10	71.47	72.98
14	71.06	69.96	68.86	69.96	69.14	69.96	71.88	71.47	68.18	74.49	70.10	70.10
15	70.78	73.39	67.76	73.39	71.60	73.39	69.68	73.25	69.14	70.23	72.70	69.41
16	67.22	73.25	70.64	72.57	71.74	72.57	70.42	73.48	70.10	71.74	72.66	71.10
17	73.66	78.19	74.35	76.54	73.25	76.54	73.39	75.03	75.99	72.98	73.94	76.82
18	74.35	81.62	85.60	86.69	80.38	86.69	81.62	81.48	84.64	80.25	82.44	85.46
19	89.44	94.79	94.79	95.47	94.51	95.47	88.20	93.96	94.24	88.48	94.10	93.83
20	93.83	99.04	97.39	97.39	99.04	97.39	94.10	99.04	96.43	94.10	99.04	96.57
21	98.22	99.45	98.77	98.63	99.35	98.63	98.08	99.45	98.77	98.08	99.45	98.77
22	98.90	100	99.59	99.59	100	99.59	98.90	100	99.59	98.90	100	99.59
23	99.73	100	99.73	99.73	100	99.73	99.73	100	99.73	99.73	100	99.73
24	99.73	100	99.73	99.73	100	99.73	99.73	100	99.73	99.73	100	99.73

In Table 7 are shown the results. We observed in these new experiments that in the early hours and the last hours of each day, the accuracy achieved by the algorithms are very high (close to 100%). However, for noon the accuracy sensible down, around 66%. These results are comprehensible: on one side, in hours between 12 am and 7 am, the human activity is low, then many factories, contamination produced by vehicles, and other factors are low. Because of this, in this lapse of time the air pollution is low, and then the accuracy obtained by the classifiers is high. After that, many human activities increase considerably, and then many contaminants are released into the atmosphere. Generally, it is required a period of 2 or 3 hours for pollutants build up in the atmosphere. Result of the above, between 12 pm and 5 pm, the air pollution is highly variable, depending of many factors, including natural factors (atmospheric conditions), and related with human activities. Because of this, the accuracy down sensible.

Finally, the last results suggest that the behavior of air pollution is moving in three stages. Because of this, we proposed new experiments where data is clustered in blocks of 8 hours: 12 a.m. to 8:59 a.m., 9 a.m. to 5:59 p.m., and 6 p.m. to 11:59 p.m. The results obtained in our experiments are shown in Table 8.

Table 8. Results obtained with data clustered in periods of 8 hours

	C4.5.			Random Forest.			MLP. 9 Neu.		
Period	PED	TLA	XAL	PED	TLA	XAL	PED	TLA	XAL
1-8 HRS	99.93	99.98	99.93	99.93	99.98	99.93	99.93	99.98	99.93
9-17HRS	80.09	81.35	80.68	81.84	82.51	82.90	81.17	82.20	81.38
18-24HRS	93.62	96.58	96.65	94.46	96.46	96.77	94.31	96.50	96.61

If we compare the results obtained in Table 7 and Table 8, we can see that the accuracy obtained for noon is increased. This mean that the new distribution of data is favorable for the classifiers, and then they can predict with more precision the air pollution. In these experiments we can see that the accuracy obtained between 9 a.m. and 17:59 p.m. is above to 80%, and the accuracy for the others periods is very high. The best results are obtained with the Random Forest algorithm, but are very close between them.

Another interesting result that can be seen from the results is the fact that the accuracy reported by the two different models (neural networks and decision trees) is very similar. As conclusion of this, it is possible to use any of the two models.

6 Conclusions and Future Work

In this research we explored models based on neural networks and decision trees capable to predict levels of ozone as an air pollutant.

We explored three different ways of building classification models for each meteorological station: the first one based on seasons observations, the second one with a observation hour by hour, and the third one with an observation in periods of 8 hours per day. In general and based on our results, we observed that the multilayer perceptron neural network and algorithms such as C4.5 and Random Forest are capable to predict the ozone with a similar accuracy.

However, it is possible to conclude that the third model, where data was organized in clusters of 8 hours, let us to build predicts with a very good accuracy. In this case, both neural networks or decision trees can be used to predict the ozone levels.

We know that there are many variables that were not considered in this study, and that may influence for the ozone levels in Mexico city, such as volcanic ash emitted by the Popocatepetl (an active volcano located 60 km southeast of Mexico city), solar radiation, particles floating into the air with less than 10 micrograms, pluvial precipitations, between others. As future work, we can include this variables, with the aim to increase the global accuracy. Moreover, it is possible to explore others approaches for predicting the ozone levels, such as support vector machines and bayesian networks.

References

1. Reyes, H., Vaquera, H., Villaseñor, J.: Estimate of tendencies in high levels of urban ozone using the quantiles of the distribution of generalized extreme values (gev) (2007) (en revisin)
2. Simat, D.: Biblioteca Virtual, México (2012), http://www.sma.df.gob.mx/simat
3. Mark Hall, Eibe Frank, G.H.B.P.P.R.I.H.W.: The weka data mining software: An update; sigkdd explorations. Master's thesis, Tesis de Maestría en Ingeniería Ambiental-UNAM, vol.11(1) (2012)
4. Seinfeld, J.: Comittee on tropospheric ozone formation and measurement; Board on Environmente Studies and Toxicology; Board on Atmospheric Sciences and Climate; Commission on Geosciencies, Environment and Resources; National Research Council, Rethinking the on ozone problem in urban and regional air Pollution. National Academic Press, Washington (1991)
5. Bravo, H.: La contaminación atmosférica por ozono en la zona metropolitana de la ciudad de méxico: evolución histórica y perspectivas. IX Comisión Nacional de los Derechos Humanos (1992)
6. I.N.E.G.I.: Estadísticas del Medio Ambiente. Semarnap, México (1999)
7. Molina, L.T.: The impacts of magacities on air pollution, environmetal aspects of urbanization, Goteborg Sweden (2004)
8. Garfias, M., Audry, J., Garfias, F.: Ozone trend analysis at pedregal station in the metropolitan area of mexico city. J. Mex. Chem. Soc. 49(4), 322–323 (2005)
9. Barron, E.A.A.A.L.J.R.: A sistem for forecast of the maximun ozone levels. Atmospheric Environment 38, 4689–4699 (2004)
10. Barai, S., Dikshit, A., Sharma, S.: Neural Network Models for Air Quality Prediction: A Comparative Study. In: Saad, A., Dahal, K., Sarfraz, M., Roy, R. (eds.) Soft Computing in Industrial Applications. ASC, vol. 39, pp. 290–305. Springer, Heidelberg (2007)
11. D.F., G.: Gaceta Oficial No. 129, pp. 1–14 (2006)
12. Proyect, R.: Biblioteca Virtual, USA (2012), http://www.r-project.org/

Automatic Discovery of Web Content Related to IT in the Mexican Internet Based on Supervised Classifiers

José-Lázaro Martínez-Rodríguez, Víctor-Jesús Sosa-Sosa, and Iván López-Arévalo

CINVESTAV IPN, Information Technology Laboratory at Technologic and Scientific
Park TECNOTAM – Km. 5.5 highway Cd. Victoria-Soto La Marina zip code 87130
Cd. Victoria, Tamps., México
{lmartinez,vjsosa,ilopez}@tamps.cinvestav.mx

Abstract. General web search engines, such as Google, Yahoo and Bing have been very successful information retrieval tools. However, many users with domain-specific interests are still disappointed with the responses obtained from these generic tools. This situation has motivated the creation of domain-specific search engines because they are able to offer increased accuracy with a minor maintenance and infrastructure cost. This paper introduces a method to discover domain-specific web content delimited by a country-context. This method allows a search engine to improve its accuracy for users that are interested in a domain-specific web content from a particular country. Our method is based on supervised classifiers and define country bounds for the search. To delimit the country context, our web content extraction process takes information from different sources, such as the Unified Resource locators (URLs), official government web pages, the Network Information Center (NIC) and the IP numbers reserved to the country of interest. Details of the system architecture are presented. A proof of concept was carried out using the Information and Communication Technologies (ICT) domain in the Mexican context. The testing prototype has obtained encouraging results.

Keywords: Web document classification, web seed obtainment, web search engine.

1 Introduction

The available amount of information in the web shows a constantly growth. For example, in the case of static web pages, studies of 1998 found 200 million in [7] and more than 320 millions in [14]. The next year, a study reported the existence of 800 millions [15]. For 2005 the reported web pages reached 11.5 billion [8]. Nowadays, it is estimated that there are between 30 and 40 billion of web pages and the growth continues[1] .

[1] http://www.worldwidewebsize.com/

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 107–118, 2013.

This web content growth becomes a difficult challenge when a generic search engine is trying to fulfill domain-specific user expectations. This situation was the main motivation to develop a method for building domain-specific search engines delimited by a country-context. To validate our method, a domain-specific search engine was developed based on the Information and Communication Technology (ICT) testing domain, which was delimited by the Mexican Web context. We defined the Mexican context as all of the web pages (and their embedded Web items) whose URLs contain the MX as the Top Level Domain (TLD) or are hosted in a Web server that uses an IP reserved to the Mexican Internet.

Due to the Web immensity, a low scale web sample was needed for testing purposes. A representative group of Mexican web sites (more than 100,000 web sites) was selected. According to [19], country selection as a web search space represents a good balance among completeness and diversity.

This paper is composed of the following sections; Section 2 presents system architecture, which is based on an open source web search engine architecture. In section 3 tools and libraries used for the implementation of the solution are presented. Section 4 shows the preliminary results. Conclusions and future works are included in section 5.

2 Description of the Solution

Our system architecture is very similar to many search engine architectures. The main components are depicted in figure 1.

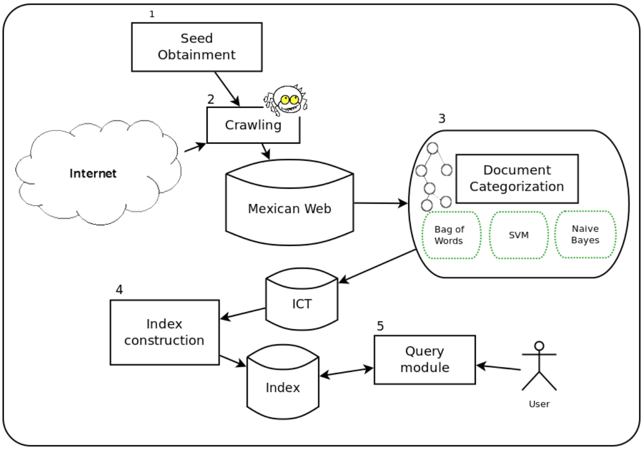


Fig. 1. System architecture

Our proposed method contains different steps that can be implemented as components in a typical search engine system architecture (figure 1) and are described as follows:

1 Seed obtainment. Seeds are the initial URLs set with which the crawl process is going to begin the content downloading, after that, the next URLs are extracted from the downloaded elements. This step is only in charge of collecting the seeds within the Mexican Web context.

2 Crawling module. It is responsible for the web page acquisition (in this case the Mexican web) to increasingly feed a repository. Initially, it will retrieve the seed URL and subsequently take the URL contained in the downloaded information until depth four. It means, the seed URL represents a Web document that is at level one, links embedded in this domain represent documents at level two and so on.

3 Document classification. This module is intended to do a document categorization of the Mexican Web repository. This process is required due to the interest in separating documents by domain (ICT).

4 Index construction. The purpose of this module is the creation of the index which is mainly based on inverted list structures. This specialized index is enriched with information taken from the previous document classification. The main aim is to response domain-specific queries.

5 Query module. Finally, we have the module responsible of the query reception. It will take online queries from users and will format the response with basic elements of information like title and excerpts of web documents.

3 Implementation

Based on the system architecture depicted in figure 1, this section offers more details about the implementation of every step considered in our methodology and included in the search engine modules.

3.1 Seed Obtainment

We propose a way to obtain as much as possible web content belonging to the Mexican context. In this module the following processes are implemented:

1 Country-specific IPs Collection. This is a revision of the IP database reserved to Mexico. There are some ways to know at which place an IP address belong to, e.g. using Geo Ip Location software like [1] and asking for those IPs in the Mexican Database. A second option is visiting some site that provides all the IPs block from a country, such as those mentioned in [2] and [3]. We found 25,862,912 IP addresses assigned to México. Whit this process, we gathered the IPs from Mexico that could potentially point to web servers containing Mexican web content.

2 Web servers detection. At this step, we look for active web servers that are hosted using an IP from the Mexican IP database. A HEAD HTTP request is sent to every IP assigned to the Mexican domain (polling process). Only those IPs responding with an HTTP 200 code (successful operation) are selected for the next step.

3 Name Discovery. In order to detect the domain names hosted in those IPs with a successful response (code 200 OK) in the above process, the next step is to get the reverse IP. A DNS service translates the name of a web site into an IP address, IP reverse is the opposite, try to obtain all hosted domains for one IP address. This was implemented using a Java API called DNSJava [4] which provides methods for obtaining the requested information.

4 Crawler initialization. The last step is to inject into the crawler all the URLs found in the above step.

3.2 Crawling

The software agent for information acquisition is the crawler. Its primordial task is to iteratively download web documents using the HTTP protocol, and store them in disk for a posterior analysis. Our crawler implementation is based on Nutch[13], which is an open source web crawler developed by the Apache Software Foundation and could be used at local or global scale.

The main goal of Nutch is to develop an open alternative to commercial web searchers (Google, Yahoo or Bing). Nutch allows developers to understand the way each module is constructed. Its architecture is based on the Hadoop Platform [2], which adds the distributed environment support.

For a complete web crawling, Nutch users recommend to create a bash script, in this way, we can save the state of the crawler if an error occurred. The constructed script is based on the next algorithm:

Algorithm 1. Whole Web crawling script

```
Require: A text file with one URL per line. (seeds)
while $seed \neq null$ do
    inject 10 pages to the frontier.
    depth=1
    while depth<=4 do
        Generate segment
        Fetch segment
        Update frontier
        Create out-links
        depth++
    end while
end while
```

Where:

- Seeds are a list of URLs that are read from a document until the end (one url by line)
- A depth level is consumed at each loop.

[2] http://hadoop.apache.org/

- "Generate segment" is a process that makes a structure from the frontier [3] in which a set of pages is prepared for being downloaded and parsed (obtaining plain text and links).
- "fetch segment" is a process that downloads pages from the created segment.
- "Update frontier". Is a process that updates the frontier with the new URLs embedded in the downloaded web documents.
- "Create out-links". Out-links are links from one site to another. This process is useful for the indexing process
- depth++ increases actual crawling level

Some pitfalls along the crawling process, are: Overcrowded domains. This problem arises from sites that automatically generate a lot of web documents. This problem can be avoided by using a procedure that takes into account a maximum number of web documents that would be feasible to extract from the same internet domain. Baeza and Castillo [6] present a characterization of national Web domains, they show a statistic saying that sites are composed of 52 to 549 web pages and each domain has an average between 1.1 and 2.5 sites, domains exceeding 10 sites probably contain spam content. For this prototype, it was defined a threshold of 5000 web documents, we thought this is enough for getting many relevant documents from one site. Once this threshold is reached, that internet domain is blocked.

3.3 Document Categorization

Document categorization (sometimes called text classification) is a task that arises from a specific document retrieval necessity that involves a huge amount of information. Classification is a very useful process in domain-specific search tasks. For that reason, documents were classified according to a class. Each document can be in zero, one or more categories or classes. The process to insert documents in classes is known as document categorization.

Formally, document categorization can be defined as follows:

Given a document collection D and a set $C = \{c_1, c_2, ..., c_L\}$ of L classes with their respective labels, a document categorizer is a binary function $F : DxC \rightarrow \{0, 1\}$, e.g. a function assigns a value of 0 or 1 to each pair $[d_j, c_p]$, such that $d_j \in D \; and \; c_p \in C$. If the assigned value is 1, it can be said that document d_j is a member from the class c_p.

When a classification like this is implemented, it is necessary to consider some metrics to evaluate the performance of the classifier. In this implementation the following three metrics were used: Precision, recall and F1 measure.

In our prototype, two relevant classification algorithms were evaluated: Naïve Bayes Multinomial and Support Vector Machines. According to the literature (as in [16,12,11]), these algorithms present high accuracy in documents categorization.

[3] A frontier is a data structure that contains the list of URLs to be visited.

Naïve Bayes Multinomial (NBM). It is a probabilistic approach [9] that is fast, precise and simple. Naïve Bayes allows to decide, among a set of possible categories, the correct one for a given document. One advantage of using a Bayesian classifier is its language independence, due to the use of probabilistic methods. We can use the Equation 1.

$$P(C_i|D) = \frac{(P(D|C_i) * P(C_i))}{P(D)} \tag{1}$$

Where:

P(C$_i$ | D): it is the probability of class C$_i$ given a document D.

P(D| C$_i$) is the probability of words in D to appear in the C$_i$ category.

P(C$_i$) is the probability of category C$_i$. P(D) is the probability of a document. We can notice that P(D) is a constance divider, so we can omit it.

D is divided into the set of words belonging to it {W$_0$,...,W$_{m-1}$}. For each word in C$_i$ is needed the probability product to calculate P(D| C$_i$), as in Equation 2.

$$P(D|C_i) = \prod_{t=1}^{|V|} P(W_t|C_i) \tag{2}$$

|V| is the vocabulary size.

Additionally, the Naïve Bayes Multinomial consider the frequency of apparition for every term in documents x_t instead a binary occurrence:

$$\prod_{t=1}^{|V|} P(W_t|C_i)^{x_t} \tag{3}$$

P(W$_t$| C$_i$) is calculated based on the number of apparitions for each term W$_t$ in a class C$_i$. To avoid probability with zero value, a Laplace estimation is used.

$$P(W_t|C_i) = \frac{1 + n(W_t, C_i)}{|V| + n(C_i)} \tag{4}$$

Where $n(W_t, C_i)$ is the occurrence number of W_t in C_i, |V| is the vocabulary size and $n(C_i)$ is the total word count in C$_i$.

Then we need an argument that maximizes de function, as in Equation 5:

$$C * (D) = argmax_{Ci} \prod_{t=1}^{|V|} P(W_t|C_i)^{x_t} \tag{5}$$

Support Vector Machines (SVM). SVM was introduced in [20] and comprises a set of supervised learning methods for data analysis and pattern recognition. A SVM model is a representation of instances as dots in space; such categories are separated by a gap as huge as possible. Then, new examples are mapped into the same space and the category is predicted based on the nearest gap.

Classification Task

The classification was conducted using WEKA [10], which is a free data mining tool which implements algorithms for classification, regression, clustering and visualization. It also contains tools for preprocessing information. In this work, Spanish language is used, altough changing the training files another language could be considered.

Data Preprocessing

Before classification training, a few steps are needed for information preparation, because the attributes are mainly textual and the required is numeric. Below are the carried out steps for data cleansing and preparation

- Text cleansing. Each document in the data set is processed to remove the digits and punctuation marks (0-9#$), because they do not provide representative information to categories.
- Information Load. Training information is supposed to be in text files in one directory with sub-directories; each one with information for the represented category. For the loading task, we chose a filter called "TextDirectoryLoader", which create an ARFF[4] file from the information contained in the supplied directory. The file is created with two attributes: text (file content) and class (document's class).
- Numeric Values Transformation. Numeric information is needed because the nature of the algorithms do not allow to work with textual information. For that reason, it was used a filter called "StringToWordVector". With this filter, Weka allows us to apply some other operations like:
  - TF-IDF (Term Frequency-Inverse Document Frequency) for weight calculation of words inside documents.
  - Stemmer. Which allow words reduction to their root form. It is useful because a word could have so many variations, e.g. ("stemmer", "stemming", "stemmed" as based on "stem").
  - Stopword elimination. Stopwords are words with low importance due to their high apparition frequency. In this list we added some other terms that frequently appear in web pages (sometimes by errors) like: "flash", "javascript", "twitter" among others.
  - Tokenizer. This process allow the division of a sentence in propositional syntagmas.

  After the transformation, the number of attributes increase, keeping the number of distinct words in the training set. It is important to say that new documents need to be processed with the same quantity of attributes from the training set, for that reason, WEKA constructs a dictionary in which the filter is supported.
- Feature Selection. It is based on an information gain filter, we select different quantity of attributes.

[4] Attribute Relation File Format. Type of file format required by WEKA.

– Training. The classifier is trained with the filtered information. WEKA has the implementation of the Bayesian algorithm but in the SVM case, it is used a wrapper for the LibSVM [5] class.

3.4 Index Construction

For the index construction, we used the SOLR [18] tool, whose core is constituted by Lucene [17]. Lucene is an open source library that offers indexing functions and search information management. SOLR provides routines for an easy data import and definition for the Lucene use. SOLR also provides filtering tools like synonyms replacement, tokenization, stopwords elimination among others.

Nutch indexes the web content by using a special document structure that follows a previously customized document scheme. e.g. if the schema had three fields: title, content and url, nutch prepares a document with those elements, extracting their information from the downloaded segments. We have modified the Nutch indexer to include a document classification task before the indexing process. It means, before the information is passed to SOLR, a multivalue field was added to the schema (thought to store the class and the roles) and a modified class called SolrWriter to train the classifier and add a new field with returned information from classification.

3.5 Query Module

It was developed a Java Server page (JSP) application which interacts with the SOLR server. It consists of an HTML form that receives the user query and through the SOLRJ API sends a query to SOLR. If there exists an XML response, this message was adapted into an HTML output page.

4 Preliminary Results

In this section are presented the obtained results of experiments about document classification and seed extraction.

Three evaluation measures were used for the classification results: Precision, recall and F1. In figure 2 is depicted the relevance of the information retrieved from a document collection.

Where:

Recall is the relevant document proportion that has been retrieved

$$Recall(R) = \frac{relevant\ documents\ (B) \cap\ retrieved\ documents(C)}{relevant\ documents(B)}$$

Precision is the relevant retrieved documents fraction

$$Precision(P) = \frac{relevant\ documents\ (B) \cap\ retrieved\ documents(C)}{retrieved\ documents(C)}$$

The first evaluation was based on document classification using two algorithms: NBM and SVM. Experiments were conducted using a corpus of 500

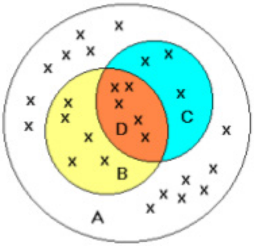


Fig. 2. Documents collection: A: all, B: relevant, C: retrieved, D: relevant retrieved

documents in spanish. Documents were extracted using the Nutch crawling with manually selected seeds and documents for training were selected by an expert to determine if they belong to the ICT domain.

During training phase, validation was achieved using a 10-fold cross validation technique [21], where a document set was divided randomly into n folds, each fold is retained once and the categorizer is trained with the other folds (n-1); then error is evaluated. In this way, the learning process is executed n times with a different training set. Finally, results are averaged.

We prepared results using different quantity of attributes 17379 (all attribute), 10000, 5000, 3000 and 1000, selected by the information gain criterion according to [22].

Table 1. Support Vector Machines results

			SVM			
Mea/N	1	2	3	4	5	AVG
Prec	0.860	0.894	0.867	0.860	0.866	0.869
Rec	0.859	0.892	0.846	0.835	0.846	0.855
F1	**0.859**	**0.892**	0.845	0.834	0.845	0.848
CCI	**85.865**	**89.240**	84.599	83.544	84.599	85.569
ICI	14.14	10.759	15.400	16.455	15.400	14.430

In table 1, Mea indicates a type of metric to evaluate the algortihms, i.e. Prec: Precision, Rec: Recall, CCI (Correctly Classified Instances), ICI (Incorrectly Classified Instances). N is the test number, test 1 used all of the attributes obtained from documents, i.e. 17,379, test 2 used 10000, test 3 used 5000, test 4 evaluated with 3000 and 5 used 1000 attributes.

In table 1, SVM presents better results when no attribute reduction is applied, because SVM can deal with few irrelevant features and high dimensional input spaces.

In table 2, the last three tests using NBM show better results than those shown by SVM. This is due to the impact of the feature selection algorithm, because NBM considers word frequencies and when including less usual words from the training set the result tends to be lower.

Table 2. Naïve Bayes Multinomial results

NBM						
Mea/N	1	2	3	4	5	AVG
Prec	0.844	0.894	0.881	0.894	0.91	0.884
Rec	0.831	0.884	0.852	0.878	0.909	0.870
F1	0.829	**0.883**	0.848	0.876	**0.909**	0.869
CCI	83.122	**88.396**	85.232	87.763	**90.928**	87.08
ICI	16.877	11.603	14.767	12.236	9.071	12.92

To show the capacity of the proposed methodology, we indexed 274,200 web pages obtained with the Nutch crawler within the Mexican Web context. The estimated time to index with the classification step was 42.3 minutes.

Methodology performance is measured by the three metrics mentioned before using five query results against two search engine versions: one implementing our proposed methodology and another using the current version of the SOLR search engine. The accomplished queries are : "software sales", "web programming", "ICT news", "Information Technology consultancy" and "Information technologies enterprise". Enumering the queries in this order, results are presented in three distinct quantities using a sample of 30 elements, first with 10, second 20 and third 30 results retrieved for each query with their respectives meassures.

Table 3 depicts the achieved results

Table 3. Generic vs Specific-domain search engine. Prec:Precision, Rec:Recall, F1:Meassure F1.

		General-SE			Domain-Oriented		
query	# results	Prec	Rec	F1	Prec	Rec	F1
1	10	.600	.750	.667	.600	.462	.522
1	20	.350	.875	.500	.500	.769	.606
1	30	.267	1.00	.421	.433	1.00	.605
2	10	.700	.636	.667	.600	.375	.462
2	20	.450	.818	.581	.550	.688	.611
2	30	.367	1.00	.537	.533	1.00	.696
3	10	.400	.364	.381	.600	.375	.462
3	20	.450	.818	.581	.550	.688	.611
3	30	.367	1.00	.537	.533	1.00	.696
4	10	.500	.556	.526	.600	.545	.571
4	20	.450	1.00	.621	.450	.818	.581
4	30	.300	1.00	.462	.367	1.00	.537
5	10	.900	.450	.600	.900	.360	.514
5	20	.400	.400	.400	.950	.760	.844
5	30	.667	1.00	.800	.833	1.00	.909

Results show that using a domain-oriented search engine delimited by the Web of a country achieves better results regard a general search engine when punctual information is required.

5 Conclusion and Future Works

This paper has shown a method for discovery of domain-specific web content in a country context that improves the accuracy of web search engines. We consider all stages of a general web search engine and create an adaptation of the Nutch crawler for the domain-specific content and the country context bounding.

Implementation of special mechanisms involving IP data bases usage by country, DNS reverse search and URL filter process have shown a great utility to delimitate Web sites within Mexican Context.

A process to index contents was achieved based on a NBM classification from spanish web contents extracted through the Nutch crawling, this showed the flexibility to modify Nutch and adjust it to our necessity. Concluding the web search architecture, a module for query management using JSP and the SOLRJ API was developed. Although the methodology presented good results at query time, classification could be improved selecting false positive/negative documents and including them into the corpus.

As a future work we pretend to increase the document corpus to be able of considering a second document classification where roles are considered. A role is the activity performed by a document in relation to the domain, e.g. education, software development, news, etc.

References

1. Geo Ip locator, `http://www.maxmind.com/` (last visit June 2012)
2. Country IP blocks, `http://www.countryipblocks.net/country-blocks/select-formats/` (last visit June 2012)
3. Ip info DB, `http://ipinfodb.com/` (last visit June 2012)
4. API DNSjava, `http://www.dnsjava.org/download` (last visit April 2012)
5. A library for Support Vector Machines, `http://www.dnsjava.org/download` (last visit June 2012)
6. Baeza-Yates, R., Castillo, C., Efthimiadis, E.N.: Characterization of national web domains. ACM Trans. Internet Technol. 7 (May 2007), `http://doi.acm.org/10.1145/1239971.1239973`
7. Bharat, K., Broder, A.: A technique for measuring the relative size and overlap of public web search engines. Comput. Netw. ISDN Syst. 30(1-7), 379–388 (1998), `http://dx.doi.org/10.1016/S0169-75529800127-5`
8. Gulli, A., Signorini, A.: The indexable web is more than 11.5 billion pages. In: Special Interest Tracks and Posters of the 14th International Conference on World Wide Web, WWW 2005, pp. 902–903. ACM, New York (2005), `http://doi.acm.org/10.1145/1062745.1062789`
9. Hadi, W.M., Salam, M., Al-Widian, J.A.: Performance of nb and svm classifiers in islamic arabic data. In: Proceedings of the 1st International Conference on Intelligent Semantic Web-Services and Applications, ISWSA 2010, pp. 14:1–14:6. ACM, New York (2010), `http://doi.acm.org/10.1145/1874590.1874604`
10. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. SIGKDD Explor. Newsl. 11(1), 10–18 (2009), `http://doi.acm.org/10.1145/1656274.1656278`

11. Husby, S., Barbosa, D.: Topic Classification of Blog Posts Using Distant Supervision. In: Proceedings of the Workshop on Semantic Analysis in Social Media. Association for Computational Linguistics, Avignon, France, pp. 28–36 (2012), http://www.aclweb.org/anthology-new/W/W12/#0600
12. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features (1998)
13. Khare, R., Cutting, D.: Nutch: A flexible and scalable open-source web search engine. Tech. rep. (2004)
14. Lawrence, S., Giles, C.: Searching the world wide web. Science 280(5360), 98 (1998)
15. Lawrence, S., Giles, C.: Accessibility of information on the web. Nature 400, 107–109 (1999)
16. Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., Watkins, C.: Text classification using string kernels. J. Mach. Learn. Res. 2, 419–444 (2002), http://dx.doi.org/10.1162/153244302760200687
17. McCandless, M., Hatcher, E., Gospodnetic, O.: Lucene in Action, Covers Apache Lucene 3.0, 2nd edn. Manning Publications Co., Greenwich (2010)
18. Smiley, D., Pugh, E.: Apache Solr 3 Enterprise Search Server. Packt Publishing, Limited (2011), http://books.google.com.mx/books?id=ChKVwotW8mYC
19. Tolosa, G., Bordignon, F., Baeza-Yates, R., Castillo, C.: Characterization of the argentinian web. Cybermetrics 11(1), 3 (2007), http://www.cindoc.csic.es/cybermetrics/articles/v11i1p3.html
20. Vapnik, V.N.: The nature of statistical learning theory. Springer-Verlag New York, Inc., New York (1995)
21. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers Inc., San Francisco (2005)
22. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: Proceedings of the Fourteenth International Conference on Machine Learning, ICML 1997, pp. 412–420. Morgan Kaufmann Publishers Inc., San Francisco (1997), http://dl.acm.org/citation.cfm?id=645526.657137

Automatic Monitoring the Content of Audio Broadcasted by Internet Radio Stations

Omar Nuñez and Antonio Camarena-Ibarrola

Universidad Michoacana de
San Nicolas de Hidalgo
onunez@faraday.fie.umich.mx, camarena@umich.mx

Abstract. Auditing the content of audio as transmitted by radio-stations is of great interest for governments, publicists, and for managers of radio-stations among others. Our approach consists of making use of a robust audio-fingerprint for characterization of the monitored audio and a proximity index for fast search of the most similar piece of audio among the collection of audio known to the system (ads mainly). Since the audio signal as broadcasted via Internet suffers little degradation, an inverted index proved to be a great solution while pivot based indexes such as the Burkhard-Keller tree and the Fixed Query array turned out to be of no use for our purpose due to the curse of dimensionality. The implemented system performed really well having a 100% recall and it it fast enough to allow real time monitoring of several radio-stations simultaneously with a single desktop computer.

1 Introduction

Monitoring audio is the process of sweeping the audio-signal by segments to verify if those segments match any segment of audio known by the system in order to fetch its related meta-data. Monitoring Radio Broadcasts serves several purposes such as auditing audio marketing campaigns (advertisements), ranking popular songs, preventing banned content from being broadcasted. In Mexico monitoring the content of the audio broadcasted by radio stations is also important to verify that the coverage of political campaigns is fairly distributed among the political parties. There are several approaches to this task, such as reading attached meta-data or simply asking radio stations for broadcast reports, this solutions are cheap, but they can not be considered real audio analysis and are therefore not reliable [1]. Other approaches such as using human observers are widely used, while this approach does imply an audio analysis and it is really easy to set up, it will not scale well and will result into a very expensive and error-prone solution. A smarter and more attractive approach is the use of audio watermarks [2], with this approach, a reliable channel is built in order to send meta-data embedded in the audio signal to the receivers. A good watermark should not create any human-audible distortions to the audio as described in [3], such approach performs a real content analysis, but requires the audio-signal to be modified before it is broadcasted in order to add the watermark. Finally,

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 119–130, 2013.

with the use of audio-fingerprints a real audio analysis is able to be performed without any alteration of the audio-signal prior to transmission.

2 Our Approach

The audio-fingerprint extraction module performs digital processing of the signal in order to determine robust perceptual features from the audio, this means that such features would be found in the audio signal even after equalization, lossy compression or any other degradation commonly suffered by audio-signals that are broadcasted by radio stations. We use a Multi-Band Spectral Entropy based signature (MBSES) which was first proposed in [4] since it was found to be highly robust to signal degradations. In fact, MBSES has already been used for offline automatic monitoring of Radio Stations in [5] with a perfect recall, MBSES also provides us with the flexibility we need, we will further explain how this audio-fingerprint is extracted in an upcoming section. As for the database querying module, we organized the collection of known audio-fingerprints in a way that we are able to find the matching audio-fingerprint avoiding computation of the distances to each and every single audio-fingerprint in the collection, this approach allows for efficient database querying even with a large database of audio-fingerprints.

3 Multi-Band Spectral Entropy Signature

The Multi-Band Spectral Entropy Signature (MBSES) is used to characterize songs, advertisements, speech, or any audio-signal for identification purposes. The process of extracting the MBSES of an audio-signal may be detailed as follows:

1. The signal is processed by frames, we use a frame size of 240 milliseconds in order to provide enough time support for entropy determination. However, we need to ensure sufficient time resolution so that the evolution of the spectral entropy is adequately observed for robust characterization of the audio-signal, for this particular purpose, the frames are overlapped. We tested several overlapping percentages between frames. In the experimental section we will show our results regarding the optimization of this parameter.
2. To each frame the Hann window is applied, and then its Discrete Fourier Transform (DFT) is determined.
3. Shannon's entropy is determined from each one of the critical bands according to the Bark scale. We use only the bands that the frequency of the broadcasted audio covers. We monitored audio sampled at 22050 Hz which implies the highest frequency is 11025 Hz, then we use 22 of the 25 critical bands defined in [6]. To compute Shannon's entropy we make use of Equation 1

$$H = \log(2\pi e) + \frac{1}{2}\log(\sigma_{xx}\sigma_{yy} - \sigma_{xy}^2) \tag{1}$$

where $\sigma_{xx} = \sigma_x^2$ is the variance of the real part of the DFT, $\sigma_{yy} = \sigma_y^2$ is the variance of the imaginary part of the DFT, and $\sigma_{xy} = \sigma_{yx}$ is the covariance between the real and the imaginary parts of the DFT.

4. For each band we obtain the sign of the derivative of the entropy as in equation 2. The bit corresponding to band b and frame n of the MBSES is determined using the entropy values of frames n and $n-1$ of band b. Therefore, three bytes for each frame are needed to store the MBSES of an audio-signal. Binarization also adds robustness to the signature [4].

$$F(n,b) = \begin{cases} 1 \text{ if } [h_b(n) - h_b(n-1)] > 0 \\ 0 \ Otherwise \end{cases} \tag{2}$$

By extracting the MBSES of an audio-signal, we end up with a binary matrix, where columns represent frames and rows stand for critical bands.

We use the Hamming distance to compare our audio-fingerprints, this is another advantage of binarization.

4 Retrieval

Retrieval is the process of querying the database of audio-fingerprints of known audio (i.e. ads) looking for a match. The brute force approach or sequential searching compares the fingerprint of a segment of audio as it is received with every audio-fingerprint in the database. Such approach would result in a slow querying system for large databases. To solve this problem, we tried three proximity indexes; The BK-Tree [7]; The Fixed Query Array (FQA) proposed in [8]; and an inverted index or look up table similar to that used in [9]. The result of the first two proximity indexes was very disappointing due to the curse of dimensionality. Both the BK-Tree and the FQA are pivot based indexes. The members of the database grouped together in the same ring with respect to a specific pivot meaning no element could be discarded using the pivot in turn, the same happens with the other pivots and so we ended up having a "sequential scam" (which means using an index that is really sweeping throughout all of the elements). Fortunately, the inverted index worked like a charm with extremely low number of Hamming distances computed for query and allowing a perfect recall.

Approaches in [7] and [8] are pivot-based indexes which exploit the triangle inequality, further reading about pivot-based indexes and metric spaces can be found in [10] and [11]. We must have a database where all the elements (i.e. audio-fingerprints) are the same size, with that in mind, we clipped the ads to fixed sizes. We use three different sizes that we specify in the Experiments section. The construction of the BK-Tree is as follows, a range of distances between elements must be defined, this range will define the number of sons each node will have. To build the BK-Tree an arbitrary element of the database is chosen as the root, the next element to insert is compared to the root to determine the range the distance between them belongs to, the element is then made a descendant on the corresponding range. If there were already a descendant on that position, then we would compare it to such element as we did with the root. We repeat this process until there are no more elements to insert into the index. The nodes of the tree are the pivots. Retrieval with this approach is as follows; having defined a range in which we want to search, we compare our query to all the pivots which fall within this range, as if we were trying to insert our query, if the query is not found, the range should be extended. For Fixed Query Arrays, the construction is as follows; we pick up a number of pivots from the elements of the database, then compute the distance between every audio-fingerprint of the database and every pivot. The distances are stored with the audio-fingerprint they belong to. We order the elements with respect to the first pivot, if there are ties, they are broken using the rest of the pivots. Construction of this index then, can be as fast as the sorting algorithm used. Querying with FQAs is as follows: First compute the distance of the query to all the pivots, then discard the elements of the database that does not fall within the predefined range the distance corresponds to. Repeat this process for each pivot. Finally perform sequential search for the audio-fingerprints that were not discarded by any pivot.

In our experiments, we found out that the normalized hamming distances between the audio-fingerprints in the database and those selected as pivots were too similar, ranging between 0.47 and 0.53, independently of which audio-fingerprints were chosen as pivots. This is a problem since the search degenerates to a sequential search. This is due to the curse of dimensionality (CoD) or more recently called "concentration effect" in [12]. In order to avoid this, we decided to trim the query range, this was at the cost of recall rate as we describe in the Experiments section.

To use an inverted index similar to the "look up table" (LUT) proposed in [9], each row of the audio-fingerprint is used as a integer number which in turn will serves as indexes to the LUT (Please remember that the MBSES of an ad is indeed a binary matrix). We use 22 critical bands, then the rows are made of 22 bits , therefore we need a table of size $2^{22} = 4,193,304$. The construction of this table is as follows: For each row of the audio-fingerprint interpreted as an integer number, add an entry to the table. Each entry should at least have the position inside the audio-fingerprint in which it occurs, and a pointer to the audio file from which it was extracted (i.e a song or an ad). Each entry in the table is really a linked list. If there were a nonempty list for a specific entry on the table, the new frame must be added at the end of this list just as it would be done with a hash table. We must do this for each row of every audio-fingerprint of the database. In Figure 1(A) we depict the look up table that results after indexing the example files shown in Figure 1(B).

We now describe the query process: Since each row of an audio-fingerprint is used as an index to the LUT, we check the (possible empty) linked list pointed by the row. Then we go to the offset of the audio-fingerprint as indicated in the linked list and get a piece of the audio-fingerprint that is the same size of the query and compute the Hamming distance between this piece and the query. If the distance is below a threshold, then we report an occurrence, if not, we

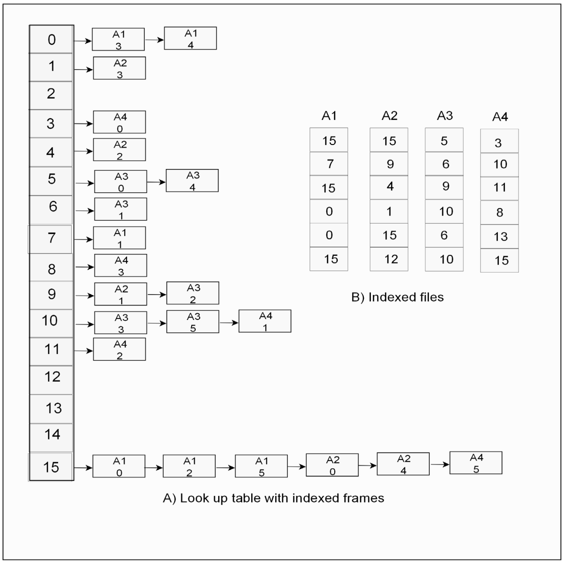


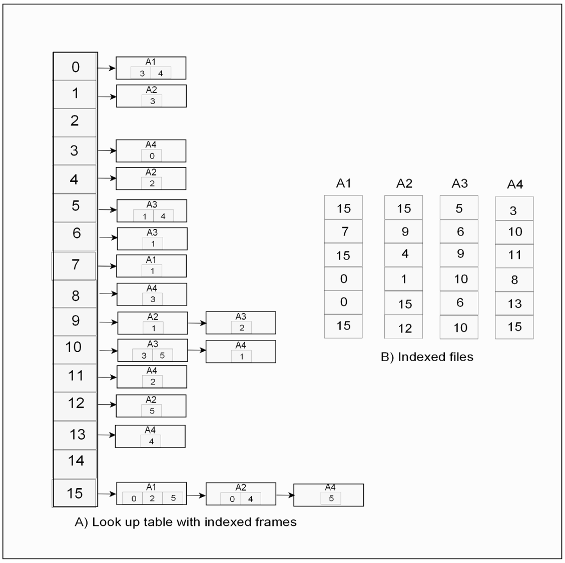
Fig. 1. (A) Look up table; (B) Related indexed files (i.e audio-fingerprints)

proceed to the next position read in the list. This is repeated until there are no more nodes on the list.

4.1 Modification to the Inverted Index

In order to save memory we modified the structure of the LUT, We grouped all occurrences on the linked lists that belong to the same audio-fingerprint (i.e. advertisement) into a single node of the list. Then we have a single node for each different row found in the audio-fingerprint, with an array of positions in each node instead of a node for each row of the audio-fingerprint. In Figure 2(A) we depict the Proposed modified inverted index and the related indexed files (i.e. audio-fingerprints) in Figure 2(B). Please compare Figure the inverted index shown in Figure 2(A) with the one shown in Figure 1(A) to see in this theoretic example how the linked lists reduce lengths and therefore save memory.

The modification of the LUT affects the way in which it is built, instead of simply adding each audio-fingerprint's row occurrence at the end of the linked list of its corresponding bucket in the LUT, a sequential search would have to be performed along the linked list. Such solution turned out to be an impractical one since it greatly increased the time needed to build the index as we will report later in the Experiments section. We came out with a simple solution to this problem. Again remember that an audio-fingerprint (the kind we are using, that is MBSES) is a binary matrix, the rows of this matrix might occur several times, this fact motivated the modification of the index aimed to save memory space. By sorting the rows of the audio-fingerprint we get a binary matrix where all



A1	A2	A3	A4
15	15	5	3
7	9	6	10
15	4	9	11
0	1	10	8
0	15	6	13
15	12	10	15

B) Indexed files

Fig. 2. (A) Modified Look up table; (B) Related indexed files (i.e audio-fingerprints)

occurrences of identical rows are grouped together, this way the sequential search we mentioned at the beginning of this paragraph is no longer needed. Sorting can be done efficiently using algorithms such as the merge sort. Of course, the original positions of the rows are kept since they have to be stored in the index. The improvement in building time was surprisingly good and is also reported in the Experiments section.

An additional slight improvement was obtained in querying time of the modified index with respect to the original one. This is due to the fact that the linked lists which have to be entirely swept are now shorter, the number of candidates are of course the same but sweeping arrays is faster than sweeping linked lists. In Figure 3 the required time for offline monitoring 23 hours is shown, the slight improvement of the modified LUT is easy to see.

4.2 Improving Querying Time by Relaxation of the Rules

At querying, Haitsma *et al* [9], verify each row of the query in the LUT, they do that hopping that at least one of the rows would occur exactly or with at most two bits changed (although they do check for three bits changed among the "weak bits"). Such approach might be affordable for Music Information Retrieval but not for Monitoring Radio Broadcasts where thousands of queries per minute are needed, especially if the system has to monitor several Radio stations simultaneously.

We then decided to make a variation to the original query scheme, instead of checking all the rows of the query on the table, we just check the first one. Such

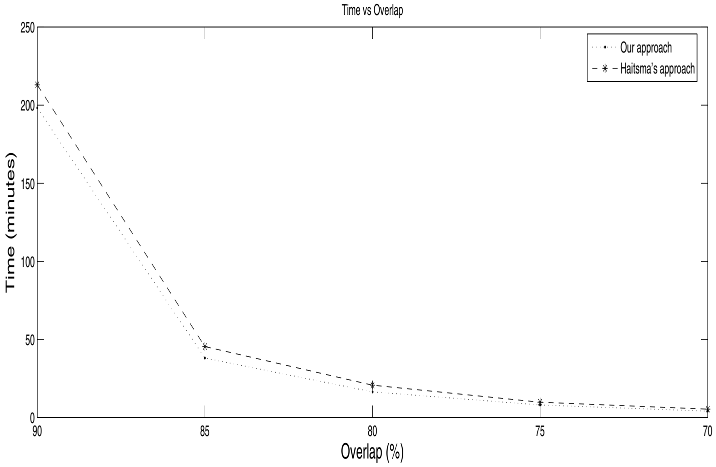


Fig. 3. Duration of the process of offline monitoring 23 Hours of Audio

variation works thanks to the robustness of the MBSES and the fact that we are monitoring radio stations via Internet where the D/A A/D deformation is avoided and the degradation is due only to lossy compression, equalization and time shift, this last degradation (time shift) by the way, depends greatly on the overlapping of the frames, the higher the overlapping, the lower the deformation due to time-shift.

5 Experiments

5.1 Offline Monitoring Results

We were concerned about the effect that overlapping had both in recall and in querying time, we see this parameter as a critical one. We tried several overlapping percentages between consecutive frames, specifically 95%, 90%, 85%, 80%, 75%, and 70%, these overlaps imply processing about 83, 42, 28, 21, 17, and 14 frames per second respectively. Our tests involve monitoring 23 hours of audio from a popular radio station where the occurrences of 83 ads were to be detected. In order to assess the system reliability we first used the "human observer" approach to monitor those 23 hours of audio and find the advertisement occurrences exhaustively, this manual work allowed us to verify that there is indeed degradation by equalization and amplitude (volume) changes, so Internet broadcasts are not degradation-free as one would expect. However, the chosen audio-fingerprint managed this degradations very well.

We first show the results of using just a small part of every ad for the monitoring. Our experiments include excerpts of 3, 5 and 7 seconds long. In our experiments we also vary the overlapping between frames so that we can arrive to some conclusions regarding both the optimal overlapping and excerpt size.

In the case of the BK-Tree based index, we use two rings, one of them to group audio-fingerprints whose normalized Hamming distance to the pivot falls in the range between 0.0 and 0.5 and the other one to group audio-fingerprints whose normalized Hamming distance to the pivot falls in the range between 0.5 and 1.0. Of course, this will result in an un-balanced binary tree. Remember we stated before that the normalized Hamming distances between audio-fingerprints according to our experiments fall in the range between 0.47 and 0.53, what we were trying to do here was for every pivot to discriminate half the audio-fingerprints of the collection.

For querying, we use search by range, but with a narrow range, that way we will just explore a small part of the tree at the cost of sometimes not finding the matching ad. This decision was motivated by the fact that increasing the range would imply an exhaustive search due to the curse of dimensionality as we discussed before, otherwise there would be no point in using a BK-Tree based proximity index.

The FQA based index did provide flexibility to choose a range, but we also had to limit the range to 2.5% (i.e. Normalized Hamming Distance of 0.025), since a bigger range would again imply searching through all the elements of the database, this range was the one that gave us the higher recall.

In Table 1, we show the time taken to review 23 Hours of the audio, also the recall rate, and the number of false positives using sequential search (brute force), a BK-Tree based index, and a FQA based index. In this experiment where short excerpts of 3 seconds taken from the middle part of each ad were used to locate occurrences of the ads we had some unexpected false positives. It is important to mention that false positives do not occur when we use whole ads (and not just a piece of it). We include only the highest and the lowest overlap for the sake of space, but as overlap is reduced, the recall rate decreases.

The BK-Tree based index is the worse of all in terms of recall rate. In this experiments we corroborated that it is not good idea to use excerpts of only 3 seconds taken from the middle of the ads to perform the review of the audio, this is because we get some false positives. Using 5 seconds, we obtained similar recall results, but we still get some false positives (although fewer). Finally, using 7 seconds is apparently the best decision of this approach using excerpts as we show in Table 2.

As we show in Table 2, by using excerpts of 7 seconds taken from the middle of each ad we managed to reduce to zero the number of false positives, nevertheless it does not enhance our recall rate. Furthermore, extracting the excerpts imply some additional work, which would be worth if we had good results, we however do not believe we have reached our goals since our aim is a much faster system and with a higher recall. This motivated the use of other alternatives and further testing.

The inverted index using a LUT as described in the Retrieval section was tested using whole ads (not just short excerpts). In Figure 4 we show the recall rate obtained as well as the time needed to complete the offline monitoring of 23 Hours of audio, as we stated before, these results depend on the overlapping

Table 1. Monitoring offline 23 Hours of audio using excerpts of ads of 3 seconds

Overlap	**Recall rate**	**False positives**	**Searching Time (mins)**
Sequential search			
95%	100%	16	88.00
70%	100%	12	5.5
B-K Tree			
95%	93%	10	19.98
70%	41%	10	0.30
Fixed Query Array			
95%	100%	12	45.25
70%	97%	2	1.32

Table 2. Monitoring offline 23 Hours of audio using excerpts of ads of 7 seconds

Overlap	**Recall rate**	**False positives**	**Time taken (mins)**
Sequential search			
95%	100%	0	166.18
70%	100%	0	7.81
B-K Tree			
95%	90%	0	25.15
70%	50%	0	0.55
Fixed Query Array			
95%	100%	0	76.65
70%	97%	0	3.28

between consecutive frames at fingerprint extraction from the audio. We included in Figure 4 searching using each and every row of the audio-fingerprint (Line B), searching using only the first row of the audio-fingerprint (Line C). We also include sequential search (Line A) as a reference.

As we observe, in Figure 4(a) at using the inverted index, checking every single row of the audio-fingerprint of the query on the LUT is the worst thing we can do, it is even worse than the sequential search approach. For overlaps of 95% it took over 1400 minutes to complete the offline monitoring, a similar result was obtained for an overlap of 90%. For an overlap of 85% the use of the inverted index outperforms the sequential search and continues to improve as the overlap percentage reduces. An amazing result is that of using just the first row of the audio-fingerprint of the query since it gets the lowest time by far for every overlap tested, in fact for 95% of overlap (the maximum overlap tested) it takes less than 17 minutes to search through all the 23 hours of audio.

As for the recall rate, as shown in Figure 4(b), with sequential search, we get a perfect recall for every overlap, meaning that in order for the index to be a better solution, it must beat the time taken by the sequential approach with the lowest overlap while still getting a 100% recall. In fact, we get perfect recall and a lower time with the inverted index with the approach of using every row

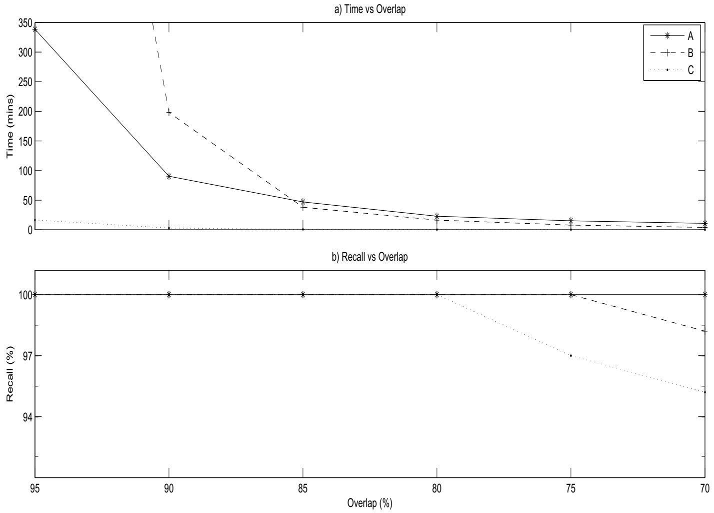


Fig. 4. Line A is the sequential search, Line B is the search with the inverted index using every row of the audio-fingerprint of the query, and Line C is the search with the inverted index using only the first row of the audio-fingerprint of the query

for an overlap of 75%, we get a lower time with an overlap of 70%, but we just manage to get about 98% of recall. Our proposed approach of using only the first row of the audio-fingerprint of the query reports by a far better time and still a perfect recall using overlaps of 90%, 85%, 80%. As for overlap of 75% and 70% it took only a few seconds, but at the cost of reducing the recall to 97% and 95% respectively.

5.2 On Line Monitoring Results

To perform Online Monitoring tests we setup a Shoutcast server and broadcasted 5 hours of the audio used in our offline tests. We picked the 5 hours with the

Table 3. Monitoring On line 23 Hours of audio using excerpts of ads of 7 seconds

Overlap	**Queries performed**	**Recall**	**Average time per query (ms)**
Sequential search			
85%	485393	100%	1.86
80%	363816	100%	1.17
Using all rows of the audio-fingerprint			
85%	485393	100%	1.003
80%	363816	100%	0.555
Using only first row of the audio-fingerprint			
85%	485393	100%	0.0286
80%	363816	100%	0.0200

highest number of advertisements. These tests were performed using the optimized parameters obtained in the offline experiments (excerpt size and overlapping), the results of such experiments are shown on Table 3, observe that using only the first row of the audio-fingerprint greatly reduces the required time to search the query while still achieving 100% of recall rate, this reduction is very important since it allows monitoring of several radio-stations with a single desktop computer.

6 Conclusions and Future Work

We found that MBSES performs really well in terms of robustness, since it is a binary matrix it enables the use of LUT to implement a fast inverted index. Since the level of degradation of audio-signals transmitted by Internet Radio Stations is relatively low, then the inverted index can be used in a way that the time required to perform a search is greatly reduced, that is using only the first row of the audio-fingerprint of the query. This is very important for on line monitoring since it allows for monitoring multiple radio stations with a single computer. However, offline search is more efficient, since several hours of audio can be monitored within few minutes (whenever broadcasts have already been fingerprinted). Besides, broadcast monitoring reports are normally required once a week or daily but rarely in real time.

As for future work, there are several proximity indexes that should be compared in order to; a) make queries even faster; b) Allow the system to work with more degradation on the monitored audio, some of them include; "permutants" based indexes, proposed in [13]; "Succinct Nearest Neighbors" based indexes proposed in [14]; and locality sensitive hashing proposed in [15]. We would also like to compare how the MBSES stands up against the popular fingerprint purposed in [16] which is the core of a very popular music tag service, in terms of performance (which includes speed, recall rate, robustness, scalability, and compactness).

References

1. Hellmuth, O., Allamanche, E., Cremer, M., Kastner, T., Neubauer, C., Schmidt, S., Siebenhaar, F.: Content-based broadcast monitoring using mpeg-7 audio fingerprints. In: International Symposium on Music Information Retrieval ISMIR (2001)
2. Nakamura, T., Tachibana, R., Kobayashi, S.: Automatic music monitoring and boundary detection for broadcast using audio watermarking. In: SPIE, pp. 170–180 (2002)
3. Eric, M.: Audio watermarking and applications (1999)
4. Camarena-Ibarrola, A.: Identificación automática de señales de audio. PhD thesis (2008)
5. Camarena-Ibarrola, A., Chávez, E., Tellez, E.S.: Robust Radio Broadcast Monitoring Using a Multi-Band Spectral Entropy Signature. In: Bayro-Corrochano, E., Eklundh, J.-O. (eds.) CIARP 2009. LNCS, vol. 5856, pp. 587–594. Springer, Heidelberg (2009)

6. Zwicker, E., Fastl, H.: Psycho-Acoustics. Facts and Models. Springer (1990)
7. Walter, B.A., Keller, R.M.: Some approaches to best-batch file searching (1973)
8. Chavez, E., Marroqun, J., Navarro, G.: Fixed query arrays: A fast economical data structure for proximity searching (2000)
9. Haitsma, J., Kalker, T.: A highly robust audio fingerprinting system. In: International Symposium on Music Information Retrieval (ISMIR) (2002)
10. Pavel, Z., Giuseppe, A.: Similarity search, the metric space approach. Springer (2005)
11. Kahmsi, M.A., Kirk, W.A.: An Introduction to metric spaces and fixed point theory. Wiley (2001)
12. Pestov, V.: Indexability, concentration, and vc theory. J. Discrete Algorithms 13, 2–18 (2012)
13. Karina, F.: Indexación efectiva de espacios métricos usando permutantes. PhD thesis (2007)
14. Tellez, E., Chavez, E., Navarro, G.: Succinct nearest neighbor search. In: The International Workshop on Similarity Search and Applications (SISAP) (2011)
15. Lv, Q., Josephson, W., Wang, Z., Charikar, M., Li, K.: A timespace efficient locality sensitive hashing method for similarity search in high dimensions (2011)
16. Li, A., Wang, C.: An industrial-strength audio search algorithm (2003)

Using Hidden Markov Model and Dempster-Shafer Theory for Evaluating and Detecting Dangerous Situations in Level Crossing Environments

Houssam Salmane[1], Yassine Ruichek[1], and Louahdi Khoudour[2]

[1] IRTES-SeT, University of Technology Belfort-Montbeliard,
13 rue Ernest-Thierry Mieg, 90010 Belfort Cedex, France
[2] LEOST-IFSTTAR, French Institute of Sciences and Technology for Transport,
Development and Networks,
20, rue Elisée Reclus, 59666 Villeneuve d'Ascq cedex, France
{houssam.salmane,yassine.ruichek}@utbm.fr,
louahdi.khoudour@developpement-durable.gouv.fr

Abstract. In this paper we present a video surveillance system for evaluating and detecting dangerous situations in level crossing environments. The system is composed of the following main parts: a robust algorithm able to detect and separate moving objects in the perceived environment, a Gaussian propagation model based dense optical flow for objects tracking, a Hidden Markov Model to recognize trajectories of detected objects, and an uncertainty model using theory of evidence to calculate the level of danger allowing to detect dangerous situations in level crossings. This method is tested on real image sequences, and the results are discussed. This work is developed within the framework of PANsafer project, supported by the ANR VTT program.

Keywords: object tracking, energy vector, optical flow propagation, Kalman filter, dangerous situations, Hidden Markov Model, Theory of evidence.

1 Introduction

The PANsafer (Towards a safer level crossing) project is concerned with increasing safety at level crossings. Video analysis technology is an efficient safety and control method, which is currently available at low costs. This solution can integrate other technologies such as using radio transmission systems to immediately transmit information about the state of a level crossing (presence of an obstacle, etc.) to users (vehicles, trains, etc.).

This paper aims to develop a mono camera based perception system for analyzing the environment of a level crossing. The proposed approach is composed of five stages. The first one consists of robustly detecting and separating moving objects. In the second stage, moving objects' pixels are tracked using a Gaussian

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 131–145, 2013.

model based optical flow. In the third stage, each moving object is segmented into different regions with homogeneous optical flow. The fourth stage is concerned with predicting a trajectory for each segmented region using a real-time Hidden Markov model. Finally, all trajectories are evaluated using Dempster-Shafer theory based strategy to estimate the degree of dangerousness related to the moving objects traveling in the level crossing environment.

The popular approach used to detect moving objects in an image is background subtraction [1], where the current frame is subtracted from the reference background image. Different approaches are proposed in order to improve the effectiveness of background removal algorithms in complex environments such as Independent Components Analysis [2], Histogram of Oriented Gradients [3], and wavelet [4]. Other algorithms are used to distinguish between detected moving objects such as equivalence table based algorithm [5], min-cut/max-flow algorithm and data association techniques [6]. These approaches are so effective, but it requires solving a lot of complicated problems to completely detect and separate the objects. In this paper, a new method which combines background subtraction with an energy vector comparison strategy is proposed for objects detection and separation. To obtain separated objects, this method consists in clustering moving pixels by comparing a specific energy vector associated to each target and each moving pixel.

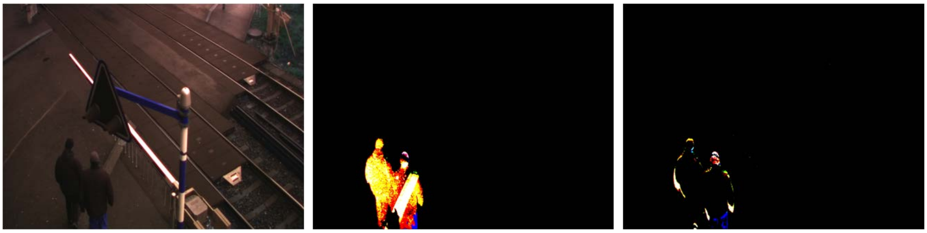
The process of tracking starts when there are enough detected pixels belonging to moving objects. Francois [7] proposes a Blob tracking algorithm based on object segmentation in successive frames. Li [8] uses an adaptive Kalman filter combined with mean shift technique, and the centre of each target candidate is tracked in a normalized color distribution. Blake [9] talks about boundary objects track using snakes and temporal fusion by Kalman filter. Yang [10] tries to calculate state-space distribution of the tracked objects using a particle filter. These different approaches are commonly used in the literature but all of them cannot track correctly all object pixels independently. In this paper, we propose a new method that improves significantly the tracking performance of each pixel within a detected object. This is achieved by a Harris points based optical flow propagation technique, followed by a Kalman filtering based correction.

Gaussian Mixture Model (GMM) [11], Hidden Markov Model (HMM) [12], and some of its extensions, such as the hierarchical Hidden Markov model (HHMM) [13] and the couple Hidden Markov model (CHMM)[14] are usually used for representing and recognizing objects' trajectories. However, these methods need a huge number of statistical measures to be effective. Hence, it is difficult to apply these methods in real time. In this paper we propose a real time Hidden Markov Model to predict trajectories of moving objects in a level crossing environment. For each moving object, the predicted trajectories only depend on the optical flow of segmented regions in the current image. As mentioned before, each object may be characterized by different trajectories, where each trajectory is associated to a region extracted by segmenting optical flow of the moving pixels. The trajectories prediction procedure takes into account the infrastructure geometry of the level crossing environment. To estimate the degree of dangerousness related to each object, each trajectory is analyzed considering different sources of danger (position, velocity, acceleration, etc.). All information provided by the sources are fused using Dempster-Shafer theory [15][16]. Once the trajectories analysis is achieved, the degree of dangerousness related to the object is simply the maximum of the degrees of dangerousness estimated for all the trajectories corresponding to the different regions of the object.

2 Objects Detection and Separation

Detecting moving objects is an important task in many applications such as video surveillance system. Among various objects' detection algorithms, background subtraction methods are the most simple and popular approaches. However, various problems such as illumination changes of environment, occlusion, and objects separation could not be easily resolved by a simple background subtraction algorithm. To deal with these problems, we have implemented a new algorithm for objects detection and separation. This algorithm has two main steps.

The first step consists of detecting moving object pixels. In this step, the current image is subtracted from the background image reference to obtain all detected pixels in the current image (Fig. 1(b)). The current image is also subtracted from the previous image to obtain moving pixels situated in the contour of the object in the current image (Fig. 1(c)).



(a) Current image (b) Detected pixels result (c) Moving pixels situated in the contour of the objects

Fig. 1. Detection of moving pixels

The second step starts by determining the targets in the current image. To achieve this task, the connectivity between the moving pixels situated in the contour of the objects is considered. A bounding box is then associated for each group of connected pixels (Fig. 2). The connected pixels situated in the contour of the object are completed by the other moving pixels situated inside the corresponding bounding box. Hence, all these pixels are represented by the same bounding box. Each moving pixel inside a current bounding box may represent a new or an old target. It depends of the intersection between the current bounding box and all the existing targets extracted from the previous image and represented by bounding boxes. All current bounding boxes that do not intersect the existing targets are considered as new targets. If a current bounding box intersects one or more existing targets, the separation method is based on an iterative procedure. At each iteration, we consider a pixel in the current bounding box, and the goal is to decide to assign it to one of the existing targets.

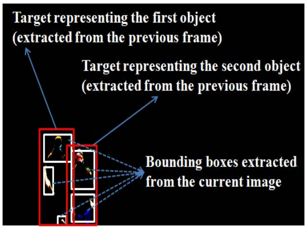

Target representing the first object
(extracted from the previous frame)
Target representing the second object
(extracted from the previous frame)
Bounding boxes extracted
from the current image

Fig. 2. bounding boxes extraction

To achieve the pixels clustering, two energy vectors are defined. The first one is concerned with each target. At the beginning of the processing of each frame, this energy vector is initialized to zero. It is then updated during each frame processing. Let E^i_{target} be the energy vector associated to the target number i. The second energy vector is defined for each pixel to be clustered. This energy vector, which is calculated with respect to each existing target, gathers appropriate characteristics of the objects, and then is used to compare different criterions that allow distinguishing a target from the other. Given a pixel located at the position (x, y) and a target number i, the energy vector E^i_{pixel} of the pixel is expressed as follows:

$$E^i_{pixel} = \left[E^i_G, E^i_I, E^i_F, E^i_D\right]^T \tag{1}$$

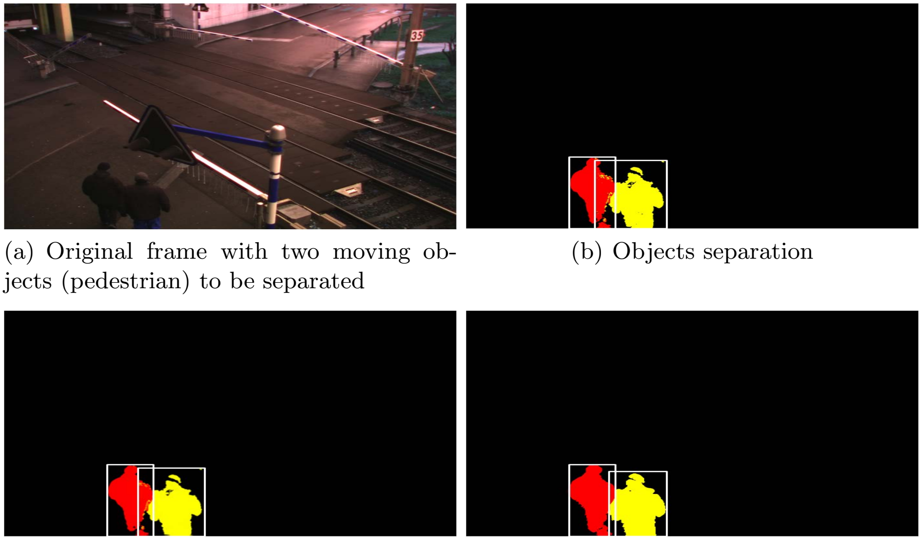
where E^i_G , E^i_I , $E^i_{E^i_F}$,E^i_D are respectively the gradient, intensity, optical flow and distance energies [17].

To assign the pixel (x, y) to a target, the energy vectors E^i_{pixel} and E^i_{target} are compared component by component, for each target. The pixel (x, y) is then assigned to the target that provides the maximum number of closest components. When a conflict situation occurs, i.e. when this criterion is respected by more than one target, the pixel is assigned to the target for which the distance energy components are the closest. Let p be the number of the target to which the pixel is assigned. After the clustering of the pixel, the energy vector of the target number p is updated as follows:

$$E^p_{target} = \left(\frac{N * E^p_{target} + E^i_{pixel}}{N + 1}\right) \tag{2}$$

where N is the number of pixels in the target number p, before adding the pixel (x, y).

Fig. 3(a) and Fig. 3(b) show respectively an original frame and the results of the objects separation method. Some clustering errors appear inside the first target (which one, must be indicated in the figure). These errors can be easily processed and corrected using mean shift filtering followed by mean shift reclassification [19] (see Fig. 3(c) and Fig. 3(d)).



(a) Original frame with two moving objects (pedestrian) to be separated

(b) Objects separation

(c) Objects separation after mean shift based filtering

(d) Objects separation after mean shift based reclassification

Fig. 3. Objects separation

3 Object Tracking

Once the targets are extracted from the current frame, the objective is to track them. To achieve that, an objects' tracking method based on optical flow is proposed. The objective is to track the maximum number of pixels with precise optical flow in order to get precise objects tracking.

The proposed method, starts by computing the optical flow of corner points, extracted by Harris operator [20], using Lucas-Kanade algorithm [21]. We consider that these particular points have a stable optical flow. The optical flow of Harris points is then propagated to compute the optical flow of the remaining pixels. In order to propagate the optical flow from textured areas into un-textured ones, we consider that the optical flow of all pixels of an extracted target follows a Gaussian distribution. The mean and standard deviation of the distribution are taken as the mean and standard deviation of the Harris points' optical flow [17][18]. In order to make the tracking process more robust against noise and to rectify the optical flow for each pixel, an iterative Kalman filter (KF) is designed [17][18]. The proposed KF provides the corrected state X_k from two inputs X_{k-1} and Y_k :

$$X_{k-1} = [p_x^{k-1}, p_y^{k-1}, ft_x^{k-1}, ft_y^{k-1}]^T \qquad Y_k = [p_x^k, p_y^k, Ft_x^{k-1}, Ft_y^{k-1}]^T \tag{3}$$

where p_x^{k-1} and p_y^{k-1} are the pixel coordinates at time $k-1$. p_x^k and p_y^k are the pixel coordinates measured at time k. ft_x^{k-1} and ft_y^{k-1} are respectively the horizontal and vertical optical flow at time $k-1$. Ft_x^{k-1} and Ft_y^{k-1} are respectively the horizontal and vertical optical flow measured at time $k-1$.

The first state X_{k-1}, which corresponds to the state to be corrected, is estimated from the optical flow propagation process at the frame $k-1$. The state Y_k corresponds to a measured state (from a virtual observation), and is computed from the optical flow propagation process at the frame $k-1$.

The tracking process is tested and evaluated in [17][18]. Fig. 4 shows an example of multi-objects tracking by combining the objects detection and separation method, and the tracking process.

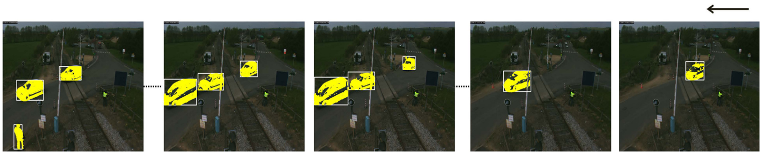


Fig. 4. Tracking process

4 Recognition of Dangerous Situations

In this section, we propose a method to recognize potential dangerous situations around a level crossing. The recognition process starts when a moving object is detected within the monitored area of the level crossing. Given a target detected by the tracking process, the proposed recognition method performs using three main steps. The first one consists in segmenting the target to extract different regions characterized by homogeneous optical flow. In the second step, an ideal trajectory is predicted from a current one for each extracted region using a Hidden Markov Model. Finally, in the third step, by considering the current ideal trajectory, the danger of the predicted ideal trajectory is estimated using Dempster-Shafer theory. Fig. 5 illustrates the synopsis of the method with its three main steps.

Optical Flow Based Object Segmentation. Given a target, the objective is to segment it into different regions basing on optical flow of its pixels. To achieve that, we use a simple recursive algorithm which compares neighboring pixels to extract regions in which the pixels have a homogeneous optical flow. Only regions with a significant size all conserved. Fig. 6 presents optical flow based segmentation results for a moving object tracked in an images' sequence.

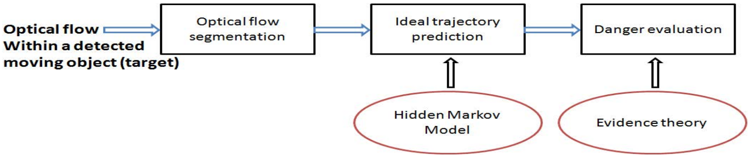


Fig. 5. Synopsis of the danger evaluation

The target is partitioned into multiple rectangular boxes representing the different regions with homogeneous optical flow. For ideal trajectories prediction (next step), each extracted region is considered by the center of its corresponding rectangular box. Hence, in the following, a trajectory is associated for each extracted region via its corresponding center.

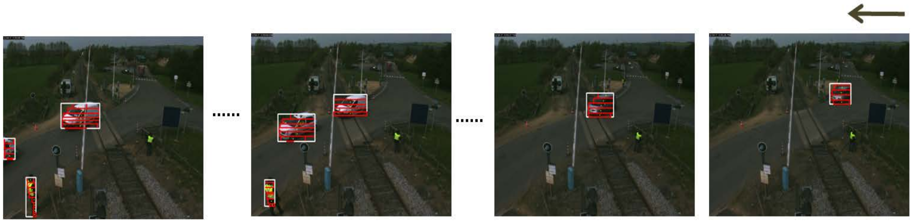


Fig. 6. Optical flow based object segmentation

Prediction of Ideal Trajectories. Given a region of a target extracted by the optical flow based segmentation process. By considering the center of the region, we define two trajectories. The first one, called current ideal trajectory, corresponds to the trajectory that the center of the region must follow to avoid potential dangerous situations. At the beginning, the current ideal trajectory is initialized as a direct line from the region center to the barrier of the level crossing, as shown in Fig. 7. From this current ideal trajectory and optical flow of the region center, we predict a new ideal trajectory that allows to the object to return to the current ideal trajectory. This new trajectory is called predicted ideal trajectory.

To predict the new ideal trajectory, we propose a statistical approach based on a Hidden Markov Model (HMM). The HMM has a set of parameters that need to be estimated to represent the level crossing environment modeling (Fig. 7). These parameters are:

- velocity and acceleration of the center of the considered region,
- direction of the current ideal trajectory of the considered region,
- distance D between the considered region and the barrier of the level crossing,
- velocity, orientation and acceleration intervals along the current ideal trajectory,
- current state of the level crossing barrier (closed or open).

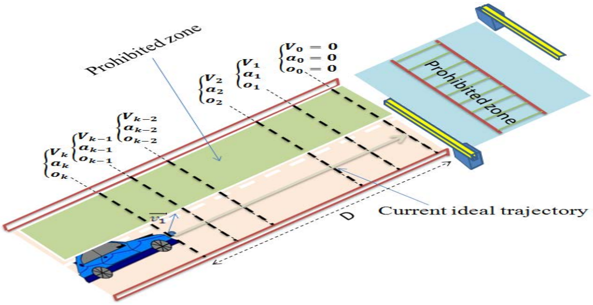


Fig. 7. Level crossing environment modeling parameters

The objective of the HMM is to find the optimal solution (trajectory) that allows to the object to keep or come back to the current ideal trajectory. Fig. 8 shows the general architecture of the proposed HMM, which is designed to perform ideal trajectory prediction from the current position of the considered region center. The random variable q_t is the hidden state at time t. It represents the current position of the considered region center. The random variable u_t is the observation at time t. It represents simultaneously the position, velocity, orientation and acceleration of the considered region center. Basing on the Markov property, the probability distribution of the hidden state variable q_t only depends of the distribution of the hidden state variable q_{t-1}. Similarly, the probability distribution of the observation variable u_t only depends of the distribution of the hidden state variable q_t at time t. The prediction of the ideal trajectory is provided by Forward-Backward, Viterbi and Baum-Welch algorithms [22][23][24][25].

Fig. 9 shows how ideal trajectory prediction is performed from the considered region center. Given the velocity vector at time t, calculated from optical flow, the state q_{t+1} in the HMM is reached from the state q_t with a probability of 1.

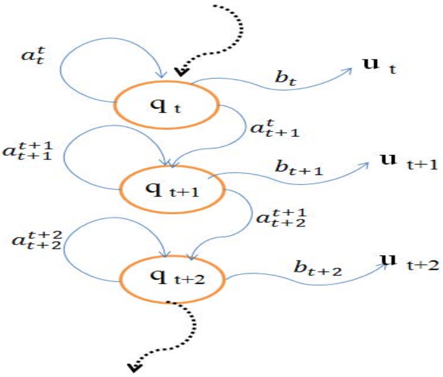


Fig. 8. HMM for ideal trajectory prediction

Given the velocity, orientation and acceleration intervals, we firstly predict the velocity vector $\overrightarrow{V_{t+1}}$ at time $t+1$. Next, we define a circle where the end point of $\overrightarrow{V_{t+1}}$ is the center and the maximum absolute acceleration within the acceleration interval is the radius. Each possible solution in the circle is represented by an observation composed with a position, velocity, orientation and acceleration. We recall at the same time that a position is represented by a state. Hence, the number of states and observations are determined from the number of possible solutions in the circle. Consequently, the number of states and the number of observations are the same, and equal to N. Then, each state has N observations.

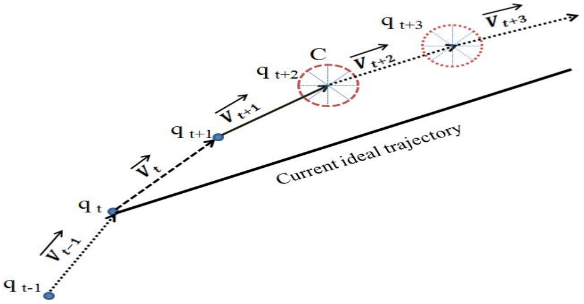


Fig. 9. Trajectory prediction model

Danger Evaluation. In this section, we describe the proposed method for evaluating the danger related to the predicted ideal trajectory of a region center. For that, the predicted ideal trajectory is analyzed considering various sources of dangerousness. The analysis is based on Dempster-Shafer theory, which allows to combine dangers induced by different sources and to obtain a degree of belief of danger that takes into account all possible hypotheses.

Firstly, we define a set of hypothesis H known as ”universal set”. It represents all the possible states of the model:

$$H = \{D, S\} \tag{4}$$

where D and S represents respectively the hypothesis of dangerousness and safety. Then, the power set 2^H that represents all subsets of H is defined as follow:

$$2^H = \{\varnothing, D, S, \phi\} \tag{5}$$

where $\varnothing$ is the empty set that represents the ”impossible proposition” and ϕ is the set that represents the ”certain proposition”. We also assign a belief mass function m, called as ”Basic Belief Assignment BBA”, to each element of the power set 2^H. This function depends on the knowledge concerning each source of dangerousness.

The next step is to determine the mass assignment for each source of dangerousness. If A is a given member of the power set 2^H, let $m^i(A)$ be the mass assignment related to the source of dangerousness number i. Given a region center, we consider five sources of dangerousness: position, velocity module, velocity orientation, acceleration, and distance between the region center and the current ideal trajectory.

Finally, all the mass assignments are combined using Demptster-Shafer paradigm in order to compute the degree of dangerousness of the considered region center.

This process is applied for each region extracted from a target. The maximum between all the computed degrees of dangerousness is chosen as the degree of dangerousness of the target.

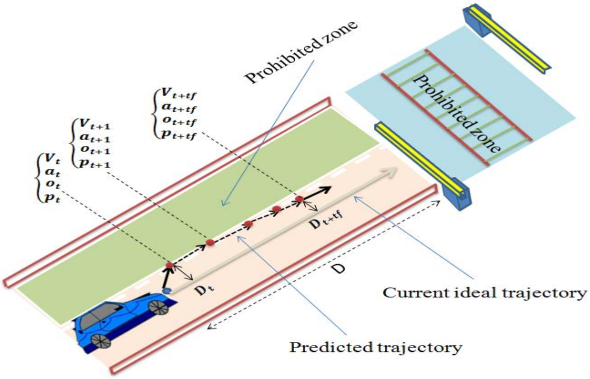


Fig. 10. Analysis of the predicted ideal trajectory

Fig. 10 presents an illustration of how to analyze the predicted ideal trajectory of a region center in order to evaluate the corresponding danger. As shown above, this trajectory is predicted using the HMM process. At each instant $(t, t+1, \ldots, t+t_f)$ of the predicted ideal trajectory, we determine the four following parameters: velocity $(v_t, \ldots, v_{t+t_f})$, acceleration $(a_t, \ldots, at+t_f)$, orientation $(o_t, \ldots, o_{t+t_f})$ and position $(p_t, \ldots, p_{t+t_f})$. These parameters are predicted for the considered region center. We also define another parameter which is the distance $(D_t, \ldots, D_{t+t_f})$ from the region center to the current ideal trajectory. These five parameters are used to define five sources of dangerousness. The first source concerns the velocity. To evaluate the danger related to this source of dangerousness, we consider the difference between the predicted velocity of the region center at time t and a predefined normal velocity (V_n). Then, we compute the mass assignment m_v that represents the degree of dangerousness related to velocity.

The second source of dangerousness concerns the acceleration. The mass assignment m_a, representing the degree of dangerousness related to the acceleration, is calculated from the difference between the predicted accelerations of the region center at times t and $t+t_f$.

The third source of danger is provided by the orientation velocity of the considered region center. The angle of the predicted velocity at time t is compared with the angle of the current ideal trajectory, and then the mass assignment mo, representing the degree of dangerousness related to the velocity orientation, is determined.

The fourth source of danger is related to the position of the considered region center. The degree of dangerousness related to this source is determined by computing the mass assignment m_p from the distance between the predicted position p_{t+t_f} at time $t + t_f$ and the barrier of the level crossing.

Finally, the fifth source of dangerousness is provided by the distance D_{t+t_f} between the predicted position p_{t+t_f} of the considered region center at time $t + t_f$, and the current ideal trajectory. The mass assignment md, representing the degree of dangerousness related to this source, is computed from the distance D_{t+t_f}.

Once the degrees of dangerousness are computed for the five sources, Demptster-Shafer combination is used to determine the degree of danger related to the considered region center:

$$Danger = Dempster - Shafer\left(m_v, m_a, m_o, m_p, m_d\right) \tag{6}$$

5 Experimental Results

In this section, we show some experimental results by applying the proposed dangerous situations recognition method on a scenario acquired in real conditions (Fig. 11). In this scenario, a first vehicle crosses the level crossing while the barriers are open. Then it stops just after the dangerous zone. Sometime later, a second vehicle tries to cross the level crossing before finding itself blocked behind the first vehicle. Finally, a third vehicle engages itself in the level crossing and blocks in its turn behind the file composed with the two first vehicles.

Fig. 11 illustrates also the obtaining results in terms of objects detection and tracking, and in terms of danger evaluation. One can see that all the moving objects are well detected and tracked during the video sequence. Concerning danger evaluation, the degree of dangerousness related to each detected vehicles increases when they move toward the level crossing, and reaches its maximum (46%) during the crossing of the zone of danger (between the barriers). When a vehicle stops in the zone of danger, the degree of dangerousness takes a value of 100%. When the vehicles move again, the degree of dangerousness decreases gradually as the vehicles leave away the level crossing. Note that, at the end of the video sequence, a pedestrian is detected and tracked, and its related danger is evaluated.

In terms of recognition of potential dangerous situations, when the of first vehicle is detected as stationary near to the level crossing, the recognized situation is the presence of a risk for creating a stopped vehicles line along the level crossing. When the second vehicle is detected blocked behind the first one in the zone of danger, an obstacle in the zone of danger is detected as well as a stopped

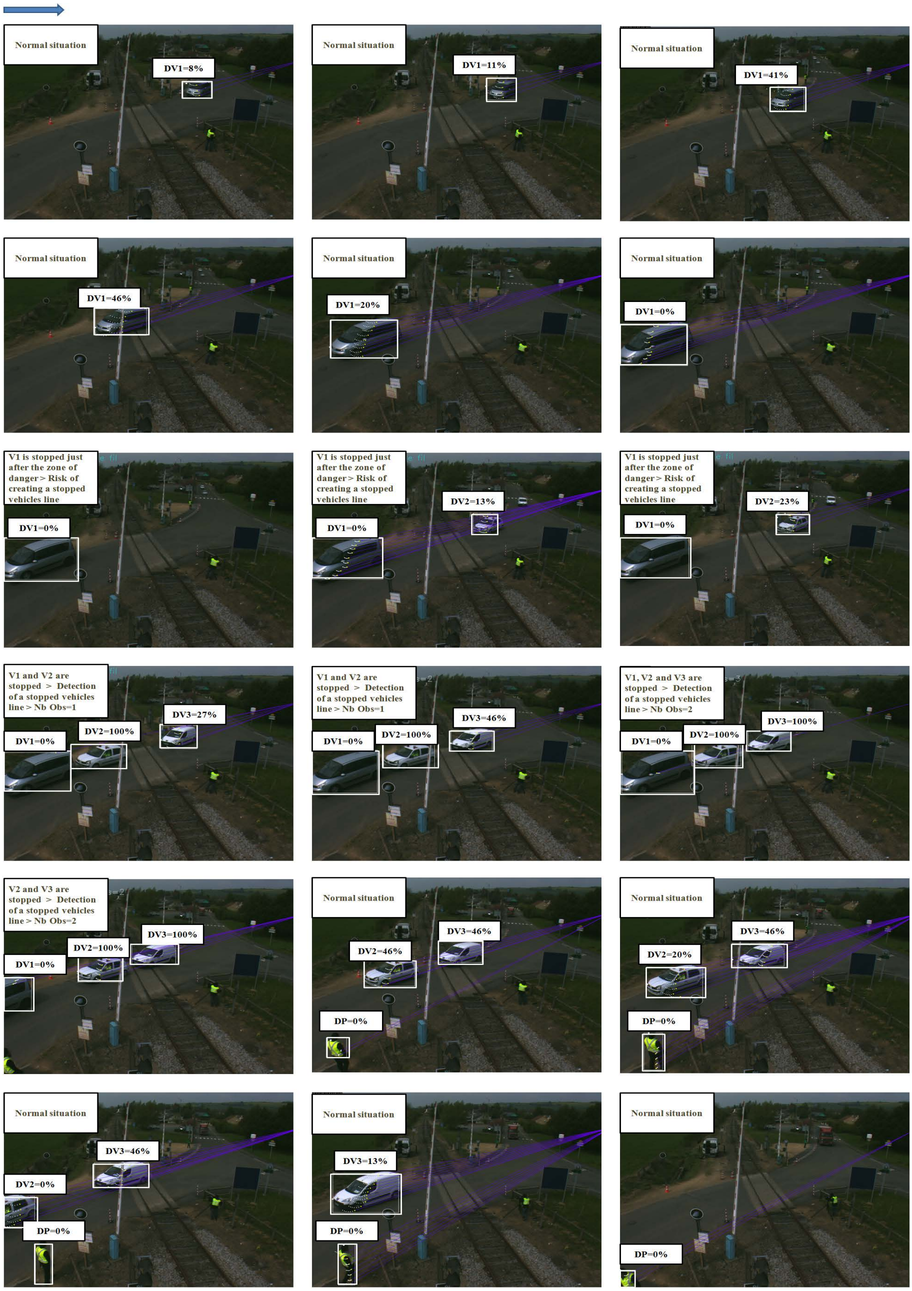

Normal situation
DV1=8%
Normal situation
DV1=11%
Normal situation
DV1=41%
Normal situation
DV1=46%
Normal situation
DV1=20%
Normal situation
DV1=0%
V1 is stopped just after the zone of danger > Risk of creating a stopped vehicles line
DV1=0%
V1 is stopped just after the zone of danger > Risk of creating a stopped vehicles line
DV2=13%
DV1=0%
V1 is stopped just after the zone of danger > Risk of creating a stopped vehicles line
DV2=23%
DV1=0%
V1 and V2 are stopped > Detection of a stopped vehicles line > Nb Obs=1
DV3=27%
DV2=100%
DV1=0%
V1 and V2 are stopped > Detection of a stopped vehicles line > Nb Obs=1
DV3=46%
DV2=100%
DV1=0%
V1, V2 and V3 are stopped > Detection of a stopped vehicles line > Nb Obs=2
DV3=100%
DV2=100%
DV1=0%
V2 and V3 are stopped > Detection of a stopped vehicles line > Nb Obs=2
DV3=100%
DV2=100%
DV1=0%
Normal situation
DV3=46%
DV2=46%
DP=0%
Normal situation
DV3=46%
DV2=20%
DP=0%
Normal situation
DV3=46%
DV2=0%
DP=0%
Normal situation
DV3=13%
DP=0%
Normal situation
DP=0%

Fig. 11. Danger evaluation and recognition of potential dangerous situations. *DVi* represents the danger related to the vehicle number *i* and DP represents the danger related to the pedestrian.

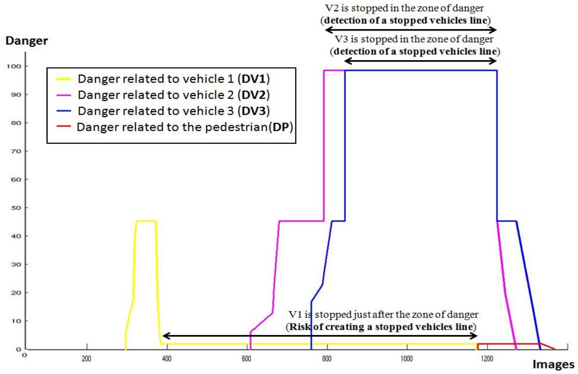


Fig. 12. Evolution of the degree of dangerousness related to each detected and tracked object

vehicles line. Finally, when the third vehicle is detected blocked behind the two first ones, we detect two obstacles in the zone of danger, and the situation of a stopped vehicles line is maintained.

Fig. 12 presents the evolution of the degree of dangerousness related to each detected and tracked object during the video sequence. One can see from this figure the three recognized potential dangerous situations.

6 Conclusion

A method to evaluate and recognize potential dangerous situations in level crossing environments is proposed. This method starts by detecting and tracking objects seen in the monitored zone by a video camera. The detection and tracking process is performed using probabilistic propagation based optical flow. The second stage of the method consists of predicting for each tracked object the ideal trajectory allowing to avoid potential dangerous situations. The ideal trajectory prediction is based on a Hidden Markov Model. The third stage is concerned with the analysis of the predicted trajectory in order to evaluate the danger related to each tracked object. This stage is performed by considering different sources of dangerousness and applying a Demptster-Shafer based combination. All the stages of the proposed method are tested and evaluated using real data. Then, the whole method is tested considering a typical scenario gathering two different potential dangerous situations (obstacle detection and detection of stopped vehicles lines). The method is applied and tested on other scenarios. The obtained results have shown the effectiveness of the proposed approach.

Future works should concern the validation of the proposed method through statistical results in terms of true/false detection. We plane also to extend the proposed approach by considering stereovision based video analysis in order to make the method more robust against shadow problems and objects detection by exploiting 3D information.

References

1. Panahi, S., Sheikhi, S., Hadadan, S., Gheissari, N.: Evaluation of Background Subtraction Methods. In: Computing: Techniques and Applications, DICTA 2008, pp. 357–364. Digital Image (2008)
2. Fakhfakh, N., Khoudour, L., El-Koursi, E.M., Bruyelle, J.L., Dufaux, A., Jacot, J.: Background Subtraction and 3D Localization of Moving and Stationary Obstacles at Level Crossings. In: International Conference on Image Processing Theory, Tools and Applications, Paris (2010)
3. Bertozzi, M., Broggi, A., Rose, M., Felisa, M.: A Pedestrian Detector Using Histograms of Oriented Gradients and a Support Vector Machine Classifier. In: Intelligent Transportation Systems Conference (2007)
4. Toreyin, B.U., Cetin, A.E., Aksay, A., Akhan, M.B.: Moving object detection in wavelet compressed video. In: Signal Processing: Image Communication, pp. 255–264 (2005)
5. Suzuki, K., Isao, H., Noboru: Linear-time Connected-component Labeling based on Sequential Local Operations. J. of Computer Vision and Image Understanding 89, 1–23 (2003)
6. Kirubarajan, T., Bar-Shalom, Y.: Data Association Combined with the Probability Hypothesis Density Filter for Multitarget Tracking. In: Proceedings of SPIE Signal and Data Processing of Small Targets, p. 52 (2004)
7. Francois, A.R.J.: Real-Time Multi-Resolution Blob Tracking. In: IRIS Technical Report IRIS-04-422. University of Southern California, Los Angeles (2004)
8. Li, X., Zhang, T., Shen, S., Jiancheng, S.: Object tracking using an adaptive Kalman filter combined with mean shift. In: Optical Engineering. SPIE (2010)
9. Blake, A., Isard, M.: Active Contours. Springer (1998)
10. Yang, C., Duraiswami, R., Davis, L.: Fast multiple object tracking via a hierarchical particle filter. In: International Conference on Computer vision (2005)
11. Bashir, F., Khokhar, A., Schonfeld, D.: Automatic Object Trajectory-Based Motion Recognition using Gaussian Mixture Models. In: IEEE International Conference on Multimedia and Expo., Amsterdam (2006)
12. Bashir, F., Khokhar, A., Schonfeld, D.: Object Trajectory-Based Activity Classification and Recognition Using Hidden Markov Models. In: IEEE Transactions on Image Processing, pp. 1912–1919 (2007)
13. Nguyen, N.T., Phung, D.Q., Venkatesh, D., Bui, H.: Learning and Detecting Activities from Movement Trajectories Using the Hierarchical Hidden Markov Model. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, CVPR (2005)
14. Natarajan, P., Nevatia, R.: Coupled Hidden Semi Markov Models for Activity Recognition. In: Proceedings of IEEE Workshop on Motion and Video Computing (2007)
15. Yager, R.R., Fedrizzif, M., Kacprzyk, J.: Advances in the Dempster-Shafer Theory of Evidence. John Wiley and Sons (1994)
16. Dubois, D.: Possibility theory and statistical reasoning. In: Computational Statistics & Data Analysis, pp. 47–69 (2006)
17. Salmane, H., Ruichek, Y., Khoudour, L.: Gaussian propagation model based dense optical flow for objects tracking. In: International Conference on Image Analysis and Recognition, ICIAR (2012)
18. Salmane, H., Ruichek, Y., Khoudour, L.: Object tracking using Harris corner points based optical flow propagation and Kalman filter. In: IEEE Intelligent Transportation Systems Conference, ITSC (2011)

19. Cheng, Y.: Mean Shift, Mode Seeking, and Clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence (1995)
20. Derpanis, K.G.: The Harris Corner Detector (2004), http://www.cse.yorku.ca/kosta/CompVis_Notes/harris_detector.pdf
21. Tamgade, S.N., Bora, V.R.: Motion Vector Estimation of Video Image by Pyramidal Implementation of Lucas Kanade Optical Flow. In: Second International Conference on Emerging Trends in Engineering & Technology (2009)
22. Baum, L.E., Sell, G.R.: Growth functions for transformations on manifolds. Pac. J. Math. Stat. 41, 211–227 (1968)
23. Viterbi, A.J.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Transactions on Information Theory, 260–269 (1967)
24. Baum, L.E., Petrie, T., Soules, G., Weiss, N.: A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Math. Statist. 41, 164–171 (1970)
25. Lawrence, R.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE (1989)

Local Features Classification for Adaptive Tracking

Alan I. Torres-Nogales, Santiago E. Conant-Pablos, and Hugo Terashima-Marín

Tecnológico de Monterrey, Campus Monterrey
Av. Eugenio Garza Sada 2051, Monterrey, N.L.
64849 Mexico
{a00342588,sconant,terashima}@itesm.mx

Abstract. In this paper we propose using invariant local features and a global appearance validation for assisting a robust object tracker initialized by a single example. Although local features have been used before for several object detection and tracking applications, these approaches often model an object as a collection of key-points and descriptors, which involves constructing a set of correspondences between object and image key-points via descriptor matching or key-point classification. However, these algorithms cannot properly adapt to long video sequences due to their limited capacity for incremental update. We differentiate from these approaches in that we obtain key-point-to-object correspondences instead of key-point-to-key-point correspondences converting the problem into an easier binary classification problem, which allows us to use a state-of-the-art algorithm to incrementally update our classifier. Our approach is embedded into the Tracking-Learning-Detection (TLD) framework by performing a set of changes in the detection stage. We show how measuring the density of positive local features given by a binary classifier trained on-line is a good signal of the object's presence, and in combination with a global appearance validation it yields a strong object detector for assisting a tracking algorithm. In order to validate our approach we compare the tracking results against the original TLD approach on a set of 10 videos.

Keywords: Random Ferns, Adaptive Tracking, Semi-supervised Learning.

1 Introduction

Recent algorithms have focused their attention into solving adaptive tracking and detection by modeling it as a semi-supervised learning problem. These algorithms start with a single example of the object and learn on-line a classifier using a set of spatio-temporal constraints, this technique has led to state-of-the-art results. The Track-Learn-Detect (TLD) system [8] developed by Zdenek Kalal is one of such methods. We focus on this approach due to its ability to label visual content with very little user interaction. We build upon this approach in our research towards generic semi-supervised object tracking and detection. Even

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 146–157, 2013.

though TLD work great for planar rigid objects, we would like to improve its generalization to be able to eventually track any kind of object. This is our first step towards achieving that goal.

In order to be able to track any object, the planarity assumption has to be broken. That is our main motivation to replacing the global classifier with a more flexible local classifier. The use of invariant local features involves finding correspondences and there are two main approaches to do this: matching and classification. Matching based approaches store descriptors for each model key-point in a database. These descriptors are designed to be invariant to various transformations and noise, they can be matched to new key-points in an image given a distance metric. Classification based approaches treat matching as multi-class classification, in which the task is to classify each image key-point as a particular key-point from a model. These classifiers are trained off-line from several examples of the object, created synthetically using random transformations. Although these approaches work very well in some cases, these algorithms suffer from either a cumbersome training stage or have a limited capacity for incremental update.

We differentiate from these approaches in that we obtain key-point-to-object correspondences instead of key-point-to-key-point correspondences, converting the problem into an easier binary classification problem, which allows us to use the TLD framework [8] to incrementally update our classifier. As in TLD the training starts with a single labeled example of the object, then the object is simultaneously tracked and detected in all following frames. Invariant local features [14] are extracted from each frame and classified using a fern classifier [11]. The classifier is then used to label key-points in the next frame as either belonging to the object or the background, giving a sparse confidence map. All features that belong to the object are then pooled across locations using a multi-scale grid structure. For each bounding box in the grid a density is calculated and the densest bounding box per scale is considered a candidate detection. The candidate patches are passed through a global validation given by a nearest neighbor classifier and integrated with the tracker results. Finally the classifiers performance is bootstrapped by using PN-Learning [8]. The preliminary results of our work shows that we are able to outperform TLD on tracking tasks.

The rest of the paper is organized as follows. First we review the related work in the next section. In Section 3 we introduce our contributions. Section 4 contains the implementation details. In Section 5 we compare our system against the state-of-the-art approach from [8]. Finally we present our conclusions and future work in Section 6.

2 Related Work

In [1] tracking is considered as a binary classification problem, where an ensemble of weak classifiers is trained on-line to distinguish between the object and the background. The ensemble of weak classifiers is combined into a strong classifier using AdaBoost [6]. This classifier is then used to label pixels in each frame as either belonging to the object or the background, giving a confidence map. The new position of the object is found using mean shift over the density map, initialized to the last know position.

The Viola-Jones cascade detector [15] introduced three key contributions: Integral Images, which allowed the features used by their detector to be computed very quickly. A learning algorithm, based on AdaBoost, to select a small number of critical visual features from a larger set and produce efficient classifiers. And the "cascade" methodology, for combining increasingly more complex classifiers, which allows background regions of the image to be quickly discarded while spending more computation on promising object-like regions. The TLD system [8] retake Viola-Jones cascade methodology [15] to design a global object detector. A new paradigm for learning from structured unlabeled data (i.e. a video) called P-N Learning is introduced, where the structure in the data is exploited by so called positive and negative structural constraints, which enforce certain labeling of the unlabeled set. P-N learning is applied to the problem of on-line learning of object detector during tracking. They show that an accurate object detector can be learned from a single example and an unlabeled video sequence.

In [5] a tracking method based on the TLD system [8] is presented that exploits the context of the object of interest by characterizing it in two terms: Distractors and Supporters. Distractors are regions which have similar appearance as the target and consistently co-occur. They keep tracking of these distractors to avoid drifting. Supporters are local key-points around the target with consistent co-occurrence and motion correlation, characterized by SURF features [2]. Their main purpose is to make a stronger validation that the object is being tracked correctly.

In [7], a framework for adaptive visual object tracking based on structured output prediction called STRUCK is presented. Traditional tracking-by-detection approaches use the detected location given by the tracker to generate a set of samples which are used to train a classifier. STRUCK avoids the need for intermediate classification steps and operates directly on the tracking output by explicitly allowing the output space to express the needs of the tracker. They use a kernelized structured output Support Vector Machine (SVM), which is learned on-line to provide adaptive tracking. To allow for real-time application, it uses a budgeting mechanism which prevents the unbounded growth in the number of support vectors which would otherwise occur during tracking.

3 Local TLD

The TLD framework [8] is a very robust and well designed approach, along with the PN-Learning technique for bootstrapping a binary classifier. To speed up the detection process, the TLD system uses a sliding-window global cascade detector, where a variance filter gives a weak signal of the object's presence, then a fern classifier provides a stronger response and a template matching technique, implemented by a nearest neighbor classifier, determines the final response, that is, it validates the detections from the previous stages.

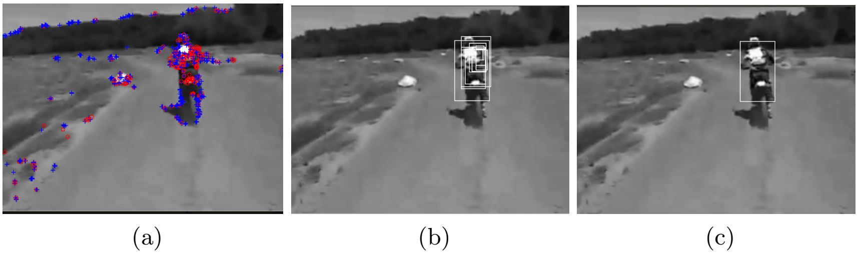


Fig. 1. Detector Overview: (a) Local features are extracted from the image and classified by a fern classifier as either belonging to the object (red circles) or the background (blue crosses). (b) All features that belong to the object are then pooled across locations using a multi-scale grid structure and a confidence is calculated for each bounding box in the grid. The most confident bounding box for each scale is considered a detection candidate. (c) Each candidate is passed through the global validation stage given by the nearest neighbor classifier, if the most confident detection is above a confidence threshold it is considered as the final detection.

The cascade configuration, as stated by Viola and Jones [15], can be viewed as an object specific focus-of-attention mechanism which provides statistical guarantees that discarded regions are unlikely to contain the object of interest. However given the nature of the global detection, sometimes this focus-of-attention is lost due to small local changes in the object, this is principally reflected by the fact that sometimes even though a layer of the detector gives a correct detection, the following and more selective layer of the detector, rejects the detection. We believe this can be avoided by relaxing the selectiveness of the detector and studying the precision and recall on each individual layer of the detector.

The principal contribution of our approach is to replace the global classifier in the firsts steps of the cascade detector by a more flexible local feature classifier. We show how the efficiency of the detector can be significantly improved even by just relaxing the planarity assumption on this stage. In order to achieve this we introduce several changes to the framework which are described below.

3.1 Local Features

For local feature detection and description we selected the ORB features [14], since they are fast to compute and already include some kind of invariance to illumination, scale and rotation changes. These features are a modified version of the FAST [13] corner detector and the BRIEF [4] descriptor, named ORB for Oriented FAST and Rotated BRIEF. They achieve invariance to rotation by using a simple measure of corner orientation, the intensity centroid [12], in the corner detection step and then rotating the descriptor accordingly. Their descriptors are binary strings. Each bit in the string represents an intensity comparison between two pixels in a given patch. They are robust to illumination changes and they can work in a multi-scale setting by using image pyramids.

The classical approach to object detection using local features is feature matching. For fast matching the authors of [14] propose to use Local Sensitive Hashing [10]. However this approach would involve growing the database of know features in order to perform adaptive tracking, this does not scale well in long videos. In [11] the matching is defined as a key-point recognition problem. However the training of the algorithm is a very computationally expensive procedure which has to be performed off-line. We are not using either of these approaches, instead we refrain from matching specific key-points and formulate the features classification problem as a binary feature classification, classifying each feature as either belonging to the background or to the object as in [1].

3.2 Local Features Classification

The BRIEF descriptor [4] was originated as an alternative version of the key-point Fern Classifier [11], removing the training phase, fixing the random features and re-formulating the problem as matching instead of recognition.

Same as the features used by the ferns [11] and BRIEF descriptor [4], each of the bits that makes up the ORB descriptor comes from a pixel comparison operation. On initialization we select random features from the descriptors and build fern structures as in [8]. These fern structures are hashing functions where each fern with S features outputs a number between 0 and $2^S - 1$ which points to a leaf node. On the training phase each leaf-node accumulates the number of positive p and negative n examples that are mapped there by the fern function. During testing each fern points to the leaf-node with posterior probability $P(y = 1|x_i)$, this posterior is computed by maximum likelihood estimator, $P(y = 1|x_i) = p/(p+n)$, or is set zero if the leaf is empty. The ferns posteriors are averaged and compared against a threshold which yields a positive or negative response that indicates if the local features belongs to the object or not.

3.3 Feature Density Estimation

Local feature classification, as explained above, infers only per key-point information about the presence of the object, this alone is a very weak indicator. This information now should be pooled across multiple key-points to generate reliable proposals for the position of the object. This procedure is similar to the mean-shift technique used in [1], however, our technique allows for an exhaustive search of maximum density regions due to fast features density calculation for each bounding box on the sliding window grid.

Here we exploit the sliding window approach embedded in the TLD framework [8]. After the local features are classified, the positive features posteriors are mapped into a confidence map and accumulated using an integral image [15]. The positive feature density of window i is estimated by

$$d(w_i) = \sum_{j \epsilon w_i} (p_j) \tag{1}$$

where p_j is the accumulated confidence of the positive features that were mapped to pixel j. This density can be compared against a learned threshold to filter out the patches locations with poor texture and low density of positive features, however there would be no way to determine how many bounding boxes would pass this filter. Instead we analyze a fixed number of detections by considering as a candidate detection the densest bounding box on each scale s of the grid:

$$w_s* = \mathrm{argmax}_{i \epsilon s} d(w_i)$$

3.4 Global Validation

All candidate detections are compared against all trained examples with a Nearest Neighbor classifier. If the query patch is highly correlated with one of the examples then it belongs to the same class. A confidence score is calculated based on its relative similarity to the nearest positive and negative patches in the following way

$$C(p) = \frac{1 - \max_i(\mathrm{ncc}(p, p_{ni}))}{2 - \max_j(\mathrm{ncc}(p, p_{pj})) - \max_i(\mathrm{ncc}(p, p_{ni}))} \tag{2}$$

where p is the query patch and p_{ni} is the ith negative example and p_{pj} is the jth positive example in the nearest neighbor database.

The similarity measure used is the normalized cross-correlation ncc(.,.) or cosine-similarity, and is calculated by:

$$\mathrm{ncc}(\overrightarrow{u}, \overrightarrow{v}) = \frac{\overrightarrow{u} \cdot \overrightarrow{v}}{||\overrightarrow{u}|| \cdot ||\overrightarrow{v}||} \tag{3}$$

where $\overrightarrow{u}$ and $\overrightarrow{v}$ are zero-mean templates. The $ncc(\cdot, \cdot)$ represents the cosine of the angle between the templates and attains a value of one when the vectors are parallel and zero when orthogonal. The most confident patch is compared against a threshold, if the confidence score is higher than this threshold then the nearest neighbor classifier gives a positive response.

Budgeting. When the videos are long and there is much variation on the object's appearance, many examples need to be learned in order to account for the variation. This does not affect the fern classifier, however it does affect the Nearest Neighbor classifier because while the database grows it takes longer to classify a new example. We overcome this issue by setting a fixed budget, that is, we only store a limited number of examples in the database, if this limit has been reached then all new examples will substitute its nearest neighbor on the database. By the implementation of this mechanism, some accuracy can be lost in the final detector since examples that are added to the database in the beginning of the video are then replaced when the database reach its capacity limit, however we are focused on improving efficiency for the tracking setting where this mechanism should not affect us.

4 Implementation

Basically, Local TLD uses the same general framework as the PN-Tracker [8]: The system is initialized in the first frame by manually setting a bounding box around the object of interest. An initial detector is trained from that single frame and the initial position of the tracker is set. For each following frame, the detector and the tracker find the location of the object. The detected patches close to the trajectory given by the tracker are given a positive label and detections far away from this trajectory are used as negative examples. If the trajectory is validated, these examples are used to progressively keep training the classifier with different views of the object. However, if there is a strong detection far away from the track, the tracker is reinitialized and the collected examples discarded. The main idea is to recognize tracking failure and only update the classifier when the tracking is correct.

4.1 Initialization

The system is initialized in the first frame by setting a bounding box around the object of interest. Then the local features are extracted from the image. This bounding box is used to create the grid structure with several bounding boxes distributed across the image in multiple scales needed for the fast density calculation. The feature density of the selected patch is calculated and is used to set a threshold for the density filter.

Both classifiers in the cascade detector are trained in the first frame and then are progressively updated in every frame with a validated tracking trajectory.

The fern classifier is initialized by selecting the random bit positions that are going to be extracted from the binary strings descriptors to create the fern structures. Once selected, these structures remain constant throughout the learning.

All the local features inside the given bounding box are given a positive label and all features outside the bounding box are given a negative label. These feature descriptors are used to initialize the training of the fern classifier. The selected patch is normalized to 15x15 zero-mean template and added to the Nearest Neighbor database as a positive example. High density patches far from the object are also normalized and added as negative examples for the Nearest Neighbor classifier. In order to improve the generalization of the classifiers, the positive examples are augmented by randomly warping the selected area 20 times with small transformations.

4.2 Tracking

The tracking is performed using the Median Flow tracker which consist of a pyramidal Lucas-Kanade tracker [3] with several modifications proposed in [9]. A slight modification was made on the points to track, the original implementation initializes a uniform 10x10 key-points grid within the last know bounding box and track those points. We instead randomly select 100 key-points given by feature extraction stage inside the last given bounding box since they are more likely to be stable for tracking purposes.

4.3 Detection

Fern Classifier. The first step in the detection stage is a local feature extraction at multiple scales using an image pyramid. These features are the input to the fern classifier which makes a series of measurements on the descriptor vector and maps each feature to a confidence score. If the confidence is above 50%, the feature is considered to belong to the object. The measurements made by the fern classifier are saved to update the classifier in case that the trajectory of the tracker is validated.

Feature Density Filter. Once the features have been classified, the positive ones are projected to a confidence map which is then represented as an integral image. Based on a sliding window strategy, the integral image is scanned across positions and scales, at each sub-window a density estimation is calculated as in Eq. 1. We select as candidate detections the densest bounding box at each scale of the grid.

Global Validation. The candidate detections made by the fern classifier are compared against all trained examples from the Nearest Neighbor classifier. And a confidence score is calculated by Eq. 2 and the most confident patch is kept. If this patch confidence is higher than 60% then the nearest neighbor classifier gives a positive response.

4.4 Integration

On the integration stage, the patches locations given by the detector are compared with the trajectory given by the tracker. If the object wasn't tracked but there is high-confidence detection, the tracker is reinitialized. If the object was tracked but the detector made a detection with higher confidence and a maximum overlap of 50% with the bounding box given by the tracker, the tracker location is reinitialized. If the detected locations and the tracker bounding box are overlapped over 70% then they're averaged together and the trajectory is validated.

4.5 Learning

The learning component of the framework is given by the PN-Learning[8] technique. If the trajectory given by the tracker is validated, it triggers the application of the P-N constraints. Meaning that all features detected inside the predicted bounding box are assigned a positive label, and all features detected in the surroundings (outside the bounding box) of the validated trajectory are given a negative label. The features labeled as positive by the P-N constraints which are below the fern confidence threshold and the negative features which are above the threshold are used to update the fern classifier.

The same procedure is used to augment the Nearest Neighbor database. If the predicted patch is below the nearest neighbor threshold it is added to the positive examples. All patches locations that were validated by the detector are compared with the predicted location, if their overlap is lower than 20% they are added to the bad examples database.

5 Experiments

The performance of the detector and the PN-Tracker is evaluated using the following metrics : precision P is the number of correct detections divided by number of all detections, recall R is the number of correct detections divided by the number of object occurrences that should have been detected and f-measure F combines these two measures as $F = 2PR/(P + R)$. A detection is considered to be correct if its overlap with the ground truth bounding box is larger than 50%. We perform all of our experiments in the publicly available TLD dataset[1], which was used by [8].

5.1 Parameters

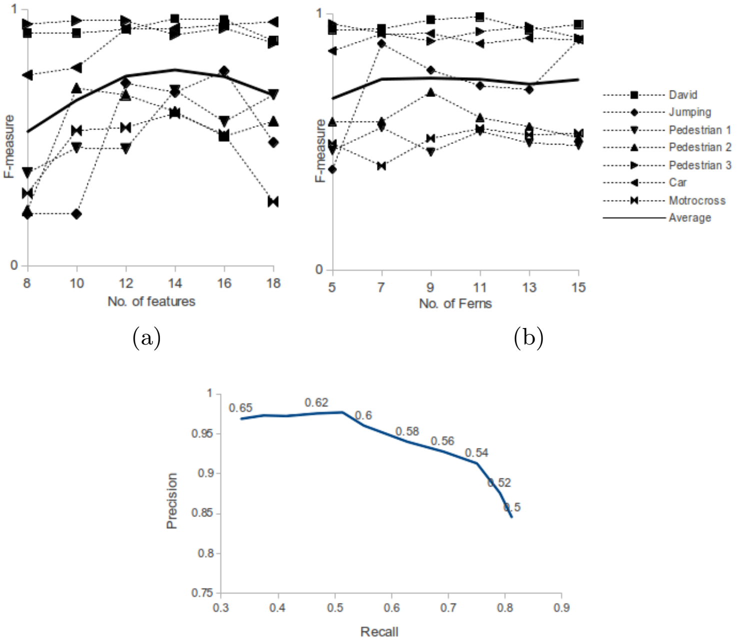
To build the fern classifier we need to set two parameters: number of features per fern, and number of independent fern structures. These parameters determine the accuracy and speed of the feature classification stage, we try to find the parameters which maximize precision and recall, and minimize processing time.

Since features are randomly selected at initialization there are some differences from one run to another which can affect the classifier's performance, the results show the average over 5 runs. In a first experiment we try to determine the optimal number of features per fern to obtain the best precision and recall. Fig. 2a shows the results of such experiments. As we can see from the graphs, the best precision and recall are obtained using around 14 features per Fern. We fix this parameter for all following experiments.

In a second experiment we test the number of ferns needed for the algorithm to have good performance in terms of speed and accuracy. While the number of ferns grows, so does the stability but speed drops linearly. From Fig. 2b we can see that 7 is the minimum number of ferns we can have without impacting accuracy.

To calibrate the confidence threshold in our detector we develop a Precision and Recall analysis in several videos (sequences 1-6 in the dataset). We consider a correct detection if the ground truth corresponds to the most confident detection given by the cascade classifier. Fig. 2c shows a P-R curve of our detector, the labels on the curve indicate the threshold value used in the last layer of the detector to validate the detection.

[1] Available at `http://info.ee.surrey.ac.uk/Personal/Z.Kalal/TLD/TLD_dataset.ZIP`



(a) (b)

(c)

Fig. 2. (a) Effect of number of features. Number of ferns is fixed at 10. (b) Effect of number of ferns. Number of features is fixed at 14. (c) Threshold P-R Curve.

5.2 Validation

In order to validate the integration of our approach into the TLD framework we compare our implementation with the results from the original implementation by Kalal et. al[8]. To allow different initializations when comparing a trajectory to ground truth, the trajectory is normalized (shift, aspect and scale correction) so that the first bounding box matches the ground truth; all remaining bounding boxes are normalized with the same parameters. The results of our experiments are show in Table 1. We can observe that in average our approach outperform the TLD in the tracking tasks using fewer appearance validations. The program was tested at an average frame rate of 28 fps on a Core i7 laptop running Ubuntu.

It is important to note a few details when working with local features: In the David video for example, local features are not detected by the ORB feature detector inside the initial bounding box, that is caused by the absence of texture in the region, however the adaptive tracker in the TLD system is able to track the face to the point where features become apparent and the classifier starts learning. A weakness of our approach when setting a fixed number of features to be returned by the features detector is that, in the presence of very cluttered backgrounds, features on the actual object are not strong enough to be retained.

Table 1. Tracking performance analysis

		TLD			Our approach		
Sequence	Frames	*P*	*R*	*F*	*P*	*R*	*F*
1. David	761	**1**	**1**	**1**	0.92	0.91	0.91
2. Jumping	313	**0.99**	**0.99**	**0.99**	0.66	0.55	0.60
3. Pedestrian 1	140	**1**	**1**	**1**	0.18	0.17	0.17
4. Pedestrian 2	338	0.89	0.92	0.91	**0.96**	**0.93**	**0.95**
5. Pedestrian 3	184	**0.99**	**1.00**	**0.99**	0.83	0.94	0.88
6. Car	945	0.92	0.97	0.94	**0.95**	**0.97**	**0.96**
7. Motocross	2665	0.67	0.58	0.62	**0.70**	**0.64**	**0.67**
8. Volkswagen	8576	0.54	0.40	0.45	**0.71**	**0.88**	**0.78**
9. Car Chase	9928	0.50	0.40	0.45	**0.71**	**0.34**	**0.46**
10. Panda	3000	**0.32**	**0.34**	**0.33**	0.35	0.21	0.26
Average	26850	0.55	0.55	0.55	**0.68**	**0.58**	**0.60**

This effect is present in the Jumping video, where the face has poor texture related to the background due to motion blur, however our classifier is able to adjust well.

6 Conclusions

We have introduced a new way for object detection by classifying local features as either belonging to the object or the background. We have shown how by simply measuring the local density of positive local features is a good enough signal of the object's presence, and in combination with a global validation it yields a strong and adaptable object detector to reinitialize a tracker from failure. We also show how a local feature binary classifier can be successfully trained on-line using PN-Learning and how we can improve the efficiency of the detection stage even by restricting the costly validation process to the most confident patches at each scale. Although in this paper we were not able to completely replace the global detector, we set the base to future work on this area. The use of local features allows the framework to be extended by replacing the nearest neighbor global appearance validation with a geometric validation procedure which would also provide an useful homography estimation. Research on this area is considered as future work.

Acknowledgments. This research was supported in part by ITESM under the Research Chairs CAT-144 and CAT-165.

References

1. Avidan, S.: Ensemble Tracking. In: Computer Vision and Pattern Recognition, pp. 494–501 (2005)
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-Up Robust Features (SURF). Comput. Vis. Image Underst. 110, 346–359 (2008)

3. Bouguet, J.-Y.: Pyramidal Implementation of the Lucas Kanade Feature Tracker. Description of the Algorithm (2000)
4. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: BRIEF: Binary Robust Independent Elementary Features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010)
5. Dinh, T.B., Vo, N., Medioni, G.: Context Tracker: Exploring Supporters and Distracters in Unconstrained Environments. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1177–1184 (June 2011)
6. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. In: Vitányi, P.M.B. (ed.) EuroCOLT 1995. LNCS, vol. 904, pp. 23–37. Springer, Heidelberg (1995)
7. Hare, S., Saffari, A., Torr, P.H.S.: STRUCK: Structured Output Tracking with Kernels. In: ICCV, pp. 263–270 (2011)
8. Kalal, Z., Matas, J., Mikolajczyk, K.: P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints. In: Conference on Computer Vision and Pattern Recognition (2010)
9. Kalal, Z., Mikolajczyk, K., Matas, J.: Forward-Backward Error: Automatic Detection of Tracking Failures. In: International Conference on Pattern Recognition (2010)
10. Lv, Q., Josephson, W., Wang, Z., Charikar, M., Li, K.: Multi-Probe LSH: Efficient Indexing for High-dimensional Similarity Search. In: Proceedings of the 33rd International Conference on Very Large Data Bases, pp. 950–961. ACM (2007)
11. Ozuysal, M., Calonder, M., Lepetit, V., Fua, P.: Fast Keypoint Recognition Using Random Ferns. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 448–461 (2010)
12. Rosin, P.L.: Measuring Corner Properties (1999)
13. Rosten, E., Drummond, T.W.: Machine Learning for High-Speed Corner Detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006)
14. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: An Efficient Alternative to SIFT or SURF. In: International Conference on Computer Vision, Barcelona (2011)
15. Viola, P.A., Jones, M.J.: Rapid Object Detection Using a Boosted Cascade of Simple Features. In: Computer Vision and Pattern Recognition, pp. 511–518 (2001)

Two Adaptive Methods Based on Edge Analysis for Improved Concealing Damaged Coded Images in Critical Error Situations

Alejandro Alvaro Ramírez-Acosta[1] and Mireya S. García-Vázquez[2]

[1] MIRAL. R&D, 1047 Palm Garden, Imperial Beach, 91932 USA
[2] Instituto Politécnico Nacional, Unidad CITEDI, Tijuana, B.C. México 22510
ramacos10@hotmail.com, mgarciav@citedi.mx

Abstract. The original coded image signal can be affected when it is transmitted over error-prone networks. Error concealment techniques for compressed image or video attempt to exploit correctly received information to recover corrupted regions that are lost. If these regions have edges, most of these conventional approaches cause noticeable visual degradations, because they not consider the edge characteristics of images. The spatial error concealment methods cannot work well; especially over high burst error condition since a great of neighboring information have been corrupted or lost (called 'critical error situations'). This paper proposes two adaptive and effective methods to select the required support area, based on edge analysis using local geometric information, suited base functions and optimal expansion coefficients, in order to conceal the damaged macroblocks in critical error situations. Experimental results show that the proposed two approaches outperform existing methods by up to 7.9 dB on average.

Keywords: Spatial error concealment, adaptive directional interpolation, adaptive frequency selective extrapolation, critical support area, H.264/AVC, image, video.

1 Introduction

The transmission over error-prone networks of still images or videos coded by block based techniques like JPEG and MPEG respectively, may lead to block loss degrading, particularly the visual quality of images. Working under this environment, such as wireless communication where retransmission may be not feasible, application of error concealment (EC) techniques is consequently required to reduce degradation caused by the missing information. These techniques attempt to exploit correctly received information to recover corrupted regions that are lost [1-3]. The main idea of EC is to utilize neighboring correctly received data of the current frame or the reference frame to recover the corrupted regions. If these regions have edges, most of these conventional approaches cause noticeable visual degradations, because they not consider the edge characteristics of images. Most of spatial error concealment algorithms consider the eight neighboring macroblocks are correctly received in the same

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 158–170, 2013.

image for restoring missing macroblock. These methods cannot work well; especially over high burst error condition since a great of neighboring information have been corrupted or lost (called 'critical error situations'). This paper proposes two adaptive and effective methods to select the required support area, based on edge analysis using local geometric information, suited base functions and optimal expansion coefficients, in order to conceal the damaged macroblocks in critical error situations. The idea is then that these methods can alleviate the disadvantages of the conventional methods effectively and provide better performance considering the critical situation of having at least one neighbor macroblock correctly received or already concealed of the considered lost or corrupted macroblock.

The remainder of this paper is organized as follows. Section 2 discusses the proposed new method based on Directional Interpolation approach; in section 3 discusses the other proposed new method based on Frequency Selective Extrapolation approach. Implementation, results and discussion are presented in section 4. Finally, in section 5, we draw conclusions and give suggestions for future work.

2 Proposed Method Based on Directional Interpolation

This well-known and highly influential ancient spatial error concealment technique was proposed in [4] as a non-normative algorithm in the MPEG-4 H.264/AVC standard [1,3,5,6]. It uses weighted averaging interpolation (WAI) of four pixel values located at vertically and horizontally neighboring boundaries of a damaged macroblock (MB) consisting usually of 16x16 pixels. This method causes noticeable visual degradation in the region including the edges (cf. section 4, figs. 3a-5a).

The Directional Interpolation (DI) which is applied for intra-frames [7,8], this is a spatial error concealment technique proposed by W. Kwok and H. Sun [8] which uses local geometric information extracted from the surroundings to detect edge components. Sobel gradient filter is employed to detect the edge in a simple and fast way. It has perfect performance in edge detection and noise restraining. This algorithm presupposes, as does WAI's method, that all eight neighboring MBs in the support area $\mathcal{R}$ (see fig.2) are usually available to apply error concealment process over the damage MB (missing area $\mathcal{L}$). If any of the neighboring MBs is not correctly received, the DI's method doesn't perform well. To alleviate the disadvantages of the DI's method effectively and provide better performance considering critical situations as having at least one neighbor of the considered lost or corrupted MB, we focus on the challenge of providing the necessary conditions to correctly estimate the edge direction and to suitably reconstruct the damaged MB with minimal information.

2.1 Computing Edge Direction

The edge direction of the lost MB is estimated considering only the neighboring correctly received MBs or concealed in the support area $\mathcal{R}$ as it shown in figure 2. The magnitude and angular direction are computing. The direction of the surviving gradient is quantized in steps corresponding to a 22.5°. The voting mechanism only takes into account those pixels at (i,j) which belong to the correctly received MBs or already concealed. After all the pixels in the surrounding neighborhood have "voted", the counter containing the largest value determines which direction to use in the interpolation.

2.2 MB Reconstruction Using Adaptive Interpolation

In order to compute the dominant edge of the support area $\mathcal{R}$, a series of one dimensional linear interpolations are carried out along the direction to obtain the pixel of the lost MB. If a projective line, representing dominant angular direction, passes through the pixels around the border of the horizontal MBs (B5,B7) and/or vertical MBs (B4,B6) which have not been correctly received, then the closest correctly received MB or concealed to the considered lost MB ($\mathcal{L}$), is selected instead. For instance, a dominant angular direction of 0° is shown in figure 1. For the pixel p(x,y) belonging to the lost MB, the dominant angular direction edgDir of $\mathcal{R}$ passes the boundary pixels of MBs B5 and B7, then MB B5 will be selected to replace B7 (fig. 1a). Similarly, under the same principle, in fig. 1b, MB B7 will replace B5. If MBs B5 and B7 are incorrectly received, then the algorithm will find the closest neighboring MB for each one from B1,B2,B0 and B3 (fig.1c). This method promotes efficiency in the condition that horizontal, vertical and diagonals neighbors are not available.

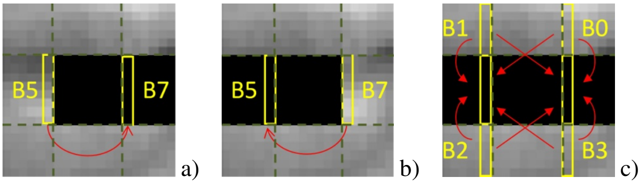


Fig. 1. Adaptive interpolation of the lost MB

The following represents, in generic form, the interpolation process to obtain the pixel values of the lost MB taking into account only those MBs which have been correctly received or concealed:

For **angular direction 0°**:

$$p(x,y) = \frac{f_{(i,j)}^{D5} \times Distant_{(i,j)}^{D7} + f_{(i,j)}^{D7} \times Distant_{(i,j)}^{D5}}{Distant_{(i,j)}^{D5} + Distant_{(i,j)}^{D7}}\,, \quad \begin{cases} (x,y) \in lost\ intra\ MB \\ B5 \neq 0\,, B7 \neq 0 \end{cases} \tag{1}$$

where $f_{(i,j)}$ is the estimated pixel and $Distant_{(i,j)}$ is the distance between the estimated pixel and the missing one (x,y).

When any of the horizontal neighboring MBs to the lost MB ($\mathcal{L}$) is incorrectly received, the following equations are evaluated:

$$p(x,y) = f_{(i,j)}^{D5}\,, \quad \begin{cases} (x,y) \in lost\ intra\ MB \\ B5 \neq 0\,, B7 = 0 \end{cases} \tag{2}$$

$$p(x,y) = f_{(i,j)}^{D7}\,, \quad \begin{cases} (x,y) \in lost\ intra\ MB \\ B5 = 0\,, B7 \neq 0 \end{cases} \tag{3}$$

When both of the horizontal neighboring MBs to the lost MB ($\mathcal{L}$) are incorrectly received, the following equation is evaluated:

$$p(x,y) = \min_{(i,j)\in N} Dist\left(f_{(i,j)} \to p_{(x,y)}\right)\,, \quad \begin{cases} (x,y) \in lost\ intra\ MB \\ N = \{B1, B2, B0, B3\}\,, \forall B_i \neq 0 \end{cases} \tag{4}$$

For **angular direction 90**°:

$$p(x,y) = \frac{f_{(i,j)}^{D4} \times Distant_{(i,j)}^{D6} + f_{(i,j)}^{D6} \times Distant_{(i,j)}^{D4}}{Distant_{(i,j)}^{D4} + Distant_{(i,j)}^{D6}}\,, \quad \begin{cases} (x,y) \in lost\ intra\ MB \\ B4 \neq 0\,, B6 \neq 0 \end{cases} \tag{5}$$

When any or both of the vertical neighboring MBs to the lost MB ($\mathcal{L}$) are incorrectly received, a procedure similar to angular direction **0**° is performed.

For angular directions different to 0° and 90°, the two estimated pixel values $f_{(i,j)}^{DA}$, $f_{(i,j)}^{DB}$, $A, B \in k$, $k = \{0,1,\ldots 7\}$ which are needed to obtain the interpolated pixel value $p(x,y)$, is computed by the following equations:

Option 1: estimated pixel position $f_{(sas,x1)}^{Dk}$

$$x1 = \left(\frac{j + (sas - i)}{\tan\left(edgDir \times \left(rad(22.5^0)\right)\right)}\right), \quad \begin{cases} & x1 \geq sas \\ if & and \\ & x1 \leq sas + bs + 1 \end{cases} \tag{6}$$

⟹ The availability of the B4, B1 and B0 MBs is analyzed to obtain the estimated pixel value following the equations (they are 8 cases):

$$f_{(i,j)}^{Dk} = f_{(sas,x1)}^{D4D1D0}\,, \quad \{B4 \neq 0\,, B1 \neq 0\,, B0 \neq 0 \tag{7}$$

$$f_{(i,j)}^{Dk} = f_{(sas,x1)}^{D4D0}\,, \quad \begin{cases} B4 \neq 0\,, B1 = 0\,, B0 \neq 0 \\ if\ x1 == sas \\ \Longrightarrow\ x1 = sas + 1 \end{cases} \tag{8}$$

$$f_{(i,j)}^{Dk} = f_{(sas,x1)}^{D4D1}\,, \quad \begin{cases} B4 \neq 0\,, B1 \neq 0\,, B0 = 0 \\ if\ x1 == sas + bs + 1 \\ \Longrightarrow\ x1 = sas + bs \end{cases} \tag{9}$$

$$f_{(i,j)}^{Dk} = f_{(sas,x1)}^{D4}\,, \quad \begin{cases} B4 \neq 0\,, B1 = 0\,, B0 = 0 \\ if\ x1 == sas \\ \Longrightarrow\ x1 = sas + 1 \\ elseif\ x1 == sas + bs + 1 \\ \Longrightarrow\ x1 = sas + bs \end{cases} \tag{10}$$

$$f_{(i,j)}^{Dk} = f_{(sas,x1)}^{D0}, \begin{cases} B4 = 0\,, B1 = 0\,, B0 \neq 0 \\ if\ x1 \geq sas\ \ and\ x1 < 2 \times sas \\ \implies\ x1 = x1 + bs + 1 \\ elseif\ x1 \geq 2 \times sas\ \ and\ \ x1 \leq sas + bs \\ \implies\ x1 = sas + bs + 1 \end{cases} \tag{11}$$

$$f_{(i,j)}^{Dk} = f_{(sas,x1)}^{D1}, \begin{cases} B4 = 0\,, B1 \neq 0\,, B0 = 0 \\ if\ x1 > sas\ \ and\ x1 \leq 2 \times sas \\ \implies\ x1 = x1 - bs + 2 \\ elseif\ x1 > 2 \times sas\ \ and\ \ x1 \leq sas + bs + 1 \\ \implies\ x1 = sas \end{cases} \tag{12}$$

$$f_{(i,j)}^{Dk} = f_{(sas,x1)}^{D1D0}, \begin{cases} B4 = 0\,, B1 \neq 0\,, B0 \neq 0 \\ if\ x1 > sas\ \ and\ x1 \leq sas +\ (bs/2) \\ \implies\ x1 = x1 - (bs/2) \\ elseif\ x1 > sas +\ (bs/2)\ and\ \ x1 < sas + bs + 1 \\ \implies\ x1 = x1 + (bs/2) \end{cases} \tag{13}$$

If the B4, B1 and B0 neighboring MBs are incorrectly received or concealed, then the neighboring MBs B6, B2 and B3 will be considered to obtain the estimated pixel value. To do this, the equations 7 to 13 are evaluated replacing the following variables: B4 by B6, B1 by B2, B0 by B3; D4 by D6, D1 by D2, D0 by D3 and $f_{(sas,x1)}^{Dk}$ by $f_{(sas+bs+1,x1)}^{Dk}$.

Option 2: estimated pixel position $f_{(sas+bs+1,x2)}^{Dk}$

$$x2 = \left(\frac{j + (sas + bs + 1 - i)}{\tan\left(edgDir \times \left(rad(22.5^{0})\right)\right)}\right), \begin{cases} & x2 \geq sas \\ if & and \\ & x2 \leq sas + bs + 1 \end{cases} \tag{14}$$

⟹ The availability of the B6, B2 and B3 MBs is analyzed to obtain the estimated pixel value. A procedure similar to option 1 is performed, following the equations 7 to 13 and replacing the corresponding variables.

Option 3: estimated pixel position $f_{(y3,sas)}^{Dk}$

$$y3 = \left(i + (sas - j)\right) \times \tan\left(edgDir \times \left(rad(22.5^{0})\right)\right), \begin{cases} & y3 > sas \\ if & and \\ & y3 < sas + bs + 1 \end{cases} \tag{15}$$

⟹ The availability of the B5, B1 and B2 MBs is analyzed to obtain the estimated pixel value. The considerations of the option 1 are also realized.

Option 4: estimated pixel position $f_{(y4,sas+bs+1)}^{Dk}$

$$y4 = \left(i + (sas + bs + 1 - j)\right) \times \tan\left(edgDir \times \left(rad(22.5^{0})\right)\right), \begin{cases} & y4 > sas \\ if & and \\ & y4 < sas + bs + 1 \end{cases} \tag{16}$$

⟹ The availability of the B7, B0 and B3 MBs is analyzed to obtain the estimated pixel value. The considerations of the option 2 are also realized.

After obtention of the two estimated pixel values, the interpolated pixel value of the lost MB is computed by the following equation:

$$p(x,y) = \frac{f_{(i,j)}^{DA} \times Distant_{(i,j)}^{DB} + f_{(i,j)}^{DB} \times Distant_{(i,j)}^{DA}}{Distant_{(i,j)}^{DA} + Distant_{(i,j)}^{DB}}\,, \quad \{(x,y) \in lost\ intra\ MB \quad (17)$$

where $f_{(i,j)}^{DA}$, $f_{(i,j)}^{DB}$ are the estimated pixels and $Distant_{(i,j)}^{DA}$, $Distant_{(i,j)}^{DB}$ are the distances between the estimated pixels and the missing one (x,y).

3 Proposed Method Based in Frequency Selective Extrapolation

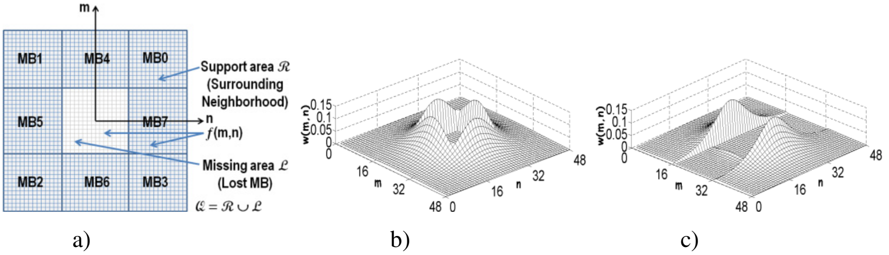
For the images EC application, Kaup et al [9-11] proposed a frequency selective extrapolation (FSE) approach based on successive approximation technique [12,13]. So, the image content (see fig. 2a) of the known macroblocks (support area $\mathcal{R}$) is successively approximated through a parametric model $g(m,n)$ and the missing macroblock (missing area $\mathcal{L}$) is obtained by extrapolation according to an error criterion based on the energy weighted function, described as follows:

$$E_{\mathcal{R}} = \sum_{(m,n)\in\mathcal{Q}} w(m,n)[f(m,n) - g(m,n)]^2\,, \quad \forall\{Bk \in \mathcal{R}\} \tag{18}$$

where $w(m,n)$ is the weighting function, $f(m,n)$ are the values of the samples in the area $\mathcal{Q}$, $g(m,n)$ is the parametric model and Bk is the MB in the support area $\mathcal{R}$. The parametric model in each iteration is:

$$g^{(v)}(m,n) = \sum_{(p,q)\in\mathcal{P}_v} c_{p,q}^{(v)} \varphi_{p,q}(m,n)\,, \quad \{m,n \in \mathcal{Q}\} \tag{19}$$

$\mathcal{P}_v$ denoting the set of basis functions $\varphi_{p,q}(m,n)$ weighted by the expansion coefficients $c_{p,q}^{(v)}$ used in the iteration v, m,n indicates the row and column index. The number of available basis functions equals the number of samples in the entire area $\mathcal{Q}$.



a) b) c)

Fig. 2. a) Representation of missing and support area. Weighting function: b) Isolated lost block, c) Consecutive lost block.

The FSE's method which is applied for intra-frames [9-11], takes all eight surrounding MBs (support area $\mathcal{R}$), to restore the lost one (missing area $\mathcal{L}$). If any of the neighboring MBs is not correctly received, the FSE's method doesn't perform well (cf. section 4, figs. 3d-5d). To alleviate the disadvantages of the FSE's method effectively and provide better performance considering critical situations as having at least one correctly received or concealed neighbor MB, we focus on the challenge of providing the necessary conditions to correctly estimate the signal in the available support area $\mathcal{R}$ through a suited base functions and an optimal expansion coefficients to suitably reconstruct the damaged MB with minimal information.

3.1 Computing Basis Functions and Expansion Coefficients

For emphasizing closer regions to missing area $\mathcal{L}$, Kaup et al [9-11] use a weighting function $w(m,n)$ (see eq.18) based on an isotropic model $\rho(m,n)$.

$$w(m,n) = \begin{cases} \rho(m,n) = \hat{\rho}^{\sqrt{\left(m-\frac{M}{2}\right)^2+\left(n-\frac{N}{2}\right)^2}}, & 0<\hat{\rho}<1, \{m,n \in \mathcal{R}, \forall Bk \in \mathcal{R}\} \\ 0, & \{m,n \in \mathcal{L}\} \end{cases} \tag{21}$$

Figure 2b depicts the resulting weighting function for a single lost MB. This lost MB is located in the center of this weighting function having zero intensities and all surrounding MB's area has a certain weight defined by ρ.

Kaup et al [9-11], also treat the consecutive macroblock loss. The resulting weighting function $w(m,n)$ on the available support area $\mathcal{R}$, for the case of consecutive macroblock loss, is shown in figure 2c. In this figure, the macroblock MB7 is not available and macroblock MB5 has been concealed (extrapolated). Then, in order to include the MB5 in the concealment procedure, Kaup et al [9-11] limit their influence by assigning a weight of 0.1 trying to prevent the spread of approximation errors. It is known that the influence of the weighting function decreases radial symmetrically with distance from the center of the lost area at $(M/2, N/2)$, then, we compute the contribution of each sub-area (surrounding MBs) in the support area $\mathcal{R}$ in terms of energy: $MB0 = MB1 = MB2 = MB3 = 0.14096 \times 10^{-4}\ J/Hz$, and for horizontal and vertical macroblocks: $MB4 = MB5 = MB6 = MB7 = 1.6858 \times 10^{-4}\ J/Hz$. Based on this results, we propose to use the information of the concealed neighbors MB's throughout the support area $\mathcal{R}$, under the same conditions as the correctly received MB's. So, we apply the weighting function in the area of the concealed MB without limiting its influence. Our FSE algorithm optimization allows the concealment of the lost MB for any error pattern structure present in the Intra frame. The implemented optimization algorithm is described as follows:

1. **Establishment of the Weighting Function**

In the proposed method, the resulting weighting function is variable because it depends on the available support area $\mathcal{R}$ for each lost MB that needs to be concealed. This condition can be represented as follows:

$$w_{Bk}(m,n) = \begin{cases} \rho(m,n) = \hat{\rho}^{\sqrt{\left(m-\frac{M}{2}\right)^2+\left(n-\frac{N}{2}\right)^2}}, & \begin{matrix} 0<\hat{\rho}<1, \{m,n \in \mathcal{R}\} \\ \{\forall Bk \in \mathcal{R} \wedge Bk \neq 0\} \end{matrix} \\ 0, & \{m,n \in \mathcal{L}\} \end{cases} \tag{22}$$

Bk is the MB correctly received or concealed in the variable support area $\mathcal{R}$.

2. **Initialization of the Parametric Model**

The parametric model is initialized as Kaup et al [9-11], $g^{(0)}(m,n) = 0$.

3. **Initialization of the Approximation Weighted Residual Error**

The initialization $\{v = 0\}$ of the approximation weighted residual error is done by the following:

$$r_{w_{Bk}}^{(v)}(m,n) = w_{Bk}(m,n) \times r^{(v)}(m,n), \quad \{m,n \in \mathcal{R}\}, \forall\{Bk \in \mathcal{R} \wedge Bk \neq 0\} \tag{23}$$

where

$$r^{(v)}(m,n) = \left[f(m,n) - g^{(v)}(m,n)\right] \tag{24}$$

4. **Iterative Procedure of Frequency Successive Approximation**

4.1. Best Fitting Basis Function Determination

Based on the equation 18 as well as the conditions presented in equations 22 and 23, and using the 2D Discrete Fourier Transform (DFT) base functions, the maximal decrease of the weighted error criterion to select the best fitting basis function in the frequency domain can be expressed as follows:

$$\Delta E_{\mathcal{R}_{Bk}}^{(v)} = \frac{\boldsymbol{R}_{w_{Bk}}^{(v)}[p,q]^2}{\boldsymbol{W}_{Bk}[0,0]}, \quad \forall\{Bk \in \mathcal{R} \wedge Bk \neq 0\} \tag{25}$$

4.2. Expansion Coefficients Determination

After doing some mathematical simplifications from equation 18, the expansion coefficient update Δc for each iteration v can expressed in the frequency domain as:

$$\Delta c = MN\frac{\boldsymbol{R}_{w_{Bk}}^{(v)}[u,v]}{\boldsymbol{W}_{Bk}[0,0]}, \quad \forall\{Bk \in \mathcal{R} \wedge Bk \neq 0\} \tag{26}$$

The expansion coefficient $c_{u,v}^{(v+1)}$ is then update by

$$c_{u,v}^{(v+1)} = c_{u,v}^{(v)} + \Delta c \tag{27}$$

4.3. Updating the Parametric Model

The parametric model $g^{(v)}(m,n)$: $\boldsymbol{G}^{(v)}(m,n)$ is updated for each iteration v based on the proposed conditions mentioned before, allowing obtain suitable base functions and optimal expansion coefficients to correctly estimate the available support area $\mathcal{R}$.

4.4. New Approximation Error Determination

To obtain the weighted residual error signal in the next iteration $v+1$, the following equation is computed:

$$\boldsymbol{R}_{w_{Bk}}^{(v+1)}[p,q] = \boldsymbol{R}_{w_{Bk}}^{(v)}[p,q] - \frac{1}{MN}\Delta c\boldsymbol{W}_{Bk}[p-u,q-v], \quad \forall\{Bk \in \mathcal{R} \wedge Bk \neq 0\} \tag{28}$$

5. The Final Parametric Model

After all iterations are done, the final parametric model is obtained by an inverse DFT. This parametric model is then, the closest approximation to the signal data in the available support area $\mathcal{R}$:

$$g[m,n] = IDFT_{M,N}\{G[p,q]\} \tag{29}$$

Finally, the proper concealed lost MB is obtained from the parametric model which is optimized for the available support area $\mathcal{R}$.

4 Results

In order to evaluate the performance of the spatial concealment methods, experiments were conducted using corrupted representative Intra Images with significant losses (20% to 42%) due to node congestion or excessive delay in mobile communications. These experiments also include different errors distributions. It is important to highlight that, in real communications, consecutive blocks corrupted usually happen and that the select images are a major challenge for the EC algorithms. We compared the performances of the following spatial error concealment methods: WAI method [4,14], directional interpolation method [8,15], FSE method with all MBs correctly received [9-11], FSE method for consecutive macroblock loss (FSE weighted 0.1) [9-11] and ours proposed methods. In order to evaluate the quality of reconstruction of an image Intra we use the peak to signal-to-noise ratio of its YUV color space luminance component (Y-PSNR). According to the results shown in the Table 1, in general, the proposed methods outperforms existing methods by up to 7.9 dB on average.

Table 1. Performance of the spatial error concealment methods

Image Intra	Error distribution	% losses	WAI (dB)	DI (dB)	Our method DI (dB)	FSE (all MBs) (dB)	FSE weighted 0.1 (dB)	Our method FSE (dB)
Lena	Bursty	42.48	22.82	10.99	**26.87**	10.93	26.37	26.59
	Checkerboard	41.02	23.13	25.86	**28.44**	23.42	---	28.34
	Uniform	21.97	25.83	31.29	**31.29**	30.71	---	30.71
Baboon	Bursty	42.48	21.15	10.39	21.96	10.43	22.21	**22.35**
	Checkerboard	41.02	21.35	22.06	22.70	21.24	---	**23.13**
	Uniform	21.97	24.02	25.58	25.58	26.04	---	**26.04**
Foreman	Bursty	38.38	22.65	9.01	26.66	9.52	26.84	**27.25**
	Checkerboard	35.88	23.06	28.01	**31.47**	23.52	---	30.09
	Uniform	20.20	26.00	32.90	**32.90**	32.67	---	32.67

Figures 3-6 show some concealed images Intra applying the spatial error concealment methods discussed in this paper. The error distributions are the bursty and checkerboard. As shown in these figures, the best spatial error concealment methods (proposed methods) yield the best performances according to Y-PSNR criteria and the visual quality. This can be explained by the fact that ours methods takes into account only the correctly received neighboring MBs or concealed neighboring MBs to compute the edge angular direction and the frequency selective extrapolation. Moreover, it selects the proper neighboring MBs according to the formulation done in sections 2.1, 2.2 and 3.1. On the other hand, the WAI's method works well in regions with smooth texture area. However, as we can see in the figures, it causes important visual degradations in the region including edges. The conventional DI's method presents good results when vertical and horizontal neighboring MBs have been correctly received (uniform error distribution). The conventional FSE's method [9-11] presents good results when all neighboring MBs have been correctly received (uniform error distribution). On the contrary, these two last methods show catastrophic results taking into account all eight neighboring MBs without considering their availability to reconstruct the lost MB.

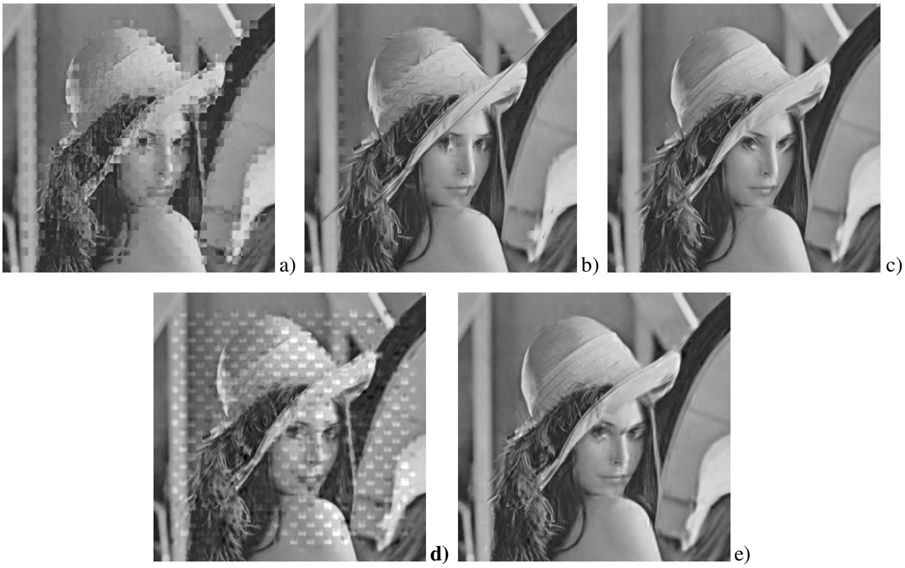


Fig. 3. Error distribution: checkerboard. Concealed image Intra of "Lena" with the: a) WAI, Y-PSNR=23.13dB; b) Directional Interpolation, Y-PSNR=25.86dB; c) Our method DI, Y-PSNR=28.44dB; d) FSE all 8 MB's, Y-PSNR=23.42dB; e) Our method FSE, Y-PSNR=28.34dB.

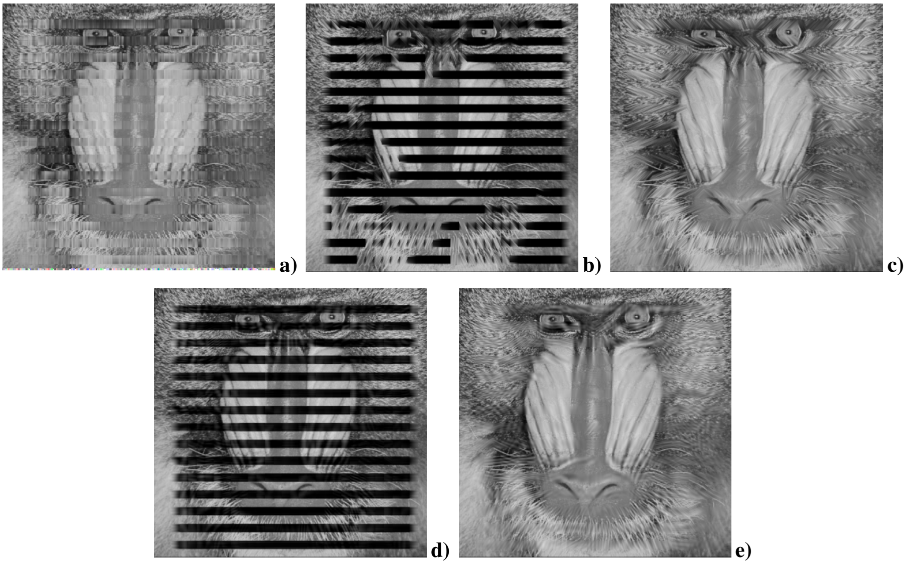


Fig. 4. Error distribution: bursty. Concealed image Intra of "Baboon" with the: a) WAI, Y-PSNR=21.15dB; b) Directional Interpolation, Y-PSNR=10.39dB; c) Our method DI, Y-PSNR=21.96dB; d) FSE all 8 MB's, Y-PSNR=10.43dB; e) Our method FSE, Y-PSNR=22.35dB.

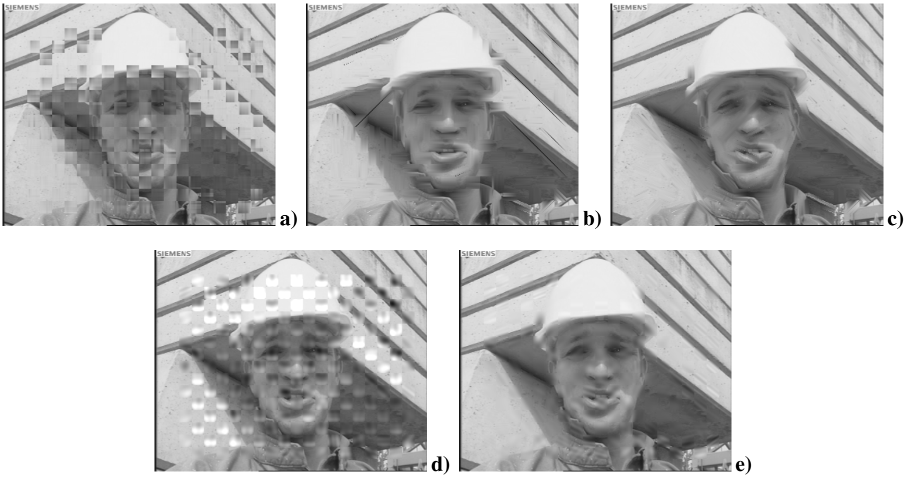


Fig. 5. Error distribution: checkerboard. Concealed image Intra of "Foreman" with the: a) WAI, Y-PSNR=23.06dB; b) Directional Interpolation, Y-PSNR=28.01dB; c) Our method DI, Y-PSNR=31.47dB; d) FSE all 8 MB's, Y-PSNR=23.52dB; e) Our method FSE, Y-PSNR=30.09dB.

For consecutive macroblock loss, our method FSE has on average of Y-PSNR=0.26 dB which is superior to FSE weighted 0.1 [9-11] (see Table 1). The figure 6 shows that the FSE weighted 0.1 method generates blocking effects, this is due to the weight assigned of 0.1 to MB5 in the support area. It is also shown that our method FSE has better performance in avoiding error propagation.

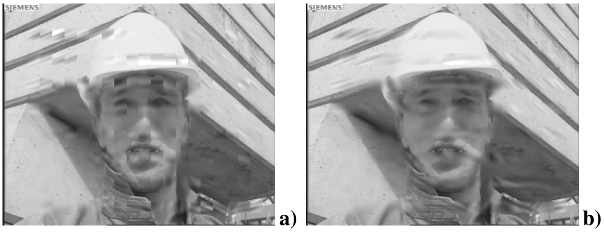


Fig. 6. Error distribution: bursty. Concealed image Intra of "Foreman" with the: a) FSE weighted 0.1, Y-PSNR=26.84dB; b) Our method FSE, Y-PSNR=27.25dB.

5 Conclusions

We investigate how to resolve the problem of significant loss of information (20% to 42% of errors with different distributions) due to node congestion or excessive delay in mobile communications. Thus, in this paper we proposed two adaptive and effective methods to select the required support area, based on edge analysis, suited base functions and optimal expansion coefficients, in order to conceal the damaged macroblocks in critical error situations. These methods performed well in current environments, where the images Intra are corrupted with different errors distributions covering up to 42% of the image area. Compared with four spatial error concealment algorithm, WAI [4,14], Directional Interpolation [8,15], frequency selective extrapolation with all 8 MB's [9-11] and frequency selective extrapolation with weighted 0.1 [9-11], the adaptive DI and the adaptive FSE proposed techniques have the best results against several errors distribution, according to the metric Y-PSNR. The proposed methods provide better performance considering the critical situation of having at least one neighbor macroblock correctly received or already concealed of the considered lost or corrupted macroblock. The proposed methods can be combined with temporal replacement algorithms to provide improved error concealment for block-based video sequence coding. The proposed algorithms are implemented in Matlab®, so they can be applied in real-time video transition, of course, after corresponding accelerations to functions such as FFT.

Acknowledgment. This research was supported by grant SIP2012 from IPN.

References

1. Richardson, I.E.G.: H.264 and MPEG-4 Video Compression, Video Coding for Next–generation Multimedia. John Wiley & Sons, Ltd. (2004)
2. Wang, Y., Zhu, Q.-F.: Error Control and Concealment for Video Communication: A Review. Proceedings of the IEEE 86(5), 985–995 (1998)

3. Kumar, S., Xu, L., Mandal, M.K., Panchanathan, S.: Error Resiliency Schemes in H.264/AVC standard. Visual Communications & Image Representation 17(2) (2006)
4. Varsa, V., Hannuksela, M.M., Wang, Y.K.: Non-normative error concealment algorithms. ITU-T VCEG-N62 (September 2001)
5. Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG: Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 –ISO/IEC 14496-10 AVC), Doc. JVT-G050r1 (May 2003)
6. Wiegand, T., Sullivan, G.J., Bjøntegaard, G., Luthra, A.: Overview of the H.264/AVC Video Coding Standard. IEEE Trans. Circ. and Syst. for Vid. Tech. 13(7) (July 2003)
7. Nemethova, O., Al-moghrabi, A.: Flexible Error Concealment for H.264 Based on Directional Interpolation, Wir. Networks. In: Int. Conf. on Comm. and Mob. Comp. (2005)
8. Kwok, W., Sun, H.: Multi-Directional Interpolation for Spatial Error Concealment. IEEE Trans. on Cons. Electr. 39(3), 455–460 (1993)
9. Kaup, A., Meisinger, K., Aach, T.: Frequency selective signal extrapolation with applications to error concealment in image communication. Int. J. of Elect. Comm. 59, 147–156 (2005)
10. Meisinger, K., Kaup, A.: Minimizing a weighted error criterion for spatial error concealment of missing image data. In: Proc. IEEE Int. Conf. on Image Proc., ICIP 2004, pp. 813–816 (2004)
11. Meisinger, K., Kaup, A.: Spatial error concealment of corrupted image data using frequency selective extrapolation. In: ICASSP, pp. 209–212 (May 2004)
12. Kaup, A., Aach, T.: Efficient prediction of uncovered background in inter frame coding using spatial extrapolation. In: ICASSP, pp. 501–504 (April 1994)
13. Kaup, A., Aach, T.: Coding of segmented images using shape independent basis functions. IEEE Trans. Image Process. 7, 937–947 (1998)
14. H.264/AVC Codec Software: JM14.2 Video Coding Standard
15. Suh, J.W., Ho, Y.S.: Error concealment based on directional interpolation. IEEE Trans. Consumer Electron. 43(3), 295–302 (1997)

Phase Correlation Based Image Alignment with Subpixel Accuracy

Alfonso Alba*, Ruth M. Aguilar-Ponce,
Javier Flavio Vigueras-Gómez, and Edgar Arce-Santana

Facultad de Ciencias, Universidad Autónoma de San Luis Potosí,
Av. Salvador Nava Mtz. S/N, Zona Universitaria,
78290, San Luis Potosí, SLP, México
fac@fc.uaslp.mx, {rmariela,arce}@fciencias.uaslp.mx, flavio@fc.uaslp.mx

Abstract. The phase correlation method is a well-known image alignment technique with broad applications in medical image processing, image stitching, and computer vision. This method relies on estimating the maximum of the phase-only correlation (POC) function, which is defined as the inverse Fourier transform of the normalized cross-spectrum between two images. The coordinates of the maximum correspond to the translation between the two images. One of the main drawbacks of this method, in its basic form, is that the location of the maximum can only be obtained with integer accuracy. In this paper, we propose a new technique to estimate the location with subpixel accuracy, by minimizing the magnitude of gradient of the POC function around a point near the maximum. We also present some experimental results where the proposed method shows an increased accuracy of at least one order of magnitude with respect to the base method. Finally, we illustrate the application of the proposed algorithm to the rigid registration of digital images.

1 Introduction

The phase correlation method [1] is a frequency domain technique used to estimate the delay or shift between two copies of the same signal. This technique is based on the shift properties of the Fourier transform. Specifically, consider two discrete periodic signals $f(x)$ and $g(x)$, and let $F(\omega)$ and $G(\omega)$ be their respective Fourier transforms. The normalized cross-spectrum $R(\omega)$ of f and g is given by

$$R(\omega) = \frac{F(\omega)G^*(\omega)}{|F(\omega)G^*(\omega)|}, \tag{1}$$

where G^* is the complex conjugate of G. Note that $|R(\omega)| = 1$ for all ω. Also, the phase-only correlation (POC) function $r(x)$ is defined as the inverse Fourier transform of $R(\omega)$.

Now suppose g is simply a delayed copy of f; that is, $g(x) = f(x + d)$, where d is an unknown integer. The shift property of the Fourier transform states

* A. Alba was supported in part by Conacyt grant 154623.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 171–182, 2013.

that $G(\omega) = F(\omega)\exp\{j\omega d\}$, where $j = \sqrt{-1}$. In this case, it is easy to see that $R(\omega) = \exp\{-j\omega d\}$ and $r(x) = \delta(x-d)$, where δ is the discrete impulse function (i.e., $\delta(0) = 1$ and $\delta(x) = 0$ for $x \neq 0$). Therefore, one can recover d by simply locating the maximum of $r(x)$.

This method can be easily extended to 2D and 3D images, and has been successfully applied in several image processing and computer vision problems, such as image registration [2], [3], [4], [5], biometrics [6], [7], [8], stereo disparity estimation [9] [10], motion and optical flow estimation [11], [10], and video encoding [12], [10].

One of the most important drawbacks of the phase correlation method, at least in its basic form, is that the recovered displacements have integer accuracy; i.e., the coordinates of the maximum of the discrete POC function will be a rounded version of the components of the true displacement vector. Various alternatives have been devised to estimate the displacements with non-integer (subpixel) accuracy. Among the most popular are those which rely on local function fitting: one can first obtain the displacement d_0 with integer accuracy using the basic phase correlation method and fit a simple analytical function $f(d)$ (e.g., a polynomial) to the POC values in a neighborhood of d_0; then one maximizes $f(d)$ to estimate the true maximum. The most common fitting functions are quadratic polynomials and Gaussian functions [13], cubic splines [6], and Dirichlet or sinc functions [14], [15], [11]. Most of these methods perform reasonably well under controlled conditions but their performance is seriously degraded by noise, border effects, and the presence of multiple motions. This limits the application of these methods to many computer vision problems, such as stereo depth or optical flow estimation.

In this paper, we introduce a new method for the estimation of POC maxima with subpixel accuracy, which is based on finding approximate zeros of the gradient of the POC function. The proposed method, which is presented in Section 2, is capable of high-accuracy estimations while maintaining adequate robustness to noise and multiple motions. In Section 3, we use synthetic data to demonstrate the advantages of our method for the estimation of rigid image transformations. Finally, our conclusions are presented in Section 4.

2 Methodology

Instead of relying on local function fitting, the proposed method attempts to estimate the POC maxima by finding approximate zeros of the POC gradient near the integer-valued displacements. Throughout rest of the article, we will use square brackets to denote N-periodic discrete-time signals (e.g., $f[x]$, where $x = 0, \ldots, N_x - 1$), and parentheses for continuous-time signals (e.g., $f(x)$ with $x \in \mathbb{R}$). Also note that in most cases we will be dealing with periodic signals and will obviate the need of mod-N indexing (e.g., $f[x + N] = f[x]$).

2.1 Gradient Estimation of the POC Function

Consider again the 1D case of the POC function $r[x]$, which is defined as the inverse discrete Fourier transform of the normalized cross-correlation $R[k]$; in other words,

$$r[x] = \sum_{k=0}^{N-1} R[k] \exp\left\{2\pi jkx/N\right\}, \tag{2}$$

where N is the period of the signals.

The right-hand side of the above equation provides, in fact, a band-limited, continuous representation of the phase correlation function. By differentiating this expression with respect to x, one can obtain an analytical, continuous expression of the derivative of the POC function $r'(x)$, which is given by

$$r'(x) = j\frac{2\pi}{N} \sum_{k=0}^{N-1} kR[k] \exp\left\{2\pi jkx/N\right\}. \tag{3}$$

Note that, for real-valued input signals $f(x)$ and $g(x)$, the POC function is also real, so its derivatives must be real as well. This means one can also compute $r'(x)$ as

$$r'(x) = -\frac{2\pi}{N} \sum_{k=0}^{N-1} k \ \mathfrak{Im}\left\{R[k] \exp\left\{2\pi jkx/N\right\}\right\}, \tag{4}$$

which in some cases may be computationally more efficient.

The problem of finding the extrema of the POC function with sub-pixel accuracy is equivalent to finding the zeros of $r'(x)$. In 1D, the approach is straightforward: use the integer-valued displacement obtained from the discrete POC function as a starting point for a root-finding algorithm such as bisection or Newton-Raphson (the second derivative of the POC, required for Newton-Raphson, is also easy to obtain, although somewhat unstable). However, generalizing this idea to 2D (or higher dimensions) carries some difficulties. Since the POC function is now bivariate (e.g., $r(x, y)$), its derivative takes the form of a vector-valued bivariate gradient function, given by

$$\nabla r(x, y) = \left[\frac{\partial r}{\partial x}(x, y),\ \frac{\partial r}{\partial y}(x, y)\right], \tag{5}$$

with

$$\frac{\partial r}{\partial x}(x, y) = -\frac{2\pi}{N_x} \sum_{l=0}^{N_y-1} \sum_{k=0}^{N_x-1} k \ \mathfrak{Im}\left\{R[k, l] \exp\left\{2\pi j\left(\frac{kx}{N_x} + \frac{ly}{N_y}\right)\right\}\right\}, \tag{6}$$

and

$$\frac{\partial r}{\partial x}(x, y) = -\frac{2\pi}{N_y} \sum_{l=0}^{N_y-1} \sum_{k=0}^{N_x-1} l \ \mathfrak{Im}\left\{R[k, l] \exp\left\{2\pi j\left(\frac{kx}{N_x} + \frac{ly}{N_y}\right)\right\}\right\}, \tag{7}$$

where N_x and N_y are, respectively, the width and height of the input images.

Instead of finding the zeros of $\nabla r(x,y)$, one can approximate them by minimizing the real-valued magnitude of the gradient. In other words, the estimated displacement (d_x, d_y) is given by

$$(d_x, d_y) = \arg \min_{(x,y)\in\mathcal{N}(x_0,y_0)} h(x,y), \tag{8}$$

where $h(x,y) = |\nabla r(x,y)|^2$, (x_0, y_0) is a suitable initial solution (e.g., the integer-valued displacement estimation), and $\mathcal{N}(x_0, y_0)$ is a neighborhood centered at this point.

2.2 Minimization of Gradient Magnitude

Equation 3 shows that the spectrum of the derivative of the POC function is obtained as the normalized cross-spectrum $R[k]$ multiplied by the frequency k. This means that $r'(x)$ will, in general, have an increased high-frequency content with respect to $r(x)$. If one were to compute the second derivative $r''(x)$, it would show even more high-frequency content, making it very sensitive to noise and border effects. For this reason, we have chosen to avoid methods based on the derivatives of $h(x,y)$ (i.e., the second-order derivatives of the POC function) such as gradient or Newton descent methods. Instead, we have chosen two methods which rely only on the evaluation of the function to be optimized.

Grid Search. The initial solution (x_0, y_0) is refined by evaluating $h(x,y)$ at the nodes of a $(2n+1)\times(2n+1)$ grid centered at (x_0, y_0), with a spacing of w units between adjacent nodes. The node $(\hat{x}, \hat{y})$ with the lowest h is chosen as the best solution. Further refinement is obtained by reducing w and iterating the method with $(\hat{x}, \hat{y})$ as the new center point. We have obtained good results after 2 or 3 iterations with n between 5 and 10. The initial spacing is $w = 1/(2n)$ so that the initial search area covers exactly 1 pixel; after each iteration, w is divided by n, restricting the search to an area equivalent to one cell of the previous grid.

Nelder-Mead Optimization. The Nelder-Mead method [16] is a well-known minimization heuristic which relies only on the evaluation of the function to be minimized. In order to minimize a function f of n variables, the method requires $n+1$ points, $x_1, \dots, x_{n+1} \in \mathbb{R}^n$ forming a simplex (i.e., no two points can be colinear). The method works iteratively by selecting the worst point in the set; that is, a point x_h such that $f(x_i) \leq f(x_h)$ for $i = 1, \dots, n$, and replacing it by a new, better point x_0, which lies along the line defined by x_h and the centroid of the remaining n points. Depending on how good the new point is with respect to the given points, it can be reflected, expanded, or contracted along the line (see [16]). When applying this method to minimize the POC gradient $h(x,y)$ one requires 3 starting points; the first one is the integer solution (x_0, y_0), and the remaining two were chosen as $(x_0 \pm 0.5, y_0)$ and $(x_0, y_0 \pm 0.5)$ where the sign was selected depending on which neighbor was best. The results were very similar to the grid search method; however, many less points were evaluated with the

Nelder-Mead approach, making it computationally more efficient. The reflection, expansion, and contraction coefficients we used were $\alpha = 1$, $\gamma = 2$, and $\beta = 0.5$, respectively, and the method usually converged in less than 100 iterations.

2.3 Bandlimited Phase Correlation

Most of the methods which attempt to estimate the maxima of the POC function [14], [17], [15] with subpixel accuracy are seriously affected by noise, aliasing, and border effects, which are mostly present in the higher components of the frequency spectrum of the signals. Therefore, it is strongly suggested to limit the bandwidth of the POC function prior to the estimation of the POC maxima. In many cases, a simple ideal lowpass filter is sufficient and very easy to implement, since it only requires zeroing those coefficients in $R[k]$ corresponding to higher frequencies, before taking its inverse Fourier transform. In the proposed method, we can apply this filter directly in the estimation of the partial derivatives (Eqs. 6 and 7) by summing only those terms with $k < \kappa N_x$ and $l < \kappa N_y$, where κ represents the cutoff frequency of the filter and must be between 0 and 0.5. In our tests, good results were obtained with $\kappa \approx 0.3$.

2.4 Application to Rigid Image Registration

To demonstrate the applicability of the proposed method, we have implemented a rigid image registration algorithm based on the one proposed by Reddy et al. [3], with the exception that the POC maxima are estimated with sub-pixel accuracy. By definition, a rigid transformation is composed only of rotations and translations; however, the proposed algorithm can also deal with isotropic scalings (equal scaling along all axes), and could be easily extended to 3D images (e.g., MRI volumes). This algorithm can be summarized in the following steps (see [3] for details):

1. Let I_1 and I_2 be the input images of equal size, where I_2 is assumed to be a rigidly transformed version of I_1.
2. Compute the discrete Fourier transform $\hat{I}_k$ of each input image I_k.
3. Compute the log-magnitude of the spectra M_k of each image as

$$M_k[k,l] = W[k,l] \log \left| \hat{I}_k[k,l] \right|, \tag{9}$$

   where $W[k,l]$ is the frequency response of a high-pass filter. Here we use the one suggested by Reddy, which is given by $W[k,l] = (1 - X[k,l])(2 - X[k,l])$ with $X[k,l] = \cos(\pi k/N_x)\cos(\pi l/N_y)$ for $-N_x/2 \leq k < N_x/2$ and $-N_y/2 \leq l < N_y/2$. It can be shown that M_k is invariant to translation.
4. Transform M_k from cartesian coordinates $[x,y]$ to log-polar coordinates $[\rho,\theta]$, where $\rho = K\sqrt{x^2+y^2}$, $\theta = \text{atan2}(y,x)$, and K is an adequate scaling factor which controls the resolution of the radius (ρ) axis. We have obtained good results with $K = (N_x + N_y)/8$, where $N_x \times N_y$ is the size of the input images.

5. Compute the normalized cross-correlation $R_M[k,l]$ and the POC function $r_M[\rho,\theta]$ between $M_1[\rho,\theta]$ and $M_2[\rho,\theta]$.
6. Estimate the maximum of r_M with subpixel accuracy using the proposed method. The coordinates of this maximum correspond to the scaling and rotation parameters (s and ϕ, respectively) between I_1 and I_2.
7. Let $\tilde{I}_1[x,y] = I_1\left(T_{s,\phi}\{x,y\}\right)$, where $T_{s,\phi}$ is a transformation defined as

$$T_{s,\phi}\{x,y\} = \begin{bmatrix} s\cos\phi & -s\sin\phi \\ s\sin\phi & s\cos\phi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{10}$$

   In other words, $\tilde{I}_1$ is a rectified version of I_1, rotated and scaled according to the parameters found in the previous step.
8. Compute the normalized cross-correlation $R[k,l]$ and the POC function $r[x,y]$ between $\tilde{I}_1$ and I_2.
9. Estimate the maximum of r with subpixel accuracy. The coordinates of the maximum correspond to the translation between the input images.

3 Results and Discussion

In this section we present some results obtained with the application of our approach to different image registration situations. To evaluate the accuracy of the registrations, we take a reference image and perform an artificial transformation with known parameters. The reference and transformed images are then registered, and the estimated parameters are compared against the true ones using the True Mean Relative Error (TRME) [18], defined as

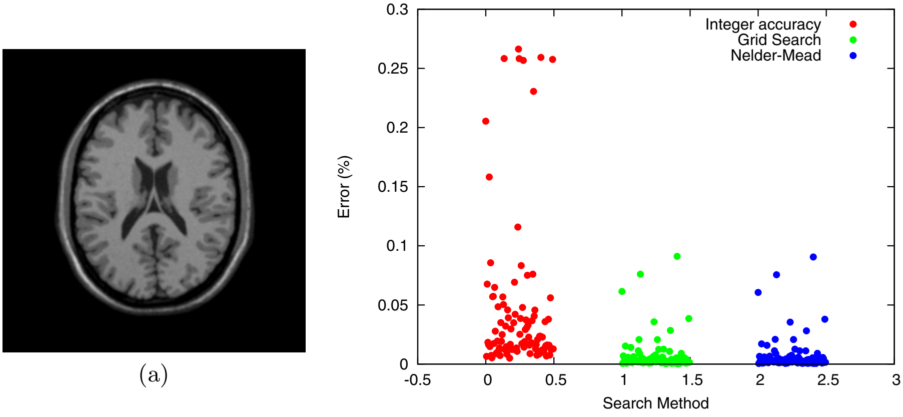
$$\text{TRME} = \frac{1}{4}\left[\frac{s-\hat{s}}{s} + \frac{\phi-\hat{\phi}}{\phi} + \frac{d_x-\hat{d}_x}{d_x} + \frac{d_y-\hat{d}_y}{d_y}\right], \tag{11}$$

where (s,ϕ,d_x,d_y) are the true parameters and $(\hat{s},\hat{\phi},\hat{d}_x,\hat{d}_y)$ the estimated ones. Note that this error measure is more sensitive to accuracy when the transformation parameters are relatively small.

3.1 Optimization Approaches

In the first set of experiments we compared the performance of the original phase correlation approach with integer accuracy, and the subpixel grid search optimization approach and the Nelder-Mead simplex method.

A set composed of 100 affine transformations was generated assigning random values for the scale s, 2D rotation ϕ, and for the translation vector (d_x,d_y). These values were sampled uniformly in the following way: s in the interval $(2^{-0.5}, 2^{0.5})$, ϕ in the interval $(-30°, 30°)$, d_x and d_y both in the interval $(-32, 32)$ (pixels). A 260 × 260 T1 MRI image (Fig. 1(a)) was used for registration. Reference images were created applying the 100 affine transformations described above to the original image and then computing the estimated transformation through



(a)

(b)

Fig. 1. (a) T1 MRI reference image used for registration tests; a test case is generated by applying a random transformation to this image. (b) TRME error for each test case and each registration method.

the approach presented in this paper. Fig. 1(b) shows the relative errors for all the transformations estimated through the (a) POC with integer accuracy, (b) subpixel grid search, and (c) Nelder-Mead subpixel optimization as they were described in Section 2. Errors between the grid search and the Nelder-Mead optimization are very similar and, both of them are smaller than the original POC approach with integer accuracy.

We consider a *successful estimation* if the corresponding relative error was at most 0.1, and we compute the average and the median of all the successful estimations for the three approaches, and the computation time for each one (see Table 1). Success mean errors and success median errors are almost the same for the grid search and Nelder-Mead approaches and they are much smaller than the corresponding values with integer accuracy. The success rate error was also improved from 90% at the integer accuracy approach to 100% in the other two approaches. Neverheless, the main difference between the Grid Search and Nelder-Mead is the overhead computation time: Nelder-Mead required in average 10.6ms (in addition to the basic POC search method) to get a solution similar to the one recovered by Grid Search in 184.4ms in this experiment. For this reason, we have conducted the rest of our experiments using only the Nelder-Mead optimization.

3.2 Robustness against Transformation Complexity

Nine sets composed of 100 affine transformations were generated assigning random values for the transformation parameters. These values were sampled uniformly in the following way: s in the interval $(1-0.1a, 1+0.1a)$, ϕ in the interval $(-10a, 10a)$ (degrees), d_x and d_y both in the interval $(-10a, 10a)$ (pixels), where

Table 1. Results obtained after 100 synthetic registration cases using the three methods under discussion

Search method	Success rate	Success mean error	Success median error	Average time (ms)
Integer accuracy	90%	2.74947%	1.93281%	125.5
Grid Search	100%	0.700544%	0.287313%	309.9
Nelder-Mead	100%	0.700457%	0.292196%	136.1

$a = 1, \cdots, 9$ (each value of a corresponds to one of the nine sets). The a-th set is called the tranformation complexity of level a. Transformations with higher complexity are more difficult to be successfully estimated because of possibly small scale factors, or because parts of the transformed image may be cropped when they lie outside of the image frame, producing artificial borders and loss of data.

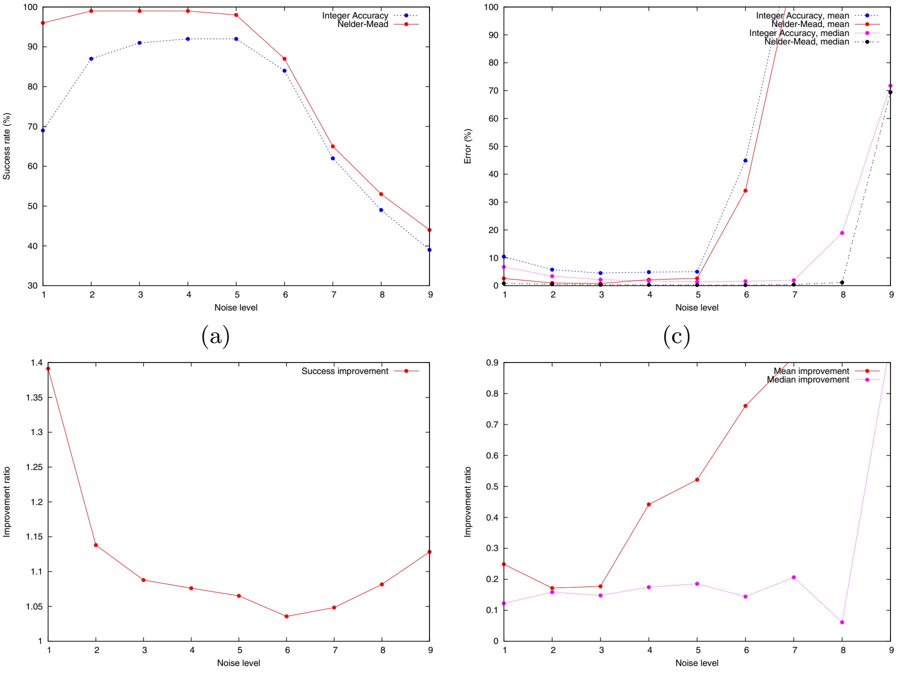
Fig. 2(a) shows the graph of success rate with respect to the complexity of the affine transformations. It is observed that subpixel approach based on Nelder-Mead optimization always improve the success rate with integer accuracy, and this rate suddenly decreases after level 6. Fig. 2(b) shows that success improvement is better for low complexity transformations. Due to the fact that we use a relative measurement of the error, it seems logical that the relative error seems large for low complexity transformations although the absolute error is still proportional to the transformation parameters.

In Fig. 2(c), the median and average errors are shown for both approaches: integer accuracy and Nelder-Mead. In both approaches, average error is under the threshold of 0.1 relative error and it increases for complexity level 6 and larger, and Nelder-Mead always improve the error with respect to the integer accuracy POC. Moreover, median error remains under the threshold of 0.05 of relative error up to level 7, i.e. more than 50% of the tests are under this threshold although the average error increases significatively (some estimations are very far from the expected values).

Fig. 2(d) shows that the improvement on the True Relative Mean Error (TRME) is better for complexity levels 1 to 3, although the improvement is still noticeable up to level 6. For this experiment, the improvement ratio on the best subpixel TRME is approximately 0.2 times the TRME obtained with integer accuracy, while the improvement ratio on the median of the error is almost constant for levels 1 to 8, being 0.15 times the median obtained with integer accuracy.

3.3 Robustness to Noise and Missing Data

In order to test the robustness of the method against noise, we consider additive noise with Gaussian distribution, which was added to the transformed image. Then, we vary the standard deviation from 0 to 0.3 in a normalized scale (i.e., the gray levels go from 0 to 1). The true transformation was fixed with the



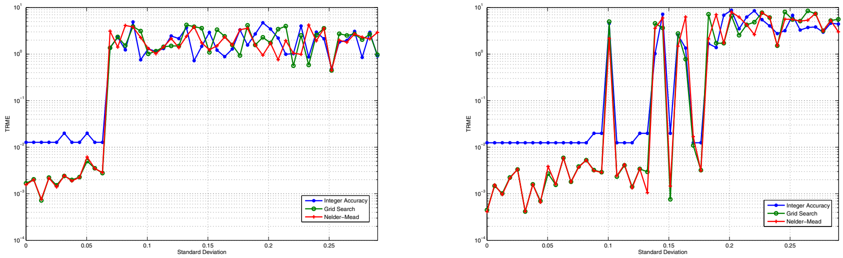
(a) (c)

(b) (d)

Fig. 2. Results obtained with transformations of varying complexity: (a) success rate, (b) success improvement ratio, (c) mean and median error, (d) mean and median error improvement ratio. See text for details.

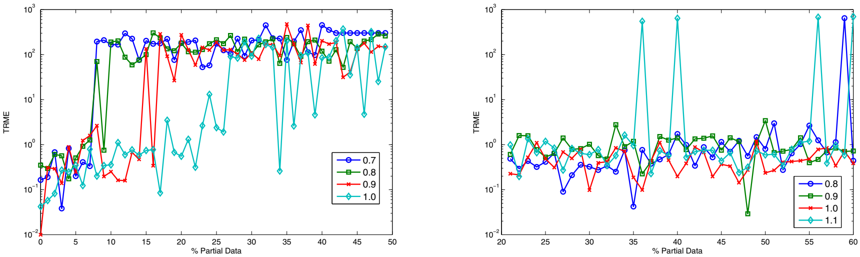
following values: horizontal translation of 10.35 pixels, vertical translation of -20.78 pixels, rotation angle of 30.42 degrees and scale of 1.21. The transformation was estimated using the three algorithms under study. The results are shown in Figure 3 for both the T1 MRI image (left) and the Lena image (right). The results for Lena show that the correct translation can be found well under a standard deviation of 0.1, while the brain image can only achieve a correct transformation under a standard deviation of 0.07.

In another experiment, we tested the robustness of the method with respect to partial/missing data. For the first experiment, we used the Lena image, and replaced the pixel values within a circumference centered at the center of the transformed image with zeros, therefore reducing the amount of useful information. Figure 4(a) shows a plot of the log-TRME versus the percentage of deleted data (with respect to the total number of pixels) for four rigid transformations composed by $\phi = 30.42°$, $d_x = 10.35$, $d_y = -20.78$, and scaling factors of 0.8, 0.9, 1.0, and 1.1, respectively. We used different scales since a little variation in their values may affect significantly the amount of matchable pixel information. One can observe in this plot that in the case of scales less than 1.0, the TRME-values remain close to 0.01 even for images with 50% of missing data.



(a) (b)

Fig. 3. Plots of the TRME with respect to the standard deviation of Gaussian noise added to the input images: (a) results with the T1 MRI image, (b) results with the Lena image



(a) (b)

Fig. 4. Plots of the TRME with respect to the percentage of missing data. (a) results with the T1 MRI image, (b) results with the Lena image. Each trace corresponds to a different value of the scaling parameter (the rotation and translation parameters were fixed).

The experiment was repeated with the T1 MRI image shown in Fig. 1(a), and with scaling factors of 0.7, 0.8, 0.9, and 1.0. These results are shown in Figure 4(b). In this case, the matchable data does not cover the full image frame; therefore, the deletion of data from the center outwards has a bigger impact on the registration performance, especially when the scaling factor is small.

3.4 Results with Real Images

We also applied the proposed method to real image pairs where the true transformation is unknown. Figure 5 shows the result obtained from registering two 512 × 512 aerial images. Once the transformation was found, it was possible to build a larger map. Note that, in this case, the overlapping area between both images is relatively small (about one third of the image size), but the proposed method is still able to solve the problem accurately.

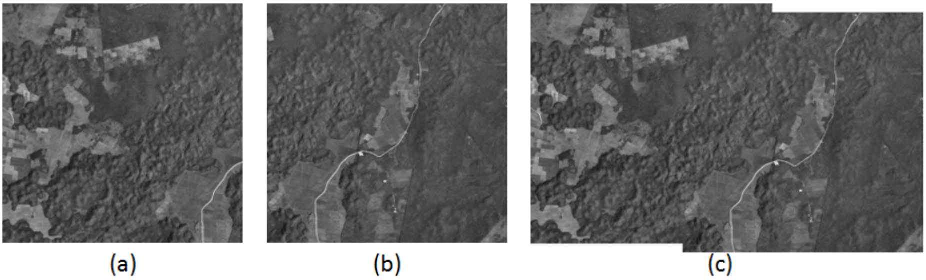


Fig. 5. Example of rigid registration of real images, where the true transformation is unknown. In this example, two aerial images are registered using the proposed method; once the transformation is found, the images can be stitched to form a larger map.

4 Conclusions

A novel approach to estimate the maxima of the phase-only correlation (POC) function was presented in this paper. This approach is based on minimizing the magnitude of the gradient of the POC, which can be done using heuristic techniques that rely only on the evaluation of the function to be minimized. The proposed method was applied to the rigid registration of two images, where it was evaluated in terms of precision and robustness to noise and missing data using synthetic examples. The results of these evaluations are favorable and demonstrate a significant improvement with respect to the classical phase correlation method where integer-valued translations are estimated. The registration algorithm was also applied to real image pairs where the true transformation is unknown, obtaining satisfactory results. A quantitative comparison between the proposed method and other state of the art methods for accurate estimation of the POC maxima is currently being performed. We are currently performing a comparative study between the proposed method and the most relevant techniques in the literature for the estimation of POC maxima with subpixel accuracy. Future work will focus on increasing the success rate of these methods for the rigid registration problem by analyzing multiple maxima of the phase correlation function.

References

1. Kuglin, C.D., Hines, D.C.: The Phase Correlation Image Alignment Method. In: Proc. of the IEEE Int. Conf. on Cybernetics and Society, pp. 163–165 (1975)
2. De Castro, E., Morandi, C.: Registration of Translated and Rotated Images Using Finite Fourier Transforms. IEEE Transactions on Pattern Analysis and Machine Intelligence 9, 700–703 (1987)
3. Reddy, B.S., Chatterji, B.N.: An FFT-Based Technique for Translation, Rotation, and Scale-Invariant Image Registration. IEEE Transactions on Image Processing 5, 1266–1271 (1996)

4. Keller, Y., Averbuch, A., Moshe, I.: Pseudopolar-based estimation of large translations, rotations, and scalings in images. IEEE Transactions on Image Processing 14, 12–22 (2005)
5. Keller, Y., Shkolnisky, Y., Averbuch, A.: The angular difference function and its application to image registration. IEEE Transactions on Pattern Analysis and Machine Intelligence 27, 969–976 (2005)
6. Huang, J.Z., Tan, T.N., Ma, L., Wang, Y.H.: Phase correlation based iris image registration model. J. Comput. Sci. & Technol. 20, 419–425 (2005)
7. Ito, K., Morita, A., Aoki, T., Higuchi, T., Nakajima, H., Kobayashi, K.: A fingerprint recognition algorithm using phase-based image matching for low-quality fingerprints. In: IEEE International Conference on Image Processing (ICIP 2005), vol. 2, pp. 33–36 (2005)
8. Kolar, R., Sikula, V., Base, M.: Retinal image registration using phase correlation. In: Analysis of Biomedical Signals and Images (Proceedings of the 20th International Eurasip Conference), vol. 20, pp. 244–252 (2010)
9. Muquit, M.A., Shibahara, T., Aoki, T.: A High-Accuracy Passive 3D Measurement System Using Phase-Based Image Matching. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. E89-A, 686–697 (2006)
10. Alba, A., Arce-Santana, E., Aguilar Ponce, R.M., Campos-Delgado, D.U.: Phase-correlation guided area matching for realtime vision and video encoding. Journal of Real-Time Image Processing (in press, 2012)
11. Takita, K., Muquit, M.A., Aoki, T., Higuchi, T.: A Sub-Pixel Correspondence Search Technique for Computer Vision Applications. IECIE Trans. Fundamentals E87-A, 1913–1923 (2004)
12. Chien, L.H., Aoki, T.: Robust Motion Estimation for Video Sequences Based on Phase-Only Correlation. In: 6th IASTED International Conference on Signal and Image Processing, pp. 441–446 (2004)
13. Abdou, I.E.: Practical approach to the registration of multiple frames of video images. In: Proc. SPIE, Visual Communications and Image Processing 1999, vol. 3653, pp. 371–382 (1998)
14. Foroosh, H., Zerubia, J.B., Berthod, M.: Extension of phase correlation to subpixel registration. IEEE Transactions on Image Processing 11, 188–200 (2002)
15. Takita, K., Aoki, T., Sasaki, Y., Higuchi, T., Kobayashi, K.: High-accuracy subpixel image registration based on phase-only correlation. IEICE Trans. Fundamentals E86-A, 1925–1934 (2003)
16. Nelder, J.A., Mead, R.: A simplex method for function minimization. Computer Journal 7, 308–313 (1965)
17. Hoge, W.S.: A subspace identification extension to the phase correlation method. IEEE Transactions on Medical Imaging 22, 277–280 (2003)
18. Arce-Santana, E., Alba, A.: Image Registration Using Markov Random Coefficient Fields. In: Brimkov, V.E., Barneva, R.P., Hauptman, H.A. (eds.) IWCIA 2008. LNCS, vol. 4958, pp. 306–317. Springer, Heidelberg (2008)

A Novel Encryption Method with Associative Approach for Gray-Scale Images

Federico Felipe[1], Ángel Martínez[2], Elena Acevedo[1,*], and Marco Antonio Acevedo[1]

[1] Escuela Superior de Ingeniería Mecánica y Eléctrica del Instituto Politécnico Nacional,
Av. IPN s/n Edif Z-4, 3er piso, Telecomunicaciones,
Col. Lindavista, C.P. 07738, Mexico City, Mexico
[2] Alumno Becario de PIFI ESIME-Zacatenco
josekun13@gmail.com, {ffelipe,eacevedo,macevedo}@ipn.mx

Abstract. This paper introduces a novel method for encrypting images with an associative approach. We use a symmetric cryptosystem. The encryption method divides the gray-scale image in *n* squares with which a Morphological Associative Memory is built. Then, the information from the image is stored in the array which represents the memory. The meaning of the array is very difficult to decipher without the private key and the previous knowledge for operating the associative memory. The original and the recovered images were correlated, and the recovering is 100% in all cases.

Keywords: Artificial Intelligence, Image Encryption, Associative approach, Morphological associative memories.

1 Introduction

Encryption is a common technique to uphold image security. Image and video encryption have applications in various fields including internet communication, multimedia systems, medical imaging, Tele-medicine and military communication.

Traditional image encryption algorithms are private key encryption standards (DES and AES), public key standards such as Rivest Shamir Adleman (RSA), and the family of elliptic-curve-based encryption (ECC), as well as the international data encryption algorithm (IDEA).

Current encryption algorithms can be classified into different techniques such as optical [1-6], value transformation [7-11], pixels position permutation [12-14] and chaos-based [15-19].

In this paper, we propose a novel encryption method based on Morphological Associative Memories (MAM). The idea was taken from [20] where the image is divided in blocks and then, they are scrambled to form a new image. The authors applied two methods to reconstruct the original image. This problem is processed as a puzzle issue.

[*] Corresponding author.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 183–193, 2013.

In our method, we also divide a gray-scale image in blocks but we do not scramble them, instead, the blocks are vectorized. These set of vectors correspond to the output patterns of a MAM. The private keys are the input patterns. We build two MAMs: *max* and *min*, which are the encrypted images. In fact, *max* memory can be obtained from the *min* memory and vice versa. This way of encrypting has an advantage: the encrypted image does not have the same dimensions of the original image, therefore not only the data in the encrypted image is different from the original but also the dimensions are not the same. Then it is twice difficult to understand the information from the encrypted image.

Section 2 introduces the basic concepts of Associative Memories together with the description of the operation of the Morphological model. In Section 3, our proposal for encrypting images is described. Experiments and results are presented in Section 4. Finally, the results are analyzed and discussed in the Conclusions section.

2 Associative Memories

This part will discuss the basic concepts of Associative Memories and the operation of the Morphological Associative Memories.

An associative memory is a system that relates input patterns with output patterns. The purpose of an associative memory is to retrieve the corresponding output pattern when the input pattern is presented.

The design of an associative memory requires two phases: learning and recalling phases. In the first phase, the memory associates every input pattern with its corresponding output pattern. The set of associated patterns is called fundamental set. In the recalling phase, the input pattern is presented to the associative memory and the response should be its corresponding output pattern.

Associative memories are a class of artificial intelligence, the interesting thing about them is the fact that can recall an output pattern even if one of the input patterns is not identical to that used in the association process, i.e., the pattern presented to the memory can be a noisy version of a pattern in the fundamental set.

Most associative memories work only with binary numbers, which is a big disadvantage when handling color or gray-level images. For example, if we have a gray-scale image of 50 x 50, a memory that uses real numbers would deal with vectors of 2500 elements but with a binary memory, first each pixel should be converted to 8 bits therefore the dimension of the matrix would be of 400 x 400 and we would have vectors with dimension of 160000. Due to the amount of information becomes very large, we decided to use an associative memory called Morphological Memory which allows working with real numbers, therefore the handled amount of data is reduced and the process is faster.

2.1 Morphological Associative Memories

The fundamental difference between classic associative memories (*Lernmatrix* [21], *Correlograph* [22], *Linear Associator* [23] and Hopfield [24]) and Morphological associative memories [25] lies in the operational bases of the latter, which are the morphological operations: dilation and erosion. This model broke out of the traditional mould of classic memories which use conventional operations for vectors and matrices in learning phase and sum of multiplications for recovering patterns. Morphological associative memories change products to sums and sums to maximum or minimum in both phases.

The basic computations occurring in the proposed morphological network are based on the algebraic lattice structure (**R**, $\vee$, $\wedge$, +), where the symbols $\vee$ and $\wedge$ denote the binary operations of maximum and minimum, respectively. Using the lattice structure (**R**, $\vee$, $\wedge$, +), for an *m* x *n* matrix *A* and a *p* x *n* matrix *B* with entries from **R**, the matrix product $C = A \nabla B$, also called the *max* product of *A* and *B*, is defined by equation (1).

$$c_{ij} = \bigvee_{k=1}^{p} a_{ik} + b_{kj} = (a_{i1} + b_{1j}) \vee (a_{i2} + b_{2j}) \vee \cdots \vee (a_{ip} + b_{pj}) \tag{1}$$

The *min product* of A and B induced by the lattice structure is defined in a similar fashion. Specifically, the *i,j*th entry of $C = A \Delta B$ is given by equation (2).

$$c_{ij} = \bigwedge_{k=1}^{p} a_{ik} + b_{kj} = (a_{i1} + b_{1j}) \wedge (a_{i2} + b_{2j}) \wedge \cdots \wedge (a_{ip} + b_{pj}) \tag{2}$$

Suppose we are given a vector pair **x**= $(x_1, x_2, \ldots, x_n)^t$ and **y**= $(y_1, y_2, \ldots, y_n)^t \in$ $\mathbf{R}^m$. An associative morphological memory that will recall the vector when presented the vector is showed in equation (3)

$$W = y \nabla (-x)^{\mathrm{t}} = \begin{bmatrix} y_1 - x_1 & \cdots & y_1 - x_n \\ \vdots & \ddots & \vdots \\ y_m - x_1 & \cdots & y_m - x_n \end{bmatrix} \tag{3}$$

Since *W* satisfies the equation $W \Delta \mathbf{x} = \mathbf{y}$ as can be verified by the simple computation in equation (4)

$$W \nabla x = \begin{bmatrix} \bigvee_{i=1}^{n} (y_1 - x_i + x_i) \\ \vdots \\ \bigvee_{i=1}^{n} (y_m - x_i + x_i) \end{bmatrix} = y \tag{4}$$

Henceforth, let $(\mathbf{x}^1, \mathbf{y}^1)$, $(\mathbf{x}^2, \mathbf{y}^2)$, …, $(\mathbf{x}^p, \mathbf{y}^p)$ be *p* vector pairs with $x^k = (x_1^k, x_2^k, \ldots, x_n^k)^t \in \mathbf{R}^n$ and $y^k = (y_1^k, y_2^k, \ldots, y_m^k)^t \in \mathbf{R}^m$ for $k = 1, 2, \ldots, p$. For a given set of pattern associations $\{(\mathbf{x}^k, \mathbf{y}^k) \mid k = 1, 2, \ldots, p\}$ we define a pair of associated pattern matrices (*X*, *Y*), where $X = (\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^p)$ and $Y = (\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^p)$. Thus, *X* is of dimension *n* x *p* with *i,j*th entry x_i^j and *Y* is of dimension *m* x *p* with *i,j*th entry y_i^j. Since $\mathbf{y}^k \nabla (-\mathbf{x}^k)^t = \mathbf{y}^k \Delta (-\mathbf{x}^k)^t$, the notational burden is reduced by denoting these identical morphological outer vector products by $\mathbf{y}^k \times (-\mathbf{x}^k)^t$. With each pair of matrices (*X*, *Y*) we associate two natural morphological *m* x *n* memories *M* and *W* defined by

$$M = \bigvee_{k=1}^{p} (y^k \times (-x^k)^t) \tag{5}$$

$$W = \bigwedge_{k=1}^{p} (y^k \times (-x^k)^t) \tag{6}$$

With these definitions, we present the algorithms for the learning and recalling phase.

Learning Phase

1. For each p association ($\mathbf{x}^{\mu}$, $\mathbf{y}^{\mu}$), the minimum product is used to build the matrix $\mathbf{y}^{\mu}$ Δ $(-\mathbf{x}^{\mu})^t$ of dimensions m x n, where the input transposed negative pattern $\mathbf{x}^{\mu}$ is defined as $(-\mathbf{x}^{\mu})^t = \left(-x_1^{\mu}, -x_2^{\mu}, \ldots, -x_n^{\mu}\right)$.
2. The maximum and minimum operators ($\vee$ and $\wedge$) are applied to the p matrices to obtain M and W memories as equations (5) and (6) show.

Recalling Phase

In this phase, the minimum and maximum product, Δ and ∇, are applied between memories M or W and input pattern $\mathbf{x}^{\omega}$, where $\omega \in \{1, 2, ..., p\}$, to obtain the column vector $\mathbf{y}$ of dimension m as equations (7) and (8) shows:

$$\mathbf{y} = M \,\Delta\, \mathbf{x}^{\omega} \tag{7}$$

$$\mathbf{y} = W \,\nabla\, \mathbf{x}^{\omega} \tag{8}$$

3 Implementation

Our proposal is applied to gray-scale images, then the image have to be converted to this format. The gray-scale image is partitioned in blocks of n x n pixels. This is illustrated in Figure 1.

For the partition process, we experimentally found that the recalling was more effective when the segments of the image were square; therefore the greatest common divisor between length and width of the image was obtained, this value had to fall into the range of 10 to 100 to avoid storing arrays occupying an excessive memory space. If the greatest common divisor falls outside this range, then it was multiplied (when is minor) or divided (when is greater) by ten until the condition is reached.

Each sub-matrix (segment) is vectorized to build the m output patterns y, where m is the number of blocks (see Figure 2).

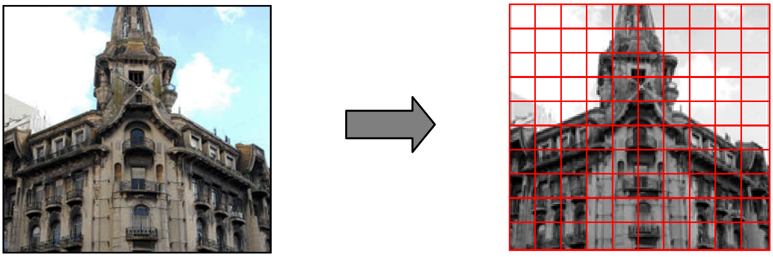


Fig. 1. The image is converted to gray-scale format and then it is partitioned

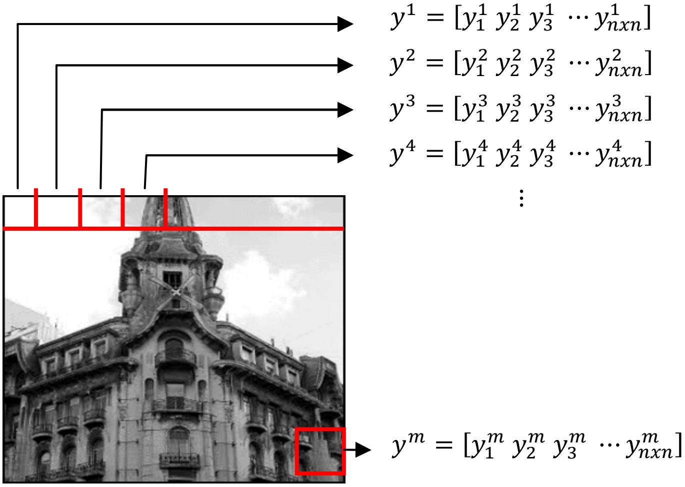

$y^1 = [y_1^1\ y_2^1\ y_3^1\ \cdots y_{nxn}^1]$
$y^2 = [y_1^2\ y_2^2\ y_3^2\ \cdots y_{nxn}^2]$
$y^3 = [y_1^3\ y_2^3\ y_3^3\ \cdots y_{nxn}^3]$
$y^4 = [y_1^4\ y_2^4\ y_3^4\ \cdots y_{nxn}^4]$
$\vdots$
$y^m = [y_1^m\ y_2^m\ y_3^m\ \cdots y_{nxn}^m]$

Fig. 2. Each segment is vectorized and they correspond to each output pattern *y*

Now, the private key is created. It corresponds to the input patterns *x*. the cardinality of each input patter is equal to the number of blocks *m*. These patterns are built as follows:

$$\begin{array}{lcl} x^1 = [a\ 0\ 0\ 0\ 0\ \cdots 0_{(m)}] & \rightarrow & y^1 \\ x^2 = [0\ a\ 0\ 0\ 0\ \cdots 0_{(m)}] & \rightarrow & y^2 \\ x^3 = [0\ 0\ a\ 0\ 0\ \cdots 0_{(m)}] & \rightarrow & y^3 \\ \cdots \quad\quad \cdots & & \cdots \\ x^m = [0\ 0\ 0\ 0\ 0\ \cdots a_{(m)}] & \rightarrow & y^m \end{array} \tag{9}$$

where *a* is a value which was determined experimentally. With these vectors and patterns y, we applied equation (5) to build a *max* Morphological associative memory (*M*).

Also, a *min* Morphological associative memory (*W*) is built with the following patterns x̄, by applying equation (6):

$$\begin{aligned}
\bar{x}^1 &= \left[b\ 0\ 0\ 0\ 0\ \cdots 0_{(m)}\right] && \rightarrow\ y^1 \\
\bar{x}^2 &= \left[0\ b\ 0\ 0\ 0\ \cdots 0_{(m)}\right] && \rightarrow\ y^2 \\
\bar{x}^3 &= \left[0\ 0\ b\ 0\ 0\ \cdots 0_{(m)}\right] && \rightarrow\ y^3 \\
\cdots & \qquad \cdots && \qquad \cdots \\
\bar{x}^m &= \left[0\ 0\ 0\ 0\ 0\ \cdots b_{(m)}\right] && \rightarrow\ y^m
\end{aligned} \tag{10}$$

where b is the symmetric of a, then $b = -a$.

After several tests, we found that the value of a achieving the best recalling was equal to 300. We gave to a values of: 10, 20, 30, 40, 50, 100, 200, 250, 290 but these values were not effective when recovering. We tested values greater than 300 until 1000 and they showed the same result obtained with 300, therefore we decided to use 300.

The encryption process is terminated and matrices M and W represent the encrypted images.

The information necessary to decrypt the image is the number of blocks (m) and the cardinality of patterns y (nxn). Also, we need to know the dimensions of the original image. The size of M or W is of (nxn) x m, therefore the information is implicit in the encrypted image.

Now, for the inverse process, the images are analyzed to obtain the information. Once we know the number of blocks, we built the input patterns (private key) both x and $\bar{x}$. Associative memories M and W are operated with x and $\bar{x}$ patterns using equations (7) and (8), respectively. For each pattern y^m, the *m-th* block is rebuilt and with the knowledge of the dimensions of the original image the blocks are accommodated in the right place.

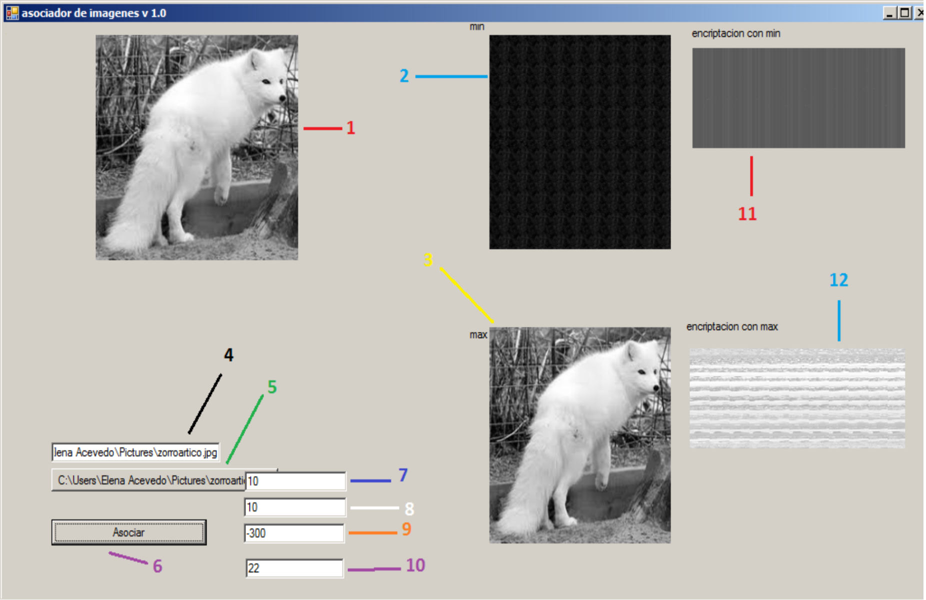
The original and the recovered images are correlated. Correlation [25] is a measure we use to tell how much two waveforms or two images are like each other. With this measure we can know if the recovered image is the same that the original image or if there are differences between them. We use the Matlab function called ***corr2*** to accomplish this task.

The following section shows the results when applying the described process.

4 Experiments and Results

The algorithm was implemented with the programming language Microsoft Visual C++ 2010 Express Edition® and was tested on a Laptop Toshiba® with Intel-ATOM® processor and 2 GB of RAM memory, the operating system was Microsoft Windows 7 Ultimate®.

We created a database containing five sets of ten images grouped by theme (animals, buildings, videogames, trees and paintings). All images were partitioned in 100 segments. Our proposal was applied to this database. Figure 3 shows an example of the software.



lena Acevedo\Pictures\zorroartico.jpg

C:\Users\Elena Acevedo\Pictures\zorroarti

10

10

-300

22

Fig. 3. Example screen from the software of our proposal

The meaning of every element in Figure 3 is described as follows:

1- Gray-scale original image
2- Image rebuild with *min* memory
3- Image rebuild with *max* memory
4- Image path
5- Selection of the image
6- Execution of the program
7- Number of segments for row
8- Number of segments for column
9- Value of *a*
10- Pixel per side of a segment
11- Encrypted image by *min* memory
12- Encrypted image by *max* memory

Data in points 7-10 is given by the user.

Figure 4 shows five images from each theme. We can observe that the first column shows the original image, in the second column the encrypted image is showed and the rebuild image is presented in column 3. It is important to notice that the encrypted image do not have the same dimensions that the original image thereby, it is more difficult to try to decipher the information in the encrypted image.

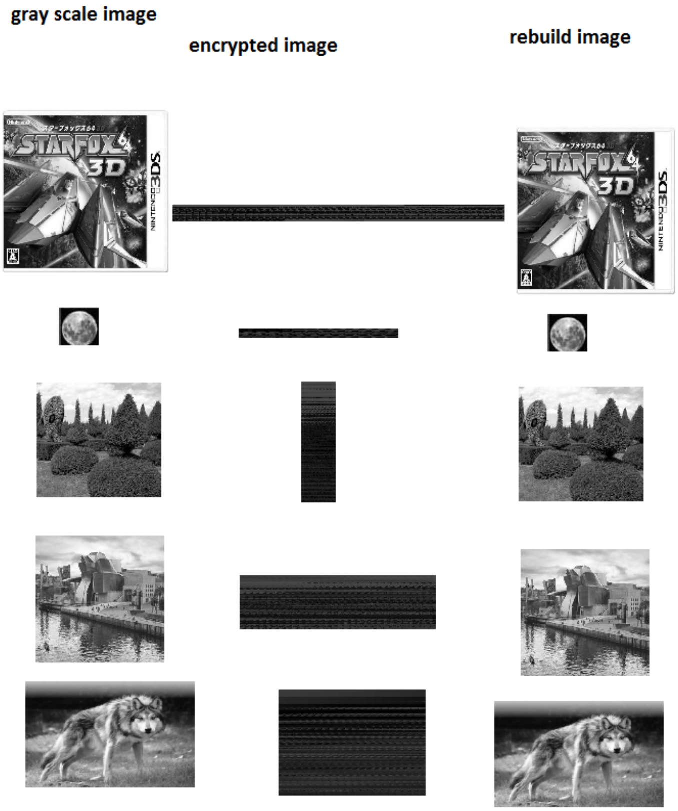


Fig. 4. Original, encrypted and rebuild image are shown in columns 1, 2 and 3, respectively

Table 1 shows the results when applying our proposal. The images were divided into 100 blocks. Column 1 indicates the number of the image. In the second column, we first used the value of -300 for both *a* and *b*, then we used 300 for these two variables. The number of pixels of each side of the block is showed in the third column.

From Table 1, one can observe that when the value of 300 is applied the correct recovering is achieved by the *min* memory and if the value of -300 is used then the best recovering is achieved by the *max* memory. Therefore, if we use *a* = -300 and *b* = -*a* for rebuilding the images we will always have a perfect recovering with both memories. Then it is just necessary to send one of the memories then depending of the applied memory will be the key (or patterns from equations (9) and (10)) we will use to decrypted the image.

Table 1. Results of the decrypted process using both morphological memories: *min* and *max*

Image	Value of *a* and *b*	Pixels per side of the block	Recovering MIN (%)	Recovering MAX (%)
1	300	30	100	0
	-300		5.5	100
2	300	30	100	6.62
	-300		5.72	100
3	300	30	100	1.75
	-300		2.47	100
4	300	30	100	7.12
	-300		0	100
5	300	30	100	7.72
	-300		0.9	100
6	300	30	100	4
	-300		4.2	100
7	300	32	100	10.04
	-300		6.67	100
8	300	30	100	2.86
	-300		3.63	100
9	300	22	100	0
	-300		2.45	100
10	300	22	100	6.11
	-300		0.39	100

5 Conclusions

The encryption model proposed in this work is novel because it uses an associative approach which it has not been applied in any other work.

Min and *max* Morphological Associative Memories represents the encrypted data.

The method has a great advantage: the encrypted image does not have the same dimensions that the original image.

Although the key is very simple to build, the latter advantage allows the method to be difficult to decipher.

The original image is always recovered when the value of 300 is used to build the private key.

The obtained results show that the associative approach is very effective when it is applied to the Encryption area.

Acknowledgments. The authors would like to thank the Instituto Politécnico Nacional (COFAA, EDI and SIP), and SNI for their economical support to develop this work.

References

1. Tao, R., Xin, Y., Wang, Y.: Double image encryption based on random phase encoding in fractional Fourier domain. Opt. Express 15, 16067–16079 (2007)
2. Ge, F., Chen, L., Zhao, D.: A half-blind image hiding and encryption method in fractional Fourier domains. Opt. Commun. 281, 4254–4260 (2008)
3. Liu, Z., Li, Q., Dai, J., Sun, A., Liu, S., Ahmad, M.: A new kind of double encryption by using a cutting spectrum in the 1-D fractional Fourier transform domains. Opt. Commun. 282, 1536–1540 (2009)
4. Wang, B., Zhang, Y.: Double images hiding based on optical interference. Opt. Commun. 282, 3439–3443 (2009)
5. Meng, X., Cai, L., Wang, Y., Yang, X., Xu, X., Dong, G., et al.: Digital image synthesis and multiple-image encryption based on parameter multiplexing and phase-shifting interferometry. Opt. Lasers Eng. 47, 96–102 (2009)
6. Weng, D., Zhu, N., Wang, Y., Xie, J., Liu, J.: Experimental verification of optical image encryption based on interference. Optics Communications 284, 2485–2487 (2011)
7. Chen, R.J., Lai, J.L.: Image security system using recursive cellular automata substitution. Pattern Recognition 40, 1621–1631 (2007)
8. Guo, Q., Liu, Z., Liu, S.: Color image encryption by using Arnold and discrete fractional random transforms in HIS space. Optics and Lasers in Engineering 48(12), 1174–1181 (2010)
9. Tao, R., Meng, X.Y., Wang, Y.: Image encryption with multiorders of fractional fourier transforms. IEEE Transactions on Information Forensics and Security 5(4), 734–738 (2010)
10. Liu, Z., Xu, L., Lin, C., Dai, J., Liu, S.: Image encryption scheme by using iterative random phase encoding in gyrator transform domains. Optics and Lasers in Engineering 49(4), 542–546 (2011)
11. Liu, Z., Gong, M., Dou, Y., et al.: Double image encryption by using Arnold transform and discrete fractional angular transform. Optics and Lasers in Engineering 50, 248–255 (2012)
12. Nien, H.H., Huang, W.T., Hung, C.M., et al.: Hybrid image encryption using multi-chaos-system. In: 7th International Conference on Information, Communications and Signal Processing, ICICS 2009, December 8-10, pp. 1–5 (2009)
13. Prasad, M., Sudha, K.L.: Chaos Image Encryption using Pixel shuffling. In: Computer Science & Information Technology (CS & IT), CS & IT 02, pp. 169–179 (2011)
14. Zhou, X., Ma, J., Du, W., Zhao, Y.: Ergodic Matrix and Hybrid-key Based Image Cryptosystem. International Journal of Image, Graphics and Signal Processing 4, 1–9 (2011)
15. Chen, L.: A Novel Image Encryption Scheme Based on Hyperchaotic Sequences. Journal of Computational Information Systems 8(10), 4159–4167 (2012)
16. Wang, X.Y., Yang, L., Liu, R., Kadir, A.: A chaotic image encryption algorithm based on perceptron model. Nonlinear Dynamics 62, 615–621 (2010)
17. Sankaran, S.K., Santhosh Krishna, B.V.: A New Chaotic Algorithm for Image Encryption and Decryption of Digital Color Images. International Journal of Information and Education Technology 1(2), 137–141 (2011)
18. Fu, C., Chen, J.J., Zou, H., Meng, W.H., et al.: A chaos-based digital image encryption scheme with an improved diffusion strategy. Optics Express 20(3), 2363–2378 (2012)
19. Seyedzadeh, S.M., Mirzakuchaki, S.: A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map. Signal Processing 92(5), 1202–1215 (2012)

20. Zhao, Y.X., Su, M.C., Chou, Z.L., Lee, J.: Puzzle Solver and Its Application in Speech Descrambling. In: Proceedings of the 2007 WSEAS International Conference on Computer Engineering and Applications, pp. 171–176 (2007)
21. Steinbuch, K.: Die Lernmatrix. Kybernetik 1(1), 36–45 (1961)
22. Willshaw, D., Buneman, O., Longuet-Higgins, H.: Non-holographic associative memory. Nature 222, 960–962 (1969)
23. Anderson, J.A.: A simple neural network generating an interactive memory. Mathematical Biosciences 14, 197–220 (1972)
24. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences 79, 2554–2558 (1982)
25. Ritter, G.X., Sussner, P., Diaz de León, J.L.: Morphological Associative Memories. IEEE Transactions on Neural Networks 9, 281–293 (1998)
26. Stearns, S.D.: Digital Signal Processing with examples in Matlab, pp. 40–42. CRC Press (2003)

FPGA-Based Architecture for Extended Associative Memories and Its Application in Image Recognition

Guzmán-Ramírez Enrique[1], Arroyo-Fernández Ignacio[1], González-Rojas Carlos[1],
Linares-Flores Jesús[1], and Pogrebnyak Oleksiy[2]

[1] Universidad Tecnológica de la Mixteca, Carretera Acatlima km 2.5, CP 6900
Huajuapan de León, Oaxaca, México
eguzman@mixteo.utm.mx, ignacio820209@yahoo.com.mx,
gorc851031@gmail.com
[2] Centro de Investigación en Computación, Ave. Juan de Dios Batis, Esq. Miguel Othón de
Mendizábal, Nueva industrila Vallejo, CP 07738, México D.F.
olek@cic.ipn.mx

Abstract. In this paper, an efficient FPGA-based architecture for Extended Associative Memories (EAM) focused on the classification stage of an image recognition system for real-time applications is presented. The EAM training phase is only used during the generation of associative memory, completed this task, this module is disconnected from the system; for this reason the hardware architecture of this module was designed for optimize the FPGA resource usage. On the other hand, the EAM can be part of a system requiring working in real time, such a perception system for a mobile robot or a personal identification system; for this reason, the hardware architecture of EAM classification phase was designed for obtaining high processing speeds. Experimental results show high performance of our proposal when altered versions of the images used to train the memory are presented.

Keywords: Image recognition, extended associative memories, reconfigurable logic, hardware architecture.

1 Introduction

Pattern recognition aims to classify data (patterns) based on either a priori knowledge or on statistical information extracted from the patterns. The patterns to be classified are usually groups of measurements or observations, defining points in an appropriate multidimensional space.

A generic pattern recognition system consists of three main stages: pre-processing, feature extraction and classification. This paper is focused on this stage, so it is appropriate to mention paradigms used in the development of classification algorithms, some of them are: artificial neural networks [1], [2], principal component analysis [3], [4], fuzzy models [5], [6], genetic algorithms [7], [8], associative memories [9], [10].

A specific and typical application for a pattern recognition system is the recognition and classification of images. The systems for image recognition and

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 194–204, 2013.

classification have diverse applications, e.g. autonomous robot navigation [11], image tracking, radar [12], face recognition [13], biometrics [14], intelligent transportation.

In this paper we concentrate on the classification stage of an image recognition system for real-time applications; depending on the nature of training and testing database and speed requirements, the classification stage could be very challenging. For these reasons, we propose the FPGA-based implementation of a classification stage using extended associative memories (EAM).

Many different hardware implementation of image recognition for real-time application based on associative memories have been discussed in literature, we mention some of them.

A high performance parallel version of Advanced Distributed Associative Memory (ADAM), implemented on Cellular Neural Network Associative Processor (C-NNAP), is described in [15]. The C-NNAP is a system composed of DSP-32C and Sum and Threshold (SAT) processor. The SAT processor performs the most demanding computational tasks, so it is implemented using an Actel A1280XL FPGA device. The results show that proposed system performance is improved when using the SAT processor instead of a coprocessor.

Heittmann, A. and Ckert, U. proposed a mixed mode digital/analog special purpose VLSI hardware implementation of an associative memory with neural architecture called BiNAM (Binary Neural Associative Memory) [16]. The BiNAM has a simple matrix structure with binary elements (synapses, connection weights) and performs a pattern mapping or completion of binary input/output patterns. The authors present a new circuit concept for BiNAM implementation. In this concept the weight storage cell and the analog summing circuit are joined together in a standard SRAM cell to minimize the additional logic needed. The results reported include the fabrication of a BiNAM-chip with a satisfactory performance.

In [17] authors present the implementation of Alpha-Beta associative memories on reconfigurable logic (Xilinx Spartan3 FPGA device) using parallel schemes of operational units; as a result, a simple but efficient embedded processing architecture that overcomes various challenges involved in pattern recognition tasks is obtained. The proposed architecture for Alpha-Beta associative memories is applied for automatic fingerprint verification task; the performance of this architecture is measured by learning large sequences of symbols and recalling them successfully. This proposal has the limitation of only working with patterns of binary data in its components.

The remaining sections of this paper are organized as follows. In next Section, a brief theoretical background of EAM is given. In Section 3 we describe the proposed FPGA-based architecture for EAM. Then, in Section 4 we present the FPGA implementation results for the proposed architecture and its application on gray-scale image recognition. Finally, Section 5 contains the conclusions of this paper.

2 Extended Associative Memories

The EAM was proposed by H. Sossa *et al.* as an associative memory model for the classification of real-valued patterns [18]. The EAM is an extension of the Lernmatrix model proposed by K. Steinbuch [19]. The EAM is based on the general concept of the learning function of associative memory and presents a high performance in pattern classification of real value data in its components and with altered pattern version [20]. The EAM is an associative memory designed for pattern classification which fundamental purpose is to establish a relation of an input pattern $\mathbf{x} = [x_i]_n$ with an index *i* of a class c_i.

2.1 Training Phase of the EAM

The training phase of the EAM consists on evaluating a function ϕ for each class. Then, the matrix **M** can be structured as:

$$\mathbf{M} = \begin{bmatrix} \mathbf{m}_1 \\ \vdots \\ \mathbf{m}_N \end{bmatrix} = \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_N \end{bmatrix} \tag{1}$$

where ϕ_i is the evaluation of ϕ for all patterns of class *i*, $i = 1,2,...,N$. The function ϕ can be evaluated in various manners. The arithmetical average operator (**prom**) is frequently used in signal treatment. In [18], the authors studied the performance of this operator to evaluate the function ϕ.

The goal of the training phase is to establish a relation between an input pattern $\mathbf{x} = [x_i]_n$, and the index *i* of a class c_i. Considering that each class is composed for *q* patterns $\mathbf{x} = [x_i]_n$, and that $\phi_i = (\phi_{i,1},...,\phi_{i,n})$. Then, the training phase of the EAM, when the **prom** operator is used to evaluate the function ϕ_i, is defined as:

$$\phi_{i,j} = \frac{1}{q}\sum_{l=1}^{q} x_{j,l}, \quad j = 1,...,n \tag{2}$$

The memory **M** is obtained after evaluating all functions ϕ_i. In the case when *N* classes exist and the vectors to classify are *n*-dimensional, the resultant memory $\mathbf{M} = [m_{ij}]_{N\times n}$ is:

$$\mathbf{M} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \vdots \\ \mathbf{m}_N \end{bmatrix} = \begin{bmatrix} \phi_{1,1} & \phi_{1,2} & \cdots & \phi_{1,n} \\ \phi_{2,1} & \phi_{2,2} & \cdots & \phi_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{N,1} & \phi_{N,2} & \cdots & \phi_{N,n} \end{bmatrix} = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,n} \\ m_{2,1} & m_{2,2} & \cdots & m_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ m_{N,1} & m_{N,2} & \cdots & m_{N,n} \end{bmatrix} \tag{3}$$

In the EAM, being an associative memory utilized for pattern classification, all the synaptic weights that belong to class *i* are accommodated at the *i*-th row of a matrix **M**. The final value of this row is a function ϕ of all the patterns belonging to class *i*. The ϕ function acts as a generalized learning mechanism that reflects the flexibility of the memory [18].

2.2 Classification Phase of the EAM

The goal of the classification phase is the generation of the class index to which an input pattern belongs. The pattern classification by EAM is done when a pattern $\mathbf{x}^{\mu} \in \mathbf{R}^{n}$ is presented to the memory **M** generated at the training phase. The EAM possess the feature of classifying a pattern not necessarily one of those already used to build the memory **M**. When the **prom** operator is used, the class to which **x** belongs is defined as

$$i = \underset{l}{\mathbf{arg}} \left[\bigwedge_{l=1}^{N} \bigvee_{j=1}^{n} \left| m_{lj} - x_{j} \right| \right] \tag{4}$$

In this case, the operators $\vee \equiv$ max and $\wedge \equiv$ min perform morphological operations on the difference of the absolute values of the elements m_{ij} of **M** and the component x_j of the pattern **x** to be classified.

The **Theorem 1** and **Corollaries 1-5** from [18] govern the conditions that must be satisfied to obtain a perfect pattern classification. Either the pattern may be from the fundamental set of couples or be an altered version of the pattern.

3 Proposed FPGA-Based Architecture for EAM

In this section we present the proposed FPGA-based architecture for EAM. Our proposal is modeled considering the Spartan-3 Generation FPGA architecture, using the Xilinx ISE Design Suite 12.2 as the EDA-CAD tool and following a methodology with hierarchical and modular approach described in [21]. Initially, this methodology is approached in formulated an overview of the system, then, in order to simplify the system implementation, it is partitioned into small and reusable units or subsystems. Each subsystem is then refined in yet greater detail, sometimes in many additional subsystem levels, until the entire specification is reduced to base elements.

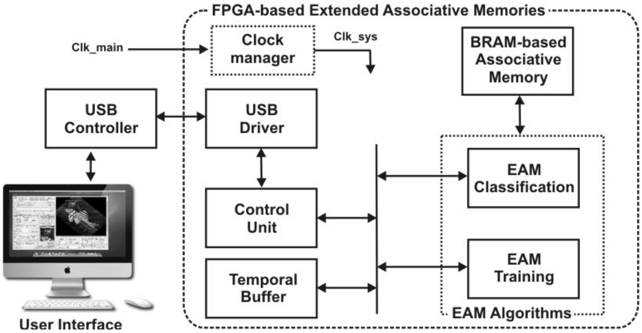
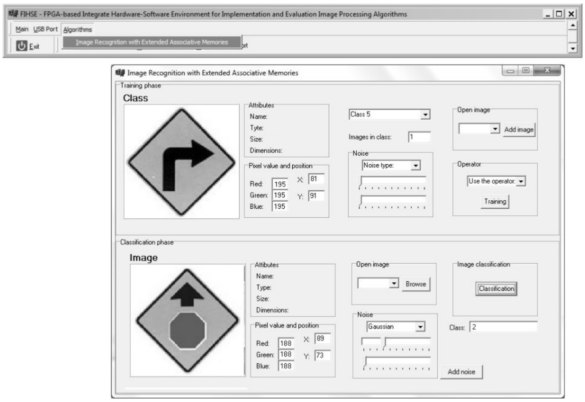


Fig. 1. Block diagram of the FPGA-based architecture for EAM

Now, based on this methodology, an initial modular partitioning step is applied on the **FPGA-based EAM** architecture, this process generate four main components, *Control Unit*, *Temporal Buffer*, *EAM Algorithms* and *BRAM-based Associative Memory* (see Fig. 1). The *Clock Manager*, *USB System* and *User Interface* components meet secondary functions. The components were modeled using the VHDL hardware description language (RTL and behavior level modeling), generated using the Xilinx CORE Generator tool or using pre-optimized elements that the device in use includes.

The *User Interface* component is application software running on the host PC. *User Interface* allows the user to explore, view and send any image to the **FPGA-based EAM**, indicate which algorithms, training or classification, will be applied on the image, and display the image classification results. The *User Interface* GUI is shown in Fig. 2; it was developed based on C++.



Class 5

Images in class: 1

Red: 195 X: 81

Green: 195 Y: 91

Blue: 195

Red: 188 X: 89

Green: 188 Y: 73

Blue: 188

Gaussian

Class: 2

Fig. 2. User *Interface* GUI

The *Clock Manager* component generates the global clock system (`Clk_sys`) for the **FPGA-based EAM** from the principal frequency (`Clk_main`). The principal element of this component is a DCM block. DCM is embedded on the Spartan3E FPGA's families; DCM provides a correction clock feature, ensuring a clean `Clk_sys` output clock with a 50% duty cycle and eliminating clock skew (it guarantees to generate zero-propagation-delay at distribution of the clock signals).

The *USB Controller* consists of a DLP-USB245R device; it is a USB-to-parallel FIFO interface module that utilizes the popular FT245R IC from FTDI. The *USB Controller* provides an effective method of transferring data to/from *User Interface* and **FPGA-based EAM** at up to 8 million bits (1 megabyte) per second via the USB port. The *USB Driver* component describes the interface protocol that allows the **FPGA-based EAM** to establish the transferring data with the *User Interface* through the DLP-USB245R.

The *Control Unit* component provides a link between the *User Interface* and **FPGA-based EAM**; this component carries out the following functions: receives and stores data into *Temporal Buffer*, enables execution of EAM algorithms and transmits the classification process results.

All Spartan-3 Generation FPGAs feature multiple blocks RAM (BRAM) memories, these elements are ideal for applications requiring large on-chip memories. Each BRAM contains 18,432 bits of fast static RAM and it supports multiple configurations or aspect ratios. An output register, which enables full-speed operation at over 250 MHz for all data widths, is included too. The Xilinx Spartan-3E1200 FPGA device has 28 embedded BARMs. The Xilinx CORE Generator tool includes an advanced custom-memory constructor that generates area and performance-optimized memories using BRAM resources in Xilinx FPGAs.

The *Temporal Buffer* is a BRAM-based component and it is used to store the image to be processed. This component consists of two BRAMs organized as 4096-locs×8-bit, allowing to manipulate images up to 4096 pixels at 8-bit (256) grayscale levels. The *Temporal buffer* can be by accessed both the *Unit Control* and *EAM Algorithms* components.

The *BRAM-based Associative Memory* component contains coded information, generated in the training phase, which relates each pattern with the class to which it belongs; further, this information represents the knowledge base that will be used in the classification phase. This component is composed of *N* modules, where *N* is the number of classes to be processed. Each module consists of 2 BRAMs organized as 4096-locs×8-bit (see Fig. 3).

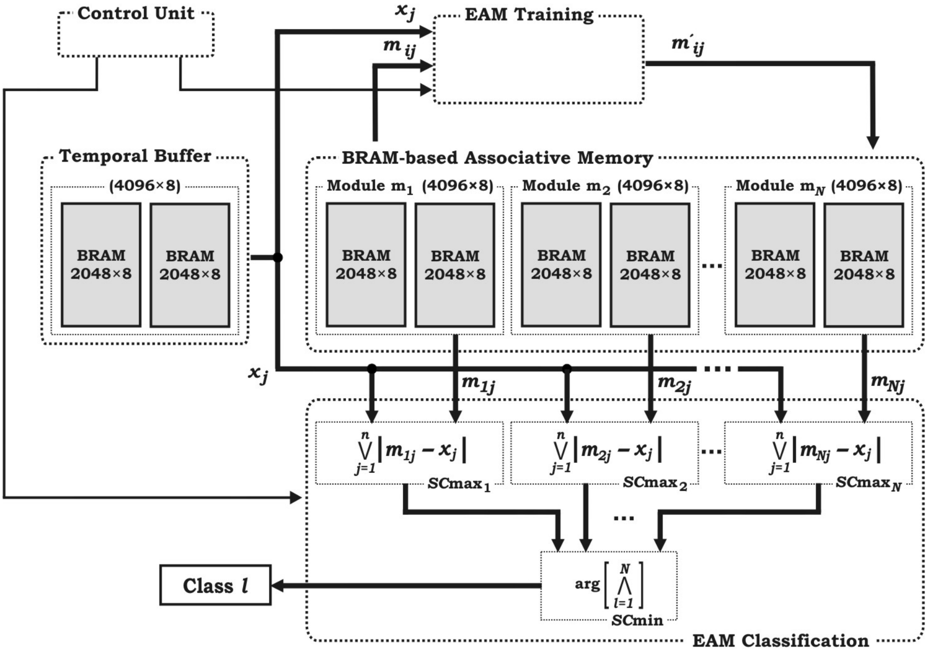


Fig. 3. Proposed FPGA-based architecture for Extended Associative Memories

Each module stores the information corresponding to one class, so the proposed system, with this configuration, is able to learn and encode up to 13 classes. The BRAMs have a synchronous access; any arrangement of modules BRAMs requires only 1 clock cycle, then the access time of both components *Temporal Buffer* and *BRAM-based Associative Memory* is defined as $t_{BRAM} = 10ns$ @ 100 MHz.

The *EAM algorithms* component comprises two modules: *EAM Training* and *EAM Classification*, which describe the training and classification phases of the EAM respectively. Due to the nature and function of each algorithm, the architectures of these modules were designed looking for different objectives.

Usually, the *EAM Training* module is only used during the generation of associative memory, completed this task, this module is disconnected from the system; for this reason the hardware architecture of this module was designed for optimize the FPGA resource usage.

On the other hand, the EAM can be part of a system requiring working in real time, such a perception system for a mobile robot or a personal identification system; for this reason, the hardware architecture of *EAM Classification* module was designed for obtaining high processing speeds.

To achieve this aim, the *BRAM-based Associative Memory* component was structured with *N* modules which can be accessed simultaneously. Thus, the architecture of *EAM Classification* component, shown in Fig. 3, consists of *N* sub-components, named $SC\max_i$ $(i = 1, 2, ..., N)$, which function is to calculate $\overset{n}{\underset{j=1}{\vee}} |m_{lj} - x_j|$. These sub-components are connected to the modules from the *BRAM-based Associative Memory* component and they are working concurrently. The access time of the $SC\mathbf{max}$ sub-component is defined as $t_{SC\mathbf{max}} = 5.345ns$.

The sub-component $SC\mathbf{min}$ complements the proposed architecture for the *EAM Classification* component; the function of this sub-component is to calculate the $\underset{l}{\mathbf{arg}}\left[\overset{N}{\underset{l=1}{\wedge}}\right]$ of all $\overset{n}{\underset{j=1}{\vee}} |m_{lj} - x_j|$ received. In order to obtain a speed optimization and improve the performance of this sub-component, it is composed of *N*-1 8-bit magnitude comparators connected to a cascading network; the stages number of the network is defined as $\lfloor \mathbf{log}_2(N) \rfloor$. The access time of the $SC\mathbf{min}$ sub-component is defined as: $t_{SC\mathbf{min}} = 21.101ns$ (for 13 classes).

Therefore, the time required for the proposed system processes an image is defined as:

$$t_{image} = (n)(t_{BRAM} + t_{SC\mathbf{max}} + t_{SC\mathbf{min}}) \tag{6}$$

where n is the number of image pixels (descriptors). Table 1 shows the estimated time that proposed architecture requires to process typical video image resolutions.

Table 1. Estimated time that proposed architecture requires to process typical video image resolutions

Image resolution	Estimated time to process one image	Images per second.
QCIF (176×120)	769.73 μs	1299.14
CIF (352×240)	3.078 ms	324.78
2CIF (704×240)	6.16 ms	162.5
4CIF (704×480)	12.31 ms	81.19
VGA (640×480)	11.19 ms	89.31

4 Experimental Results

FPGA implementation results for the proposed FPGA-based architecture for EAM and its application on gray-scale image recognition are presented in this section. In this study, FPGA design flow proposed in [22] is used as a design flow reference, and the proposed architecture is implemented on the Xilinx Spartan-3E1200-FG320 FPGA device. Different sets of images have been used to examine proposed architecture performance (recognition rates for distorted versions of the training images), the device resources utilization (slices, BRAMs) and maximum speed (MHz).

Considering the use of image up to 4096 pixel (256 gray-scale levels) and that associative memory has the capacity to encode and store up to 13 classes, a device utilization summary is listed in Table 2.

Table 2. Resources utilization and overall proposed architectures performance

Parameters	Used - Available	Utilization
Slices	809 - 8672	9 %
Slices Flip Flops	761 - 17344	4 %
4-input LUTs	1498 - 17344	8 %
BRAMs	28-28 (13 classes)	100 %
Maximum frequency	103.288 MHz	

For the experiment we chose a test set of images shown in Fig. 4; these images were used to training the EAM. To evaluate the proposed architecture performance, altered versions, with mixed noise, of these images were presented to the classification phase of the EAM (see Fig. 4). The results of this experiment are shown in Table 3.

Experimental results show that our proposal presents an efficiency of 100 % when the original images are presented. The efficiency is maintained when images with a high percentage of erosive noise are presented. When images with 40% of mixed noise are presented, the efficiency of our proposal is decreased. This is mainly due to the implicit properties of the EAM as established in Theorem 1 and Corollaries 1-5 from [18].

The arithmetic units of architecture proposed working with integer values, while the operation of EAM generates real-value results; therefore, in training processes of EAM exists information loss. We decided to work with integer arithmetic units to reduce the complexity of the proposed architecture and obtain high processing speeds. Despite this, and based on the results, we observe an acceptable efficiency in the performance of our proposal.

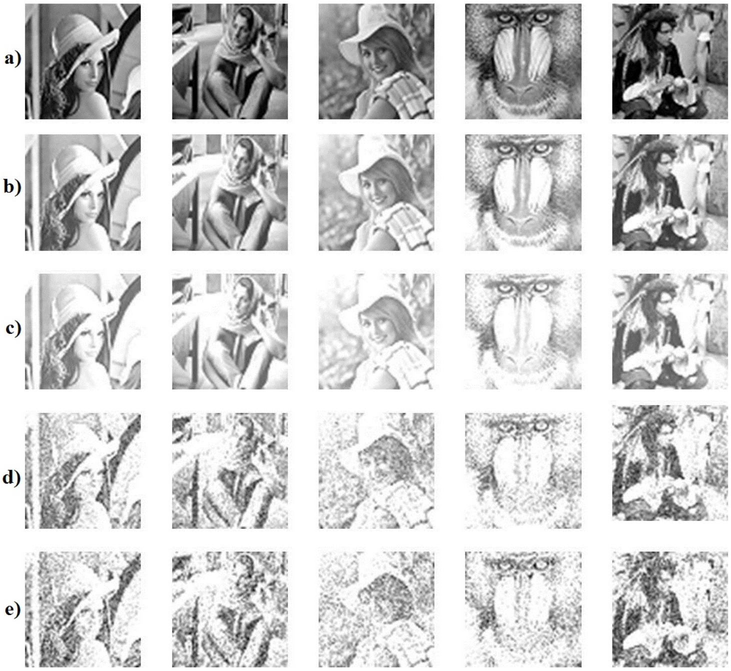


Fig. 4. Images of set-1, (a) original image. Altered images, erosive noise (b) 60%, (c) 100%; mixed noise (d) 30 %, (e) 40%.

Table 3. Performance results (recognition rate) shown by the proposed architecture with altered versions of the test images

Image	Distortion percentage added to image				
	Erosive noise			Mixed noise	
	0%	60%	100%	30%	40%
	100%	100%	100%	80%	50%
	100%	100%	100%	100%	100%
	100%	100%	100%	80%	20%
	100%	100%	100%	100%	50%
	100%	100%	100%	90%	40%

5 Conclusions

The EAM have a number of properties, including be an efficient tool for recognizing and classification patterns, have a high tolerance to dilative, erosive, or mixed noise, efficiently compute the best-match [23]; these properties make them ideal for many applications, e.g. image recognition, area in which this work focuses. However, due to the nature of the application and to the way the EAM encode and store data of patterns, one aspect that represents an obstacle for using EAM in real-time applications is the amount of data that required be processed. To solve this problem and considering that the EAM structure lets evaluated many hypotheses concurrently, FPGA-based parallel hardware architecture for EAM has been proposed. This architecture is composed of *N* sub-components working concurrently and which function is to calculate the degree of belonging of the actual image with each of the existing classes. Furthermore, the implementation of these sub-components is done through four-stage pipeline architecture. The last sub-component that integrates the proposed architecture is a network of N-1 8-bit magnitude comparators connected in cascading and which function is to calculate the index of the class to which belongs the pattern analyzed. With the structure of proposed architecture high processing speeds are obtained and it can be used in real-time image recognition applications.

In order to simplify the proposed architecture and obtain high processing speeds, integer arithmetic operations were used, therefore the EAM presents information loss during the training phase. Despite this, and based on the results, we observe that the proposed architecture presents a perfect recognition with non-altered images and it has high performance with images that have a high percentage of erosive noise. The efficiency of our proposal is compromised when images with 40% of mixed noise are presented.

References

[1] Egmont-Petersen, M., Ridder, D., Handels, H.: Image processing with neural networks-a review. Pattern Recognition 35(10), 2279–2301 (2002)

[2] Er, M.J., Wu, S., Lu, J., Toh, H.L.: Face recognition with radial basis function (RBF) neural networks. IEEE Trans. on Neural Networks 13(3), 697–710 (2002)

[3] Yang, J., Zhang, D., Frangi, A., Yang, J.-Y.: Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence 26(1), 131–137 (2004)

[4] Gottumukkal, R., Asari, V.: An improved face recognition technique based on modular PCA approach. Pattern Recognition Letters 25, 429–436 (2004)

[5] Bezdek, J., Keller, J., Krisnapuram, R., Pal, N.: Fuzzy Models and Algorithms for Pattern Recognition and Image Processing, 1st edn. Springer (2005)

[6] Mitchell, H.B.: Pattern recognition using type-II fuzzy sets. Information Sciences 170, 409–418 (2005)

[7] Bandyopadhyay, S., Maulik, U.: Genetic clustering for automatic evolution of clusters and application to image classification. Pattern Recognition 35(6), 1197–1208 (2002)

[8] Swets, D.L., Punch, B., Weng, J.: Genetic algorithms for object recognition in a complex scene. In: IEEE Int. Conf. on Image Processing, vol. 2, pp. 595–598 (1995)

[9] Vázquez, R.A., Sossa, H.: Associative Memories Applied to Image Categorization. In: Martínez-Trinidad, J.F., Carrasco Ochoa, J.A., Kittler, J. (eds.) CIARP 2006. LNCS, vol. 4225, pp. 549–558. Springer, Heidelberg (2006)

[10] Guzmán, E., Alvarado, S., Pogrebnyak, O., Sánchez Fernández, L.P., Yañez, C.: Hardware Implementation of Image Recognition System Based on Morphological Associative Memories and Discrete Wavelet Transform. In: Mery, D., Rueda, L. (eds.) PSIVT 2007. LNCS, vol. 4872, pp. 664–677. Springer, Heidelberg (2007)

[11] Na, Y.-K., Oh, S.-Y.: Hybrid Control for Autonomous Mobile Robot Navigation Using Neural Network Based Behavior Modules and Environment Classification. Journal Autonomous Robots 15(2), 193–206 (2003)

[12] Du, L., Liu, H., Bao, Z., Xing, M.: Radar HRRP target recognition based on higher order spectra. IEEE Trans. on Signal Processing 53(7), 2359–2368 (2005)

[13] Bartlett, M.S., Movellan, J.R., Sejnowski, T.J.: Face recognition by independent component analysis. IEEE Trans. on Neural Networks 13(6), 1450–1464 (2002)

[14] Zhu, Y., Tan, T., Wang, Y.: Biometric personal identification based on iris patterns. In: 15th Int. Conf. on Pattern Recognition, vol. 2, pp. 801–804 (2000)

[15] Kennedy, J.V., Austin, J., Pack, R., Cass, B.: C-NNAP - A parallel processing architecture for binary neural networks. In: IEEE Int. Conf. on Neural Networks, pp. 1037–1041 (1995)

[16] Heittmann, A., Ckert, U.: Mixed Mode VLSI Implementation of a Neural Associative Memory. Analog Integrated Circuits and Signal Processing 30, 159–172 (2002)

[17] Aldape-Pérez, M., Yáñez-Márquez, C., Argüelles-Cruz, A.J.: FPGA Implementation of Parallel Alpha-Beta Associative Memories. In: Campilho, A., Kamel, M. (eds.) ICIAR 2008. LNCS, vol. 5112, pp. 1081–1090. Springer, Heidelberg (2008)

[18] Sossa, H., Barrón, R., Vázquez, A.: Real-valued Patterns Classification based on Extended Associative Memory. In: Fifth Mexican Int. Conf. on Computer Science, ENC 2004, pp. 213–219. IEEE Computer Society (2004)

[19] Steinbuch, K.: Die Lernmatrix. Kybernetik 1(1), 26–45 (1961)

[20] Barron, R.: Associative Memories and Morphological Neural Networks for Patterns Recall. PhD dissertation, Center for Computing Research - National Polytechnic Institute (2006) (in Spanish)

[21] Riesgo, T., Torroja, Y., de la Torre, E.: Design methodologies based on hardware description languages. IEEE Trans. on Industrial Electronics 4(1), 3–12 (1999)

[22] Elléouet, D., Savary, Y., Julien, N.: An FPGA Power Aware Design Flow. In: Vounckx, J., Azémard, N., Maurine, P. (eds.) PATMOS 2006. LNCS, vol. 4148, pp. 415–424. Springer, Heidelberg (2006)

[23] Guzmán, E., Ramírez, M., Pogrebnyak, O.: Applied Extended Associative Memories to High-Speed Search Algorithm for Image Quantization. InTech Publisher (2011)

Salient Features Selection for Multiclass Texture Classification

Bharti Rana and R.K. Agrawal

School of Computer & Systems Sciences, Jawaharlal Nehru University, New Delhi-110067
{bhartirana.jnu,rkajnu}@gmail.com

Abstract. Texture classification is one of the important components in texture analysis which has drawn the attention of research community during the past few decades. Various texture feature extraction techniques have been proposed in the literature. However, combining texture methods from different families has demonstrated to produce better classification at the cost of complexity of the learning model. In this paper, we have investigated three parametric test statistics (ANOVA F statistic, Welch test statistic, Adjusted Welch test statistic) to determine salient features for multiclass texture classification. The salient features are obtained from a pool of features obtained using five textural feature extraction methods. Experiments are performed on a widely used publicly available Brodatz dataset. Experimental results show that the classification error decreases significantly with the use of all the three feature selection methods with all classifiers. The reduced set of features will also lead to significant decrease in computation time of the learning model.

Keywords: Texture classification, feature extraction, feature selection.

1 Introduction

Texture analysis is one of the basic components in image processing and computer vision and have been utilized more often in variety of application domains e.g. automated detection of defects and quality control of texture images [1], medical diagnosis [2], microscope images [3], postal address recognition and interpretation of maps [4], remote sensing [5], geological images [6], etc.

Texture classification is one of the major issues in texture analysis which has received considerable attention during the past few decades. It is a process to determine the category of texture from a set of known texture patterns to which a given image or sub-image belongs. Performance of texture classification depends mainly on two components: (i) the choice of features which represent the texture of an image and (ii) classification methods. In literature, numerous structural, statistical, model-based feature extraction techniques have been proposed to extract texture features. Some of the commonly employed approaches to extract texture features are: Gray level co-occurrence matrix (GLCM) [7], Fourier descriptors [8], Gabor filters [9-10], Discrete Wavelet transform [11-12], Fractal dimension [13-14], etc.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 205–215, 2013.

However, every feature extraction method is potentially useful to distinguish texture pattern to a larger or lesser extent. It was shown in the research work [15] that the proper integration of texture methods from different families leads to better classification than those obtained with a single feature extraction method.

However, the size of resultant feature vector after integration of texture features from all methods will be large. It may degrade the performance of classifier in terms of classification accuracy. Due to large features, it may also require high computation time and other resources for training and testing the data. Thus, determining a minimum subset of salient texture features which maximizes the final texture classification accuracy is still a challenging problem.

Feature selection for classification aims at selecting a feature subset without significantly decreasing the accuracy that the classifier reaches when it utilizes all the available features [16]. Hence, feature selection algorithms can be applied to obtain a subset of salient texture features for better multiclass texture classification.

In this paper, we investigate three popular parametric test statistics (ANOVA F statistic [17], Welch test statistic [18], and Adjusted Welch test statistic [19]) to determine salient texture features from a pool of texture features obtained using five textural feature extraction methods: (i) GLCM, (ii) Fourier descriptors, (iii) Gabor filters, (iv) Discrete Wavelet transform, and (v) Fractal method.

The paper is organized as follows: Section 2 contains a brief introduction to feature extraction techniques. Feature selection techniques are discussed in section 3. Experimental results are described in section 4 and finally, conclusion and future work is presented in section 5.

2 Feature Extraction Techniques

Feature extraction techniques used to represent texture of an image are follows:

2.1 Gray-Level Co-occurrence Matrix

Gray-level Co-occurrence (GLCM) [7] measures the relationship between groups of two (usually neighboring) pixels in the original image and determines how often gray values co-occur at two pixels separated by a fixed distance and an orientation. A co-occurrence matrix $P_{d,\theta}$, is a two-dimensional array of size $N \times N$, where N is the number of gray levels in the image. The GLCM is defined as

$$p_{d,\theta}[i,j] = n_{ij} \tag{1}$$

where n_{ij} is probability of transition from a pixel with intensity '*i*' to a pixel with intensity '*j*' lying at distance d with a given orientation θ in the image.

2.2 Fourier Transform

The Fourier transforms (FT) is a mathematical formulation to transform the data from space domain to frequency domain i.e. it provides representation of an image based on its frequency content. Given a 2D image or a function *f(x,y)* of size $M \times N$, the discrete Fourier transformation is given by [20]:

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-i2\pi(ux/M+vy/N)} \qquad (2)$$

with u=0,1,...,M-1 and v=0,1,...,N-1.

2.3 Gabor Filter

The 2-D multi-channel Gabor filter [9-10] is a windowed signal processing method. A typical 2D Gabor model used in the texture analysis is given as follows:

$$g(x,y;\lambda,\theta,\psi,\sigma,\gamma) = \exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right)\exp\left(i\left(2\pi\frac{x'}{\lambda}+\psi\right)\right) \qquad (3)$$

where $x' = x\cos(\theta) + y\sin(\theta)$, $y' = -x\sin(\theta) + y\cos(\theta)$, λ represents the wavelength of the sinusoidal factor, θ represents the orientation of the normal to the parallel stripes of a Gabor function, ψ is the phase offset, σ is the sigma of the Gaussian envelope and γ is the spatial aspect ratio that specifies the ellipticity of the support of the Gabor function.

A set of wavelengths and orientations is selected to retrieve features from an image. Gabor filters have tunable orientation, radial frequency bandwidths, tunable center frequencies, and optimal joint resolution in spatial and frequency domain. These features make it a powerful and commonly used texture analysis technique.

2.4 Wavelet Transform

Multi-resolution analysis [21] allows information conservation of an image according to certain levels of resolution or blurring. It allows zooming in and out on the underlying texture structure. Therefore, the texture extraction is not affected by the size of the pixel neighborhood. Because of this quality, wavelets have been useful in many applications such as image compression, image de-noising and edge detection.

Wavelets are mathematical functions that decompose data into different frequency components and then study each component with a resolution matched to its scale. It provides a more flexible way of analysis of both space and frequency contents by allowing the use of variable sized windows [21]. By decomposing the image into a series of high-pass and lowpass bands, the wavelet transform extracts directional details that store horizontal, vertical and diagonal information (edge).

A wavelet can decompose a signal or an image with a series of averaging and differencing operations. Wavelets work out average intensity properties as well as several detailed contrast levels distributed throughout the image. The general mother wavelet can be constructed from the following scaling $\phi(x)$ and wavelet functions $\psi(x)$:

$$\phi(x) = \sqrt{2}\sum h(k)\phi\left(2x - k\right) \tag{4}$$

$$\psi(x) = \sqrt{2}\sum g(k)\phi\left(2x - k\right) \tag{5}$$

where $g(k)=(-1)^k h(N-1-k)$, and N is the number of scaling and wavelet coefficients. The sets of scaling ($h(k)$) and wavelet ($g(k)$) function coefficients vary depending on their corresponding wavelet bases.

With more level of decomposition, compact but coarser approximation of the image is obtained. Thus, wavelets provide a simple hierarchical framework for better interpretation of the image information [21].

2.5 Fractal Analysis

A fractal is a rough or fragmented geometric shape that can be subdivided into many parts, each of which is approximately a reduced-size replica of the whole. Although, there are many fractal dimension estimation methods but Box-Counting method [14] is one of the well known and commonly used methods to estimate the fractal dimension of an object. It is easy to implement and involves simple computation.

In Box-Counting method, an image $A \in R^2$ is covered with boxes of size *r*. The number of boxes, *N(r)*, that cover the image is counted. This procedure is repeated for different values of *r*. The value of the fractal dimension *D* is estimated using:

$$D = -\lim_{r \to 0} \frac{\log(N(r))}{\log(r)} \tag{6}$$

Fractal dimension is a measure of complexity or roughness of an object. Intuitively, the fractal dimension increases with increase in the roughness of the texture [22].

3 Feature Selection Techniques

Feature selection is an important issue in classification and used for removing (selecting) irrelevant (salient) features. In this section, we will describe a general test statistical model for testing the equality of the class means. We then discuss our approach to select salient features using power and correlation.

3.1 Statistical Model

Assume there are k (≥ 2) distinct texture classes for the problem under consideration and there are p features (inputs) and n texture image samples (observations). Suppose

X_{fs} is the measurement of the feature *f* from sample *s* for $f = 1, 2, ..., p$ and $s = 1, 2, ..., n$. Data can be represented in terms of a matrix F and is given by

$$F = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ X_{p1} & X_{p2} & \cdots & X_{pn} \end{bmatrix} \tag{7}$$

Columns and rows of the matrix *F* correspond to samples and features, respectively. Note that *F* is a matrix consisting of data highly processed through preprocessing techniques such as image analysis and normalization. We assume that the data matrix *F* is standardized so that the features have mean 0 and variance 1 across samples. Given a fixed feature, let Z_{ij} be the feature from the j^{th} sample of the i^{th} class where Z_{ij} come from the corresponding row of *F*. For example, for feature 1, Z_{ij} is a rearrangement of the first row of *F*. We consider the following general model for Z_{ij}:

$$Z_{ij} = \mu_i + \varepsilon_{ij} \text{ for } i = 1, 2, \ldots, k \text{ ; } j = 1, 2, \ldots, n_i \tag{8}$$

with $n_1 + n_2 + \cdots + n_k = n$. In the model, μ_i is a parameter representing the mean value of the feature in class *i*, ε_{ij} are the error terms such that ε_{ij} are independent normal random variables, and

$$E(\varepsilon_{ij}) = 0, \; V(\varepsilon_{ij}) = \sigma_i^2 < \infty \tag{9}$$

for $i=1,2,...,k; j=1,2,...,n_i$.

Note that if the variances are equal, that is, $\sigma_1^2 = \sigma_2^2 = \cdots = \sigma_k^2$, then the above model is simply the commonly used one-way ANOVA model. For the texture dataset, we believe that heterogeneity in the variances is more realistic, since different σ_i may describe different variations of the feature across classes.

One of the main tasks associated with the above model is to detect whether or not there is some difference among the means, that is, $H_0 : \mu_1 = \mu_2 = \cdots = \mu_k$ versus H_1: not all μ_i are equal. For the case of homogeneity of variances, the well-known ANOVA F test is the optimal test to accomplish the task [17, 23]. However, with heterogeneity of the variances, the task is challenging. Therefore, some alternatives to the F test are worthy of investigation.

3.2 Test Statistics

We will consider the following parametric test statistics.

1. ANOVA F test statistics is a parametric test statistics. It measures ratio of between group variability and with-in group variability, and is defined as:

$$F = \frac{(n-k)\sum_i n_i(\overline{Z}_{i.} - \overline{Z}_{..})^2}{(k-1)\sum_i (n_i - 1)s_i^2} \tag{10}$$

where $\overline{Z}_{i.} = \sum_{j=1}^{n_i} Z_{ij} \big/ n_i$, $\overline{Z}_{..} = \sum_{i=1}^{k} n_i \overline{Z}_{i.} \big/ n$, and $s_i^2 = \sum_{j=1}^{n_i} (Z_{ij} - \overline{Z}_{i.})^2 \big/ (n_i - 1)$.

Under H_0 and assuming variance homogeneity, this well-known test statistic has a distribution of $F_{k-1,n-k}$ [17].

2. Welch test statistic [18] is a parametric test and is defined as

$$W = \frac{\sum_i w_i\left(\overline{Z}_{i.} - \sum h_i \overline{Z}_{i.}\right)^2}{(k-1) + 2(k-2)(k+1)^{-1}\sum_i (n_i - 1)^{-1}(1-h_i)^2} \tag{11}$$

with $w_i = n_i \big/ s_i^2$ and $h_i = w_i \big/ \sum_i w_i$. Under H_0, W has an approximate distribution of F_{k-1,ν_w}, where

$$\nu_w = \frac{k^2 - 1}{3\sum_i (n_i - 1)^{-1}(1-h_i)^2} \tag{12}$$

3. Adjusted Welch test statistic [19] is similar to the Welch test statistic and is given by

$$W^* = \frac{\sum_i w_i^*\left(\overline{Z}_{i.} - \sum_i h_i^* \overline{Z}_{i.}\right)^2}{(k-1) + 2(k-2)(k+1)^{-1}\sum_i (n_i - 1)^{-1}(1-h_i^*)^2} \tag{13}$$

where $w_i^* = n_i \big/ \left(\phi_i s_i^2\right)$ with ϕ_i chosen such that $1 \le \phi_i \le \left(n_i - 1\right) \big/ \left(n_i - 3\right)$, and $h_i^* = w_i^* \big/ \sum_i w_i^*$. Under H_0, W^* has an approximate distribution of F_{k-1,ν_w^*}, where

$$\nu_w^* = \frac{k^2 - 1}{3\sum_i \left(n_i - 1\right)^{-1} \left(1 - h_i^*\right)^2} \tag{14}$$

In this paper, we choose $\phi_i = \left(n_i + 2\right) \big/ \left(n_i + 1\right)$, since this choice provides reliable results for small sample sizes n_i and a large number (*k*) of populations [19].

We use the test statistics to determine the power of features in discriminating between texture images. Given a test statistic F, we define the discrimination power of a feature as the value of F evaluated at the n levels of the feature. This definition is based on the fact that with larger F the null hypothesis $H_0 : \mu_1 = \mu_2 = \cdots = \mu_k$ will be more likely rejected. Therefore, the higher the discrimination power is, the more powerful the feature is in discriminating between different texture images. Finally, we choose those features as salient features which have high power of discrimination.

4 Experimental Results

Experiments are performed on commonly used benchmark dataset i.e Brodatz [24] for evaluating different feature extraction and selection methods. Dataset consists of 1110 images grouped in 111 classes. Each class is constructed by dividing an image into nine non-overlapping samples and one overlapping, thus forming 10 samples per class. Each image is of size 200*200 with 256 gray levels. Fig. 1 shows some examples of the texture classes from Brodatz album [24].

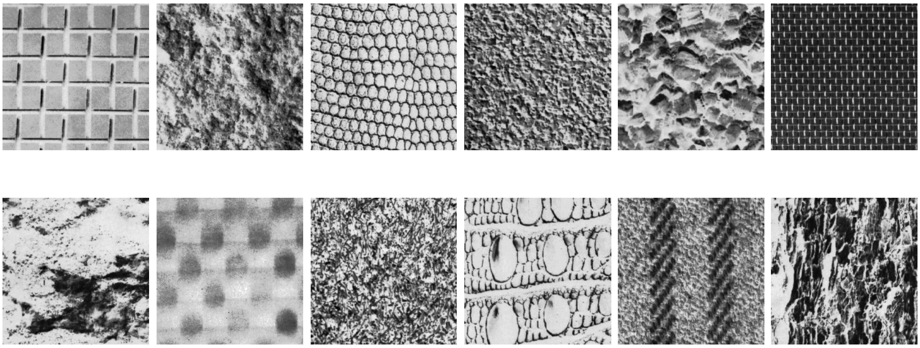


Fig. 1. Some examples of texture classes from Brodatz album

All experiments are carried out using Windows 7 environment over Pentium-IV machine, with 3 GB RAM and a processor speed of 2.39 GHz. The Matlab tools used for experiments are: Image Processing Toolbox, Wavelet Toolbox, Prtools [25], libSVM [26].

A pool of 219 texture features is obtained using the textural feature extraction methods: (i) GLCM, (ii) Fourier descriptors, (iii) Gabor Filters, (iv) Discrete Wavelet transform, and (v) Fractal method. The following paragraphs briefly describe the construction of feature vector from these feature extraction techniques.

GLCM. In the experiment, co-occurrence matrices are constructed using $d = \{1,2\}$ and $\theta = \{0^o, 45^o, 90^o, 135^o\}$. The feature vector composed of energy and entropy values computed from this matrix is constructed. Thus, 16 features are extracted from an image.

Fourier Descriptors. The coefficients of the Fourier transform are obtained from an image and shifting operation is used so that low frequencies are shifted to the center of the spectrum. Each feature corresponds to the sum of the absolute values of spectrum for a given radius *r* from the center of the spectrum. 99 different radius values are used to obtain 99 texture features from an image.

Gabor Filters. In this work, 16 features, based on energy, is calculated by convolving input image with 16 Gabor filters. These Gabor filters are constructed using following values of parameters: θ (orientation) = $\{0^o, 45^o, 90^o, 135^o\}$, λ (wavelength) = $\{2, 4, 8, 16\}/sqrt(2)$, ψ (phase offset) = 0^o, σ (sigma of the Gaussian envelope) = $0.5622 \times \lambda$ and γ (spatial aspect ratio) = 0.5.

Wavelet Descriptors. Daubechies-4 (db4) wavelet with 4 level of decomposition is used. Mean, energy and entropy features are extracted from horizontal, vertical, diagonal component at each level of decomposition and approximation component retrieved from 4th level of decomposition. Thus, feature vector of size 39 is obtained from an image.

Multi-Level Fractals. In this experiment, we employed multi-level fractal dimension estimation which uses a set of threshold values. These threshold values are computed as mentioned in research work [13]. Using these threshold values, a set of binary images are retrieved and from these binary images fractal dimension is estimated. We consider 49 such threshold values. Thus, a feature vector of length 49 is constructed from an image.

From the pool of these features, we selected 40 important features based on ranking of three different parametric feature selection statistics: ANOVA F test, Welch, Adjusted Welch test statistics. Features are added in increment of size one as per ranking and classification error is calculated. Three classifiers linear discriminant analysis (LDA), K-nearest neighbors (KNN) and support vector machine (SVM) are used. For KNN classifier, optimal value of *K* is chosen which minimizes the classification error. Leave-one-out cross-validation (LOOCV) scheme is used for training and testing of image samples. Table 1 represents the performance error of each individual feature extraction technique. Table 1 also shows the minimum classification error obtained corresponding to a minimum size subset of salient features (shown in brackets), when feature selection is applied on a pool of 40 features. The best result for a classifier is shown in bold.

Table 1. Performance of different texture feature extraction and selection techniques

		Error rate (in %)		
		LDA	**KNN**	**SVM**
Individual Methods	**GLCM**	12.79	27.75	12.52
	Fourier Descriptors	13.51	10.36	6.76
	Gabor filters	10.09	9.01	7.03
	Wavelet Descriptor	9.19	20.81	11.08
	Muti-level fractals	11.08	45.59	26.21
Feature Selection on a pool of 219 Features	**ANOVA F Test**	7.48(28)	**7.93(32)**	**4.96(28)**
	Welch Test	5.50(37)	10.00(32)	6.13(35)
	Adjusted Welch Test	**5.41(33)**	10.36(31)	6.13(35)

*Number of features corresponding to minimal classification error is shown in brackets.

We can observe the following from Table 1:

1. The classification error decreases significantly in comparison to all individual feature extraction methods with the use of all the three feature selection methods with all classifiers except one case.
2. Overall, the performance of feature selection using ANOVA F Test statistics is better in comparison to other two feature selection statistics.
3. The performance of LDA and SVM is better in comparison to KNN.
4. The classification error depends on the choice of feature extraction/selection methods and classifier.
5. The minimum texture classification error obtained is 4.96 with the combination of ANOVA F Test statistics and SVM classifier.

5 Conclusion and Future Work

In this work, we have investigated three parametric test statistics (ANOVA F statistic, Welch test statistic, Adjusted Welch test statistic) to determine salient features for multiclass texture classification. The salient features are obtained from a pool of features obtained using five textural feature extraction methods: (i) GLCM, (ii) Fourier descriptors, (iii) Gabor Filters, (iv) Discrete Wavelet transform, and (v) Fractal method. Experiments are performed on widely used publicly available Brodatz dataset. Experimental results show that the classification error decreases significantly with the use of all the three feature selection methods with all classifiers. The performance of ANOVA F Test statistics is better among the three feature selection statistics used in our experiment.

Features selected based on ranking may contain redundancy. In future, some feature selection methods can be investigated to further improve the performance of multiclass texture classification.

References

1. Li, W.-C., Tsai, D.-M.: Wavelet-based defect detection in solar wafer images with inhomogeneous texture. Pattern Recognition 45(2), 742–756 (2012)
2. Oliveira, M., Fernandes, P., Avelar, W., Santos, S., Castellano, G., Li, L.: Texture analysis of computed tomography images of acute ischemic stroke patients. Clinical Radiology 42(11), 1076–1079 (2009), doi:10.1590/S0100-879X2009005000034
3. Backes, A.R., de M. Sá Junior, J.J., Kolb, R.M., Bruno, O.M.: Plant Species Identification Using Multi-scale Fractal Dimension Applied to Images of Adaxial Surface Epidermis. In: Jiang, X., Petkov, N. (eds.) CAIP 2009. LNCS, vol. 5702, pp. 680–688. Springer, Heidelberg (2009)
4. Jain, A.K., Bhattacharjee, S.K.: Address block location on envelopes using Gabor filters. Pattern Recognition 25(12), 1459–1477 (1992)
5. Yang, C.Z.X.: Study of remote sensing image texture analysis and classification using wavelet. International Journal of Remote Sensing 19(16), 3197–3203 (1998)
6. Heidelbach, F., Kunze, K., Wenk, H.R.: Texture analysis of a recrystallized quartzite using electron diffraction in the scanning electron microscope. Journal of Structural Geology, 91–104 (2000)
7. Haralick, R.M.: Statistical and structural approaches to texture. Proceedings of the IEEE 67(5), 786–804 (1979)
8. Azencott, R., Wang, J.-P., Younes, L.: Texture classification using windowed fourier filters. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(2), 148–153 (1997)
9. Jain, A.K., Farrokhnia, F.: Unsupervised texture segmentation using Gabor filters. Pattern Recognition 24(12), 1167–1186 (1991)
10. Idrissa, M., Acheroy, M.: Texture classification using gabor filters. Pattern Recognition Letters 23(9), 1095–1102 (2002)
11. Chang, T., Kuo, C.-C.: Texture analysis and classification with tree-structure wavelet transform. IEEE Transactions on Image Processing 2(4), 429–441 (1993)
12. Randen, T., Husøy, J.H.: Filtering for texture classification: a comparative study. IEEE Transactions on Pattern Analysis and Machine Intelligence 21(4), 291–310 (1999), doi:10.1109/34.761261
13. Backes, A.R., Bruno, O.M.: A New Approach to Estimate Fractal Dimension of Texture Images. In: Elmoataz, A., Lezoray, O., Nouboud, F., Mammass, D. (eds.) ICISP 2008 2008. LNCS, vol. 5099, pp. 136–143. Springer, Heidelberg (2008)
14. Tricot, C.: Curves and Fractal Dimension. Springer, Heidelberg (1995)
15. Puig, D., García, M.A.: Pixel classification through divergence-based integration of texture methods with conflict resolution. In: IEEE ICIP, Barcelona, Spain (2003)
16. John, G.H., Kohavi, R., Pfleger, K.: Irrelevant features and the subset selection problem. In: Proceedings of the 11th International Conference on Machine Learning, New Brunswick, NJ, USA, pp. 121–129 (1994)
17. Neter, J., Kutner, M.H., Nachtsheim, C.J., et al.: Applied Linear Statistical Models, 4th edn. McGraw-Hill, Chicago (1996)

18. Welch, B.L.: On the comparison of several mean values: An alternative approach. Biometrika 38, 330–336 (1951)
19. Hartung, J., Argac, D., Makambi, K.H.: Small sample properties of tests on homogeneity in oneway ANOVA and meta-analysis. Statist. Papers 43, 197–235 (2002)
20. Gonzalez, R., Woods, R.: Image Enhancement in the Frequency Domain. In: Digital Image Processing, 2nd edn., pp. 349–408. Pearson Education (2004)
21. Mallat, S.G.: A theory of multiresolution signal decomposition: the wavelet representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 11(7), 674–693 (1980)
22. Chen, C.H., Pau, L.F., Wang, P.S.P. (eds.): The Handbook of Pattern Recognition and Computer Vision, 2nd edn., pp. 207–248. World Scientific Publishing Co. (1998)
23. Lehman, E.L.: Testing Statistical Hypotheses, 2nd edn. Wiley, NY (1986)
24. Brodatz, P.: Textures A Photographic Album for Artists and Designer. Dover, New York (1996)
25. Duin, R., Juszcak, P., Paclik, P., Pekalska, E., De Ridder, D., Tax, D.: Prtools, a matlab toolbox for pattern recognition, `http://www.prtools.org` (accessed 2004)
26. Chang, C.-C., Lin, C.-J.: LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2, 1–27 (2011), Software `http://www.csie.ntu.edu.tw/~cjlin/libsvm`

Robotic Behavior Implementation Using Two Different Differential Evolution Variants

Víctor Ricardo Cruz-Álvarez[1], Fernando Montes-Gonzalez[1],
Efrén Mezura-Montes[2], and José Santos[3]

[1] Departmet of Artificial Intelligence, Universidad Veracruzana, Mexico
victor.g2004@gmail.com, fmontes@uv.mx
[2] Laboratorio Nacional de Informática Avanzada, Mexico
emezura@lania.mx
[3] Computer Science Department, University of A Coruña, Spain
santos@udc.es

Abstract. In Evolutionary Robotics, Bioinspired Algorithms are used to generate robotic behavior. Several researchers used classic Genetic Algorithms or adaptations of Genetic Algorithms for developing experiments in ER. Here, we use Differential Evolution as an evolutionary alternative method to automatically obtain robotic behaviors, selecting a wall-following behavior as a representative example. We used an e-puck robot and the Player-Stage simulator for the experiments. We detail the results and the advantages when using the DE variants in our application with the simulated and the real robot. In order to optimize time for evolution, and test the resultant behavior in the e-puck robot, for our experiments we employed the Player-Stage simulator.

Keywords: Evolutionary Robotics, Differential Evolution, Genetic Algorithm, Behavior Based Robotics.

1 Introduction

In robotics, Evolutionary Robotics (ER) is an area that uses Bioinspired Algorithms, especially evolutionary algorithms, to develop the control architecture of an autonomous robot, and particularly the behavior controllers[1]. ER operates creating an initial population of candidate controllers and the population is repeatedly modified based on the fitness-function to optimize the robot controller[2]. The most used approach in ER is the classic Genetic Algorithm (GA)[3, 4], which generates solutions to optimization problems using operators inspired by natural evolution, these operators are crossover and mutation. Take for instance the work of Trefzer, et ál.[5] where an e-puck had been provided with a general purpose obstacle-avoidance controller both in simulation and in the real robot using GA. In similar works like[6–10, 12–14] like Evolutionary Strategies[15] and Differential Evolution (DE)[16]; some works in ER uses Genetic Programming [17] to evolve the robot controller[11, 18].

In general, ER employs GAs as optimization method, whereas uses Artificial Neural Network (NN) as the elements to evolve, since the NNs implement the

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 216–226, 2013.

robot controllers. The evolutionary method, after several generations, produces the fittest individuals (NNs) that generate the required behavior. The calibration of the associated parameters of the evolutionary method increments the chances to find near-optimal solutions.

The use of a standard platform in ER is commonly preferred. The e-puck[19] is an educational and research robot developed in the EPFL (École Polytechnique Féderale de Lausanne, Switzerland). In our experiments we use this robot. The size of the e-puck is about 7 cm diameter; the battery provides movement autonomy for about 3 hours, two stepper motors with a 20 steps per revolution and a 50:1 reduction gear. The e-puck has a ring of eight infrared sensors measuring ambient light and proximity of obstacles in a range of 4 cm. and has a VGA camera with a field of view of 36; for communication, the robot uses a wireless Bluetooth. The robot is controlled by a Microchip microprocessor dsPIC 30F6014A at 60MHz. The e-puck can be simulated under free and commercial applications, in this work, we use the Player-Stage simulator.

In the experiments, to evolve the wall-following behavior, we use an e-puck robot, the Player-Stage simulator and two DE variants for the evolutionary optimization. These variants are the *DE/rand/1/bin* and the *DE/best/1/ bin.* We compare the results of using these differents variants. In Section 2, we discuss the methods mentioned. In Section 3, some topics related to Player-Stage are discussed. Section 4 describes the parameters for the neural network, the evolutionary methods and the e-puck robot. Results are presented in Section 5. Finally, in Section 6 we provide a general conclusion about our work.

2 Evolutionary Robotics and Bioinspired Algorithms

Generally, ER relies on the use of a Bioinspired Algorithms (generally the GA with binary representation [2, 7, 9]). The population is a set of candidate controllers. Most of the times a single objective evaluation function (fitness function) is employed by the fact that the resultant robot behavior comes from a dynamic system made with the robot an its environment[20]. The use of Bioinspired Algorithms and Artificial Neural Networks offers a good solution to the problem of modeling behavior in maze-like environments[1]. Artificial Neural Networks have many applications in robotics due to their benefits as powerful classifiers, they are both noise and fault tolerant, which facilitates the robot to be driven in dynamical environments[21].

The optimization of neural controllers with an evolutionary method requires a representation, as a vector of the weights of the neural controller. Therefore, the codification of the weights vectors using an array representation is a common practice. This array represents the genetic material to be manipulated by artificial evolution. A single neural controller represents one of the many individuals that form a population, which in turn are candidates for providing a good solution for the task to be solved.

The quality of a candidate solution (*fitness*) is measured to acknowledge whether a solution is a good solution to the task we are trying to solve. The quality-space of all possible solutions forms the fitness landscape, with mountains and valleys, where landmarks in the mountains represents good solutions and landmarks in valleys are poor solutions. The search of the best solution in the landscape depends on the chosen evolutionary method. The search of the best solution requires the guide of an evolutionary method to move uphill to find better solutions, and variations in the landscape may cause the search to revolve around local minima and maximum areas.

In ER, the GA uses a population to represent the solutions (binary representation); for exploration the GA employs the next operators: the reproduction of the individuals selected in pairs by the *crossover* of their genetic material, and *mutation* of some of the genetic material of the new individuals in the next generation. Also the GA uses *selection*, where a subset of population is selected to produce the next generation. Finally the GA *replace* the population with the new individuals. The iterative application of these operators to the genetic material (the NN weights) will produce refined solutions over time.

On the other hand, DE uses a population of real numbers (called vectors) that in our case represents the weights. The main idea behind DE is a scheme for generating trial parameter-vectors based on the population distribution. The basic idea of the DE is very simple, it works with two populations: P (the old generation) and Q (the new generation) of the same size N (the size of population). A new trial vector y is composed of the current point x_i of the old generation and a new point u obtained by using *mutation.* If *f(y)* $>$ *f(*x_i*)* (in case of fitness maximization), the point y is inserted, instead of x_i, into the new generation Q. After completion of the new generation Q, the old generation P is replaced by Q and the search continues until the stopping condition is reached[23]. In related work different variants exist for producing the mutation vector. In this work we use the *DE/rand/1/bin* and *DE/best/1/bin*, which generate the point u by adding the weighted difference of two points.

$$u = r_1 + F(r_2 - r_3). \tag{1}$$

Equation 1 shows how a new point u is created; r_1, r_2 and r_3 in the *DE/rand/1 /bin* are randomly selected from P. For *DE/best/1/bin*, r_1 is the best individual in the population, r_2 and r_3 are both randomly selected.

As for the DE parameters, we have F, CR and N, which respectively represent the differential weight, F that takes the next values $F \in [0, 1]$; CR which in turn represents the crossover probability with possible values in $CR \in [0, 1]$; and the last parameter represents the population size N. A general description of the DE pseudocode is presented below (Algorithm 1).

For our work we have selected DE due to an easy parameter calibration, easy encoding, and a refined exploration search in the fitness landscape. In [22], the authors demonstrate the effectiveness of the algorithm establishing a comparison with other different evolutionary methods related to Artificial Neural Networks optimization. In some works like [24, 25] the DE had been used in robotics for mapping and localization tasks, in [26] the DE was used to find a minimal representation for a multi-sensor fusion.

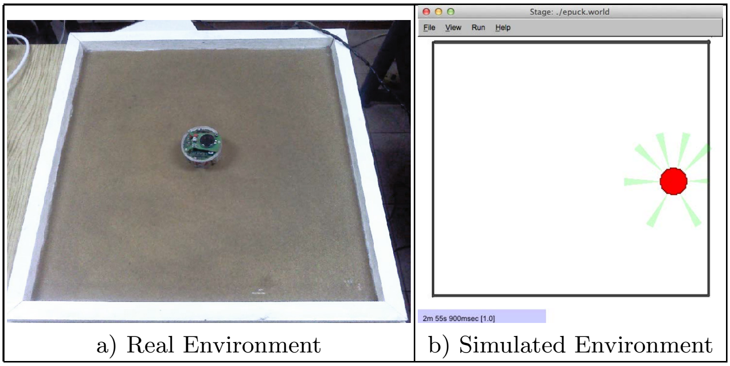
Algorithm 1. Differential Evolution Algorithm

```
Initialize the population with random positions, $P = (x_1; x_2; ...; x_N)$
repeat
    for each individual, $i = 1 \to N$ do
        select randomly $r_1, r_2, r_3$ of $P$
        select randomly $R \in 1, ..., n$ (n is the number of dimensions of the problem)
        for each position, $j = 1 \to n$ do
            select an uniformly distributed number $rand_j \in U(0, 1)$
            if $rand_j < CR$ or $j = R$ then
                generate a mutated position $u_{ij}$, being
                $y_j = u_{ij} \leftarrow r_{1j} + F(r_{2j} - r_{3j})$(Eq.1)
            else
                $y_j = x_{ij}$
            end if
        end for
        if $f(y) > f(x_i)$ then
            insert $y$ in $Q$
        else
            insert $x_i$ in $Q$
        end if
    end for
    $P = Q$
until reaching the stop condition
```

The main problem of the GA methodology is the need of tuning a series of parameters for the different genetic operators like crossover or mutation, the decision of the selection (tournament, roulette,...), and tournament size. Hence, in a standard GA is difficult to control the balance between exploration and exploitation. On the contrary, DE reduces parameter tuning and provides an automatic balanced search. As Feoktistov [27] indicates, the fundamental idea of this algorithm is to adapt the step-length $(F(x_2 - x_3))$ intrinsically along the evolutionary process. For the initial generations the step-length is large because individuals are far away from each other. As evolution goes on, population converges and the step-length becomes smaller and smaller, providing an automatic search exploration level.

3 Robot Software Simulator

Player-Stage is a free software tool for robot and sensor applications[28]. Player provides a network interface to a variety of robot and sensor hardware such as: the Khepera, Pioneer, Lego mindstorm, E-puck, etc. Player's client/server model allows robot control programs to be written in any programming language and to run in any computer with a network connection to the robot. Also it supports multiple concurrent client connections to devices, creating possibilities for distributed and collaborative sensing and control. On the other hand, Stage simulates a population of mobile robots moving and sensing in a two-dimensional bitmapped environment. Various sensor-models are provided, which includes a sonar, scanning laser rangefinder, pan-tilt-zoom camera with color blob detection, and an odometry sensor. It is important to notice that Stage devices offer a standard Player interface with little required changes for using the same code in simulation and hardware. For our experiments we implemented, in Player-Stage, a squared arena delimited by four walls which includes an e-puck robot, This simulated arena is similar in dimensions to a real arena set in our laboratory (see Fig.1).



a) Real Environment

b) Simulated Environment

Fig. 1. a) The physical robot is set inside an arena delimited by four walls. b) The virtual scenario is modeled as a squared arena, in Player-Stage, where we set the e-puck robot (red circle) with eight proximity sensors (in green).

4 Robotic Experimental Setup

The robot controller consists of a neural controller fully connected (see Fig.2) with 8 input neurons that encode the activation states of the corresponding 8 infrared sensors. Values for activation are normalized between 0 and 1, where 0 means that an obstacle is detected nearby, and 1 represents no obstacle detection. Additionally, 4 internal neurons receive connections from sensory neurons and send connections to 2 motor neurons. The first motor neuron is used to define the linear velocity, whereas the second one defines the angular velocity. The output and the internal neurons use a sigmoid transfer function.

The robot controllers were evolved, using DE as the evolutionary approach, for automatically obtain the wall-following behavior. Based on the work of Mezura et ál.[29], for solving a high-dimensionality problem, we have chosen the next DE variants: *DE/rand/1/bin* and *DE/best/1/bin*; with the following parameters: *DE/rand/1/bin* with F=0.7, CR=1.0, N=120; and the *DE/best/1/bin* with: F=0.8, CR=1.0, N=150.

For our experiments we evolved neural controllers in the Player-Stage simulator. The evolutionary process is computed by the DE with the above variants and parameters. The algorithm is left to run until an optimal level is reached. The

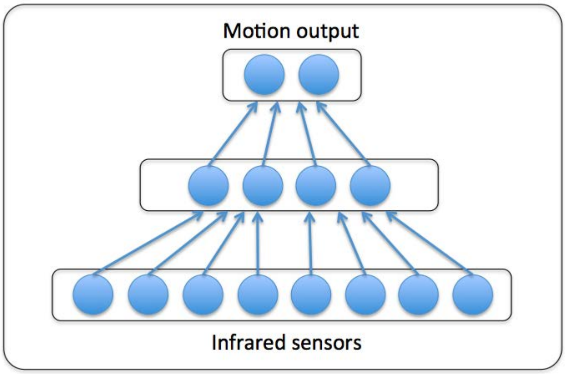


Fig. 2. The Neural Network used as robot controller for the e-puck robot, composed by 8 neurons in the input layer for the infrared sensors, 4 neurons in the hidden layer and the output layer made of 2 neurons for controlling robot-motion (notice that not all connections are shown).

fitness function looks for those individuals able to locate and follow a wall without crashing. The fitness function is defined in equation (2). Fitness is averaged during 5 different trials for an individual tested with initial random positions and orientations. The average fitness is scored as the individual fitness.

$$f = \sum_{i=0}^{n} [(linearVelocity) * (1 - abs(angularVelocity) * (1 - min\_ir))] \qquad (2)$$

where: *linearVelocity* $\in \{0, 1\}$, 0 represents the minimum value of velocity and 1 its maximum value; *angularVelocity* $\in \{-1, +1\}$, -1 represents a left-turn, +1 a right-turn and 0 forward-movement; *min_ir* $\in \{0, 1\}$ is the minimum reading of the 8 infrared sensors. Infrared sensors take values of $\in \{0, 1\}$, where 1 represents that no-obstacle is detected; in contrast 0 is obtained when an object is detected in the vicinity of the robot body. When a robot crashes, calculation of the fitness is stopped and its current value is returned as the scored fitness.

The fitness formula is set to maximize linear velocity while minimizing both the angular velocity and robot-distance to the wall. In other words, the fitness function looks for moving forward quickly and running parallel to walls. In summary, a neuro-controller was evolved in order to develop wall-following behavior using the DE variants.

5 Results

In order to verify the quality of the evolved solutions, experiments are replicated at least three times due to the expensive time execution. The best results are

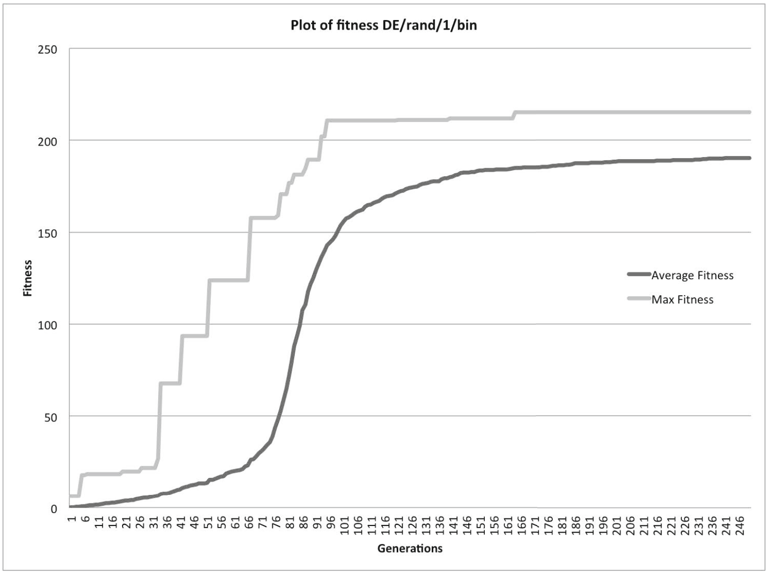


Fig. 3. Maximum and average fitness is plotted across 250 generations using the *DE/rand/1/bin* method

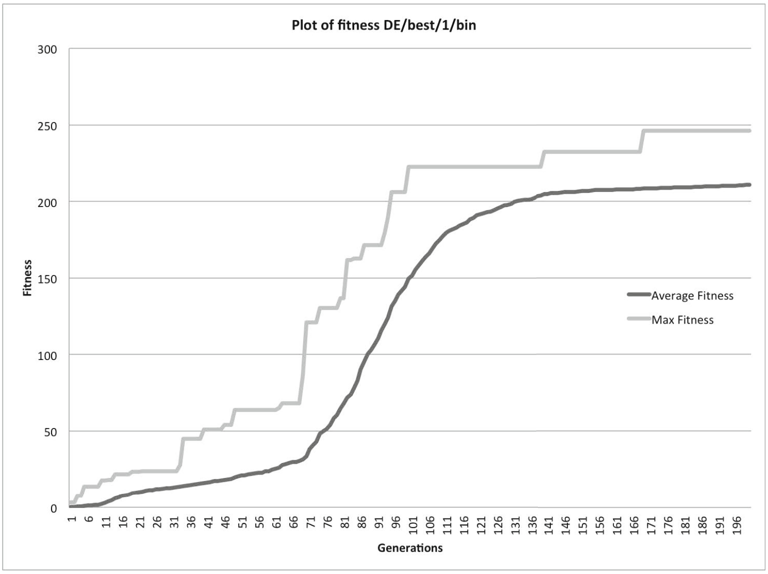


Fig. 4. Maximum and average fitness is plotted across 200 generations using the *DE/best/1/bin* method

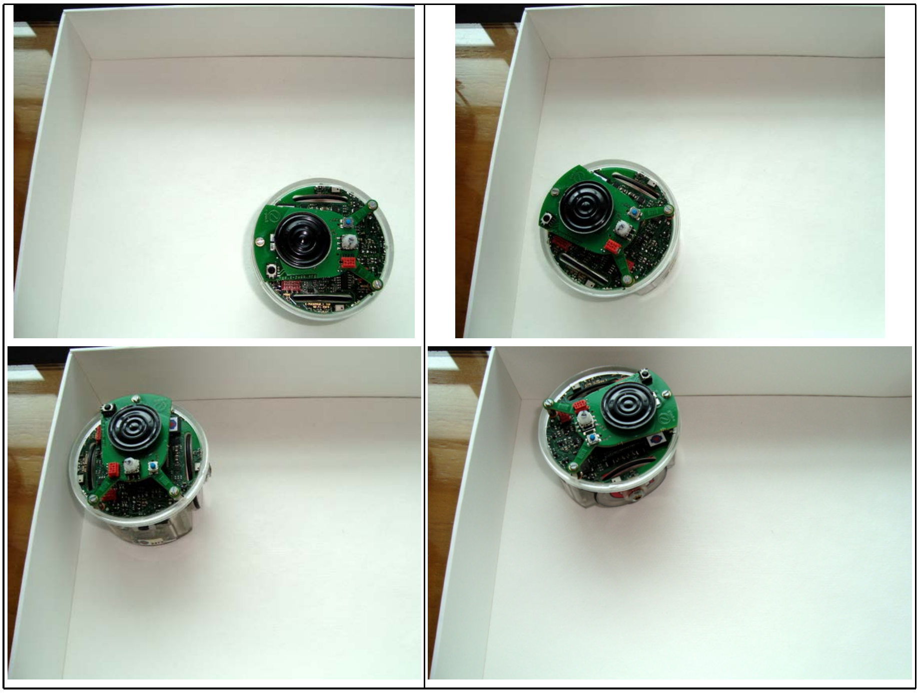


Fig. 5. Robot behavior in the real e-puck robot for the best individual of evolution using the *DE/best/1/bin* method. The robot starts searching for a wall, when a wall is found, the epuck follows the wall while avoids crashing into walls.

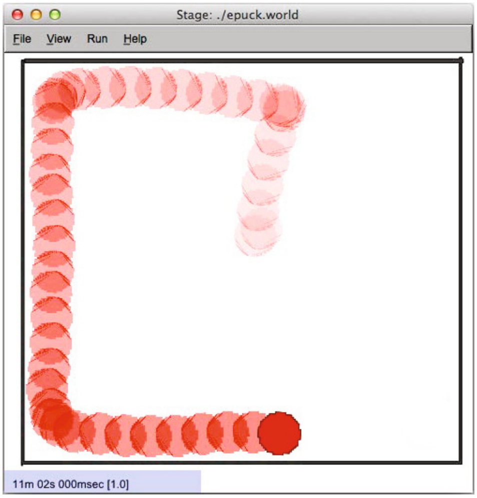


Fig. 6. The Player-Stage simulated e-puck robot for the best individual of evolution using the *DE/best/1/bin* method. A wall-following behavior is exhibited by the best individual.

obtained with the *DE/best/1/bin*, which reaches a maximum fitness of 246.66 (see Fig.4), while the *DE/rand/1/bin* reaches a maximum fitness of 215.217 (see Fig.3). For all cases each individual is evaluated 5 times with random positions and orientations. In general, the *DE/best/1/bin* makes a better exploitation of candidate neural-controllers in the fitness landscape than *DE/rand/1/bin.*

Figures 3 and 4 denote the evolution of the average quality of the population and the evolution of the quality of the best individual. In both cases, the *DE/best/1/bin* and the *DE/rand/1/bin*, the average quality approaches progressively to the best quality through generations, though maintaining an adequate diversity in the population able to obtain better solutions. Therefore, the exploration dominates completely in DE searching.

In total, evolution was carried out for 30,000 evaluations. After evolution is stopped the wall-following behavior can be described as follows: robot is located in the arena, first it starts by looking for the nearest wall, when the wall is found, the e-puck follows the wall while it avoids crashing into walls (see Fig.5 and Fig.6).

6 Discussion and Conclusions

In ER, the most common approach relies on the use of GA and NNs for evolving robot controllers. In this work we evolve neuro-controllers using two DE variants. The work on this paper shows the results of using DE variants: *DE/rand/1/bin* and *DE/best/1/bin* to evolve a wall-following behavior; using the Player-Stage simulator, which facilitated testing individuals during evolution. The resultant behavior is equivalent for both the real and the simulated e-puck robot. Here, we have shown that the use of DE produces fine candidate neuro-controller solutions for robot behavior. DE offers easy parameter calibration, fast implementation, fine exploration/exploitation of candidate solutions in the fitness landscape, and also production of mature individuals during the initial generations of the evolutionary process.

As future work, we propose to combine or hybridize DE with local search methods, like supervised and reinforcement learning in the neural network controllers. This will allow the integration of the advantages of the two search methods: the global search provided by the DE method with the fast local search of the neural learning methods.

Acknowledgments. This work has been sponsored by CONACyT-MEXICO grant SEP No. 0100895 and by the Ministry of Science and Innovation of Spain (project TIN2011-27294). The third author acknowledges support from CONACyT-MEXICO through project No. 79809.

References

1. Nolfi, S., Floreano, D.: Evolutionary Robotics, The Biology, Intelligence, and Technology of Self-Organizing Machines. The MIT Press, Cambridge (2000)
2. Cliff, D.T., Harvey, I., Husbands, P.: Explorations in Evolutionary Robotics. Adaptive Behavior 2, 73–110 (1993)

3. Holland, J.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
4. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading (1989)
5. Trefzer, M.A., Kuyucu, T., Miller, J.F., Tyrrell, A.M.: Evolution and Analysis of a Robot Controller Based on a Gene Regulatory Network. In: Tempesti, G., Tyrrell, A.M., Miller, J.F. (eds.) ICES 2010. LNCS, vol. 6274, pp. 61–72. Springer, Heidelberg (2010)
6. Nolfi, S., Floreano, D., Miglino, O., Mondada, F.: How to Evolve Autonomous Robots: Different Approaches in Evolutionary Robotics. In: Brooks, R.A., Maes, P. (eds.) Proceedings of the IV International Workshop on Artificial Life, pp. 190–197. MIT Press, Cambridge (1994)
7. Mondada, F., Floreano, D.: Evolution of Neural Control Structures: Some Experiments on Mobile Robots. J. Robotics and Autonomous Systems 16, 183–195 (1995)
8. Floreano, D., Mondada, F.: Evolution of homing navigation in a real mobile robot. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 26, 396–407 (1996)
9. Nolfi, S.: Evolving non-Trivial Behaviors on Real Robots: a garbage collecting robot. Robotics and Autonomous Systems 22, 187–198 (1997)
10. Nolfi, S.: Evolving non-trivial behavior on autonomous robots: Adaptation is more powerful than decomposition and integration. In: Gomi, T. (ed.) Evolutionary Robotics, pp. 21–48. AAI Books, Ontario (1997)
11. Kodjabachian, J., Meyer, J., Ecole, A.: Superieure France.: Evolution and development of neural controllers for locomotion, gradient-following, and obstacle-avoidance in artificial insects. J. Trans. Neur. Netw., 796–812 (1997)
12. Quinn, M., Smith, L., Mayley, G., Husbands, P.: Evolving team behavior for real robots. In: EPSRC/BBSRC International Workshop on Biologically-Inspired Robotics: The Legacy of W. Grey Walter (WGW 2002), August 14-16 (2002)
13. Nelson, A.L., Grant, E., Galeotti, J.M., Rhody, S.: Maze exploration behaviors using an integrated evolutionary robotics environment. J. Robotics and Autonomous Systems 46, 159–173 (2004)
14. Montes-Gonzalez, F., Aldana-Franco, F.: The Evolution of Signal Communication for the e-puck Robot. In: Batyrshin, I., Sidorov, G. (eds.) MICAI 2011, Part I. LNCS, vol. 7094, pp. 466–477. Springer, Heidelberg (2011)
15. Rechenberg, I.: Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. Dr.-Ing. Thesis, Technical University of Berlin, Department of Process Engineering (1973)
16. Storn, R., Price, K.: Differential Evolution a Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. J. of Global Optimization 11(4), 341–359 (1997)
17. Koza, J.R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press (1992)
18. Christopher, L., Huosheng, H.: Using Genetic Programming to Evolve Robot Behaviours. In: Proceedings of the 3rd British Conference on Autonomous Mobile Robotics & Autonomous Systems (2001)
19. Mondana, F., Bonani, M.: The e-puck education robot, http://www.e-puck.org/
20. Doncieux, S., Mouret, J.: Behavioral diversity measures for Evolutionary Robotics. In: IEEE Congress on Evolutionary Computation, pp. 1–8 (2010)
21. Santos, J., Duro, R.: Artificial Evolution and Autonomous Robotics. Ra–Ma Editorial, Spain (2004) (in Spanish)

22. Slowik, A., Bialko, M.: Training of artificial neural networks using differential evolution algorithm. In: Proc. IEEE Conf. Human Syst. Interaction, Cracow, Poland, pp. 60–65 (2008)
23. Price, K.: An introduction to differential evolution. In: Corne, D., Dorigo, M., Glover, F., Dasgupta, D., Moscato, P., Poli, R., Price, K.V. (eds.) New Ideas in Optimization, pp. 79–108. McGraw-Hill Ltd, Maidenhead (1999)
24. Moreno, L., Garrido, S., Martín, F., Munoz, M.L.: Differential evolution approach to the grid-based localization and mapping problem. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007), pp. 3479–3484 (2007)
25. Vahdat, A.R., NourAshrafoddin, N., Ghidary, S.S.: Mobile robot global localization using differential evolution and particle swarm optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), pp. 1527–1534 (2007)
26. Joshi, R., Sanderson, A.C.: Minimal representation multisensor fusion using differential evolution. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 29(1), 63–76 (1999)
27. Feoktistov, V.: Differential Evolution: In Search of Solutions. Springer (2006)
28. The Player Project. Free Software tools for robot and sensor applications, `http://playerstage.sourceforge.net/`
29. Mezura-Montes, E., Miranda-Varela, M.E., Gómez-Ramón, R.C.: Differential evolution in constrained numerical optimization: An empirical study. J. Information Sciences 180(22), 4223–4262 (2010)

Robust Visual Localization of a Humanoid Robot in a Symmetric Space

Mauricio J. García Vazquez, Jorge Francisco Madrigal, Oscar Mar, Claudia Esteves, and Jean-Bernard Hayet

Centro de Investigación en Matemáticas (CIMAT)
Guanajuato, Gto., México
{mgarcia,pacomd,oscarmar,cesteves,jbhayet}@cimat.mx

Abstract. Solving the global localization problem for a humanoid robot in a fully symmetric environment, such as the soccer field of the RoboCup games under the most recent rules, may be a difficult task. It requires to maintain the robot position distribution multi-modality whenever it is necessary, for ensuring that the correct position is among the distribution modes. We describe a three-level approach for handling this problem, where (1) a particle filter is run to implement the Bayes filter and integrate elegantly our probabilistic knowledge about the visual observations and the one on the robot motion, (2) a mode management system maintains explicitly the distribution modes and allows to guarantee the satisfaction of constraints such as unresolved symmetry, and (3) a discrete state machine over the modes is used to determine the most pertinent observation models. We present very promising results of our strategy both in simulated environments and in real configurations.

1 Introduction

Humanoid robotics has undergone a huge development in the last decade, following the success of pioneering projects such as the Asimo robot, reaching a point where the mechatronics design of stable biped platforms is not really the main problem anymore. Moreover, low-cost humanoid platforms such as the Nao robot have become affordable for many institutions in the world, permitting several research teams around the world to focus on enhancing these systems with algorithms giving them some form of autonomy. Although many of the main bricks ensuring autonomy, e.g. localization, map making..., have already been extensively studied for wheeled robots, humanoid robots have some specific features making a straightforward application of these existing algorithms somewhat difficult. For example, humanoid robots generally rely *only* on vision for their perception, which implies in particular that (1) one should solve properly the problem of the camera localization with six degrees of freedom, or at least use partial information from images in a clever way; (2) one should handle phenomena such as motion blur due to motion jerk, which are common with humanoid robots; (3) one can process images in reasonable times, giving the limited embedded computing resources of humanoid robots. Such a vision-based

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 227–238, 2013.

localization system is the object of this work, and our contribution lies in the definition and evaluation of three-layered Monte-Carlo (MC) particle filter-based localization system capable of handling symmetric environments.

2 Related Work and Overview of Our Proposal

We highlight in this section a few iconic works in humanoid robot localization and give a sketch on our strategy for robust localization in the Robocup context.

2.1 Related Work

Visual localization of humanoids has been tackled in a number of manners, both metric or non-metric. Approaches such as [1] rely on appearance-based techniques to localize the robot not in a metric map, but in an implicit *qualitative* map, relatively to some reference sequence of images taken in a learning phase, and to which are associated very simple actions in a world made of corridors. Among methods based on metric maps, some of them are purely geometrical, such as [2], where geometric algebras are used to express the localization problem as an intersection of Gaussian spheres.

However, more commonly, localization is based on probabilistic filters, i.e. on the recursive application of the Bayes rule. They combine probabilistic motion models (odometry…) and measurement models, e.g. based on stereo-vision [3]. In the aforementioned work, the estimation is done for the robot position in the plane (position and orientation), and the internal odometry is used to get the relative position of the camera with respect to the Zero Moment Point, for which the motion is given, and a depth map for that frame. It provides the probabilistic measurement model, by comparing measured and expected depth profiles.

However, in most cases of humanoid robots, *monocular* vision is the only reliable sensor to be used for localization. In that case, measurements live in a lower space, the image space, and measurements consists in image features (interest points, edges,…) associated to robot poses. For example, in [4], these features are the projection of a polyhedrical model, and observations likelihoods reflect how much the observed projections are likely, given the 3D model. This process may not seem difficult from the theoretical computer vision perspective, but specific limitations of humanoid robots (camera not precisely located w.r.t. the robot base, blurring effects) may make it more difficult than thought.

Particularly illustrative of the practical problems of humanoid monocular model-based localization is the Robocup context [5–7]: Robots are localized on a soccer field thanks to several known features, such as the white lines, the central circle, the goal posts. However, there are ambiguities (white lines are the same everywhere). Moreover, the robot swinging causes strong blurs, sometimes. Another strong limitation is that the camera has a narrow field-of-view. One of the most effective approaches is the well-known Monte-Carlo Localization, based on particle filters. At each iteration of the algorithm, a set of weighted samples over the posterior distribution on the state to estimate is maintained.

It is updated by sampling new positions according to a probabilistic robot motion model, e.g., from odometry [7], and by modifying the samples weights based on the likelihoods of detected features, given the positions estimates. The main advantage of such an approach is that it solves well the *global* localization, when ambiguities make the posterior intrinsically multimodal. It has also been adapted to *active* localization strategies [8], where for example the best action is taken at a one-step horizon in order to maximize the decrease in the distribution entropy. In the most recent version of the Robocup soccer challenges, handling multimodality on the pose distribution has become critical, since the map by itself is completely symmetrical. Our system has been developed with this context in mind, and we describe it briefly in the following.

2.2 Overall Description

Our proposal is based on a three-layer scheme described in Fig. 1. At the lower level, a particle filter (Section 3) is in charge of the estimation of the robot position; the particularity of this filter is that it has an explicit handling of the multi-modality in the distribution. At the intermediate level, we manage the existence of mixture components (Section 4.1); this implies for example creating new mixtures when the variance of one mixture grows too much, or when we need to reinforce symmetry. At the higher level, an explicit representation of the current state of the localization is maintaine (e.g., "Lost", "Bi-modal", "Mono-modal"). It conditions the way the other two levels work (Section 4.2). Our contributions are the following:

- we propose a mixture-based particle filter with a mixture management oriented to the problem of localization in a symmetric environment;
- we propose to handle explicitly the multi-modality of the particle distribution linked to symmetry by means of the mixture;
- we propose a per-observation model for detecting cases of localization failures requiring "rescue" sampling schemes;
- we conducted extensive experiments of the proposed system in simulations and onboard of a small-sized humanoid robot in the Robocup context.

3 Mixture-Based Particle Filter

3.1 The Classical Particle Filter

In this work, we use as a basis filter a variant of the Monte-Carlo Localization (MCL) algorithm for humanoid robots, so that we first describe in its great lines the typical MCL algorithm, that is also explained for example in details in [9]. MCL uses a set of particles (Monte-Carlo samples) to describe the posterior probability distribution $p(\mathbf{X}_t|\mathbf{U}_{1:t}, \mathbf{Z}_{1:t})$ over the state $\mathbf{X}_t$, given a sequence of controls done by the robot $\mathbf{U}_{1:t} \overset{\text{def}}{=} \mathbf{U}_1, \mathbf{U}_2, \ldots, \mathbf{U}_t$ (measured by internal sensors such as odometry) and a sequence of measurements $\mathbf{Z}_{1:t}$. In our case, $\mathbf{X}_t$ includes

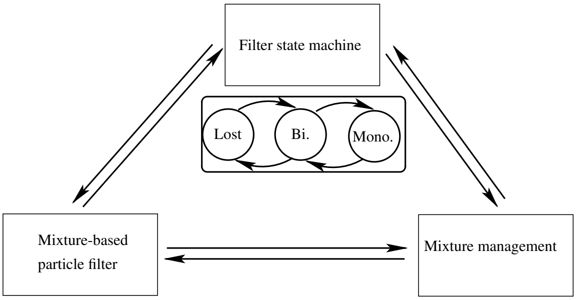


Fig. 1. Overview of our particle filter-based localization system: the localization state machine controls both the behavior of the mixture management system (in charge of reorganizing the mixture components, if necessary), and of the particle filter itself.

the three degrees of freedom of the robot we want to estimate (position and orientation on the plane), even though a humanoid robot has generally many more degrees of freedom. Under the Markov assumption, one can use the recursive Bayesian formulation of the posterior through

$$p(\mathbf{X}_t|\mathbf{U}_{1:t},\mathbf{Z}_{1:t}) \propto p(\mathbf{Z}_t|\mathbf{X}_t)\int_{\mathbf{X}_{t-1}} p(\mathbf{X}_t|\mathbf{U}_t,\mathbf{X}_{t-1})p(\mathbf{X}_{t-1}|\mathbf{U}_{1:t-1},\mathbf{Z}_{1:t-1})d\mathbf{X}_{t-1}.$$

MCL represents the posterior (left term above) as a set of weighted samples $(\mathbf{X}_t^{(n)},\omega_t^{(n)})$, generated at each time step from a proposal distribution q. The algorithm iterates the following :

1. from the previous set of particles, generate a new set from the proposal distribution, i.e. for all n, $\mathbf{X}_t^{(n)} \sim q(\mathbf{X}_t|\mathbf{X}_{1:t-1}^{(n)},\mathbf{U}_{1:t},\mathbf{Z}_{1:t})$,
2. update and normalize particle weights with the particle state likelihood, the motion model and the proposal, i.e. for all n

$$\tilde{\omega}_t^{(n)} = \frac{p(\mathbf{Z}_t|\mathbf{X}_t^{(n)})p(\mathbf{X}_t^{(n)}|\mathbf{X}_{t-1}^{(n)})}{q(\mathbf{X}_t^{(n)}|\mathbf{X}_{1:t-1}^{(n)},\mathbf{Z}_{1:t})}\omega_{t-1}^{(n)},\quad \omega_t^{(n)} = \frac{\tilde{\omega}_t^{(n)}}{\sum_m \tilde{\omega}_t^{(m)}},$$

   where $\tilde{\omega}_t^{(n)}$ are the un-normalized weights;
3. compute the average state; compute the number of effective particles, approximately $\frac{1}{\sum_n [\omega_t^{(n)}]^2}$; if it is inferior to some threshold, proceed to resampling of the particles with replacement.

Here, the probabilistic motion model $p(\mathbf{X}_t|\mathbf{U}_t,\mathbf{X}_{t-1})$ is a bit particular, as a humanoid robot does not have such things as optical encoders to measure the displacement done, like in wheeled robots. What we can get are the footprints computed by the planning algorithm for any motion request done to the robot, and the values of the angles of the robot joints, from which a "form of odometry" is available, and will be used here in the probabilistic motion model.

3.2 Particle Filter with Explicit Mode Management

One of the main problems of the basic particle filter described above is that, because of its global resampling step, it is possible that, even if the belief over the robot position has a multi-modal nature, some of the modes disappear. In the case of a robot evolving in a symmetrical environment, this could cause catastrophic failure of the localization system. A very interesting proposal has been presented in [10] to partially overcome this problem. The idea is to represent the aforementioned posterior as a weighted sum of distributions,

$$p(\mathbf{X}_t|\mathbf{U}_{1:t}, \mathbf{Z}_{1:t}) = \sum_{k=1}^{M_t} \alpha_t^{(k)} p^{(k)}(\mathbf{X}_t|\mathbf{U}_{1:t}, \mathbf{Z}_{1:t})$$

where M_t is the (variable) number of distributions in the mixture. We represent all the distributions $p^{(k)}(\mathbf{X}_t|\mathbf{U}_{1:t}, \mathbf{Z}_{1:t})$ of the mixture as a particle distribution, i.e. sets of $N_t^{(k)}$ weighted particles $\{\mathbf{X}_t^{(n,k)}, \omega_t^{(n,k)}\}_{1\leq n\leq N_k}$. What has been shown in [10] is that, remarkably, the application of the prediction-correction steps of the classical particle filter to the mixture is quite straightforward:

- the particles in each part of the mixture are updated in a similar way as in the classical particle filter: the sampling, correction, and resampling steps are done locally to each mixture component; the particles weights are normalized on a per-component basis; note that the data association is also done on a per-component basis (more details will be given in Section 5.2);
- the weights of each component are updated in function of the un-normalized particle weights (i.e. the evaluated likelihoods over the particles)

$$\alpha_t^{(k)} \approx \frac{\alpha_{t-1}^{(k)} \sum\limits_{n=1}^{N_t^{(k)}} \tilde{\omega}_t^{(n,k)}}{\sum\limits_{k=1}^{M_t} \alpha_{t-1}^{(k)} \sum\limits_{n=1}^{N_t^{(k)}} \tilde{\omega}_t^{(n,k)}}, \tag{1}$$

  where $\alpha_t^{(k)}$ are components weights, and $\tilde{\omega}_t^{(n,k)}$ are particles weights.

The management of the modes is independent of the mechanics of the particle filter, and in the particular case of the localization on a soccer field, we will see in Section 4.1 a few principles to handle the creation, fusion, deletion of mixture components. Computationally, it is important to highlight that, besides the weights update of Eq. 1, the involved operations are the same as in the classical particle filter, and that if $N \overset{\text{def}}{=} \sum\limits_{k=1}^{M_t} N_t^{(k)}$, the complexity of the particle filter algorithm is linear in N.

4 Mixture Management and Filter State Machine

4.1 Mixture Management

The mixture components described above evolve according to a few atomic operations aimed at improving the representation of the posterior distribution.

Mixture Components Formation. At initialization, components are created by clustering by a simple variant of expectation-maximization.

Mixture Component Fusion. This operation simply consists in fusing two existing components whenever their means are close enough. By using a metric d on the state space of positions, and defining a threshold δ on this metric, a fusion is done between two components k and l whenever

$$d(\bar{\mathbf{X}}^{(k)} - \bar{\mathbf{X}}^{(l)}) < \delta,$$

where $\bar{\mathbf{X}}^{(k)}$ is the mean of component k. The fusion consists in reassigning all the $N^{(k)}$ and $N^{(l)}$ particles under the same components, with $N^{(k)} + N^{(l)}$ particles. The new component weight is the sum $\alpha_t^{(k)} + \alpha_t^{(l)}$.

Mixture Component Deletion. When the weight of a given component k is repeatedly reported as below a threshold (i.e. the likelihoods corresponding to the particles assigned to this component are low), for a given number of localization cycles, then it is deleted. Deletion consists in (1) sharing uniformly the weight of the deleted component to the other components and (2) distributing the $N^{(k)}$ particles among the other components. For this second step, for each particle m among the $N^{(k)}$ particles of component k, we select randomly a component $j \neq k$ and a particle n from j to replace the state of the particle from k:

$$(\mathbf{X}_t^{(m,k)}, \omega_t^{(m,k)}), (\mathbf{X}_t^{(n,j)}, \omega_t^{(n,j)}) \to (\mathbf{X}_t^{(n,j)}, \frac{1}{2}\omega_t^{(n,j)}), (\mathbf{X}_t^{(n,j)}, \frac{1}{2}\omega_t^{(n,j)}),$$

that preserves the consistency of the particle representation in component j.

Rescue Component Generation. When the level of measured likelihoods is poor for some observation model compared to its "usual" value, or when components are not associated to a persistent observation for a while, we create a rescue component r. The quite simple and well known idea [9] consists in generating random samples along the manifold corresponding to the inverse of the current observation. For example, if a single pole is observed, particles are generated along an arc of circle with a radius associated to the apparent diameter of the pole in the image. In our component-based filter, this is implemented by (1) computing the number $N^{(r)}$ of rescue samples to generate, (2) on a particle basis, we select an existing component k and a particle of this component n, that we replace by the sample from the rescue distribution. The component weight is initialized to a constant value (which is also subtracted to the other components), and the particle weights are uniform, i.e.

$$(\mathbf{X}_t^{(n,k)}, \omega_t^{(n,k)}) \rightarrow (\mathbf{X}_t^{(r,j)}, \frac{1}{N^{(r)}}).$$

Symmetrization. When the belief over the position is by essence multi-modal (i.e. at one of the multi-modal states described in the following section), we may have to reinforce the symmetry of the distribution. The process is quite straightforward: for all components, we check that a close-to-symmetrical component exists in the current mixture; if not, we create a new component with the same process as the one described above for rescue component generation.

4.2 Upper Level Control

At the highest level, a state machine (see Fig 1) controls the behavior of the particle filter and of the mixture components management. In our current implementation, we count on three states: 'Lost'[1], 'Bi-modal' and 'Mono-modal'. At initialization, the state is 'Lost', then transitions are ruled by the following:

- from state 'Lost' to state 'Bi-modal', the components should be two, and close to symmetrical;
- from state 'Bi-modal' to state 'Mono-modal', an external process must intervene to raise the ambiguity. More specifically, this could mean that a place/object recognition process has been able to decide which of the symmetrical map element has produced an ambiguous observation;
- the degradation from 'Bi-modal' or 'Mono-modal' to 'Lost' may occur when the covariance and quality of the single component reaches too high values (in which case the clustering process is run again), or when rescue components are created in the mixture.

Moreover, the localization state conditions several aspects of the filter: (1) all observation models are not necessarily used, (2) the recognition process that disambiguates the field symmetry is used only at state 'Bi-modal', (3) the symmetrization component generation explained above is only used in state 'Lost' and 'Bi-modal'. The following table sums up these properties.

State	Observation models (see Section 5.1)	Recognition process	Simmetrization (see Section 4.1)
'Lost'	(1)	×	✓
'Bi-modal'	(1,2,3)	✓	✓
'Mono-modal'	(1,2,3)	×	×

[1] 'Lost' is a somewhat abusive word, as in reality the distribution is simply multimodal.

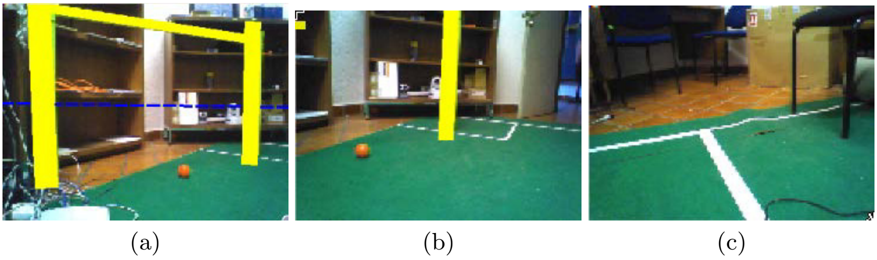


Fig. 2. The three observation models used in our system: (a) goal post model, made of the image position of the projected goal post; note that a unique position of the robot can be derived from a single observation; (b) single goal pole model, made of a single projected goal pole; (c) line cross model, made of the geometric properties of white line crosses. The last two models are not invertible.

5 Application to the Localization on the Robocup Field

In this section, we describe a few elements specific to the application that motivated this work. In particular, we describe briefly the probabilistic observation models, the data association strategy, and the rescue schemes associated to each observation model.

5.1 Observation Models

Our localization system relies on the use of three heterogeneous observation models, driven by the basic image processing functionalities coded on our robot:

1. the observation of goal posts, made by one or two vertical poles and an horizontal pole (see Fig. 2(a)) extracted by our visual segmentation system; the resulting observation is a 5×1 vector including the mean position of the observed vertices, the post width and height, and the angle of the ray pointing through the intersection of the horizontal post with the horizon line (in dashed blue in Fig. 2(a));
2. the observation of single poles (see Fig. 2(b)), extracted in the same way as the goal posts; the resulting observation is a 3×1 vector holding the mean pole position and its width in the image;
3. the observation of line crosses (see Fig. 2(c)), extracted by a simple line segmentation; the resulting observation is a 5×1 vector holding the point position, the angles of the lines forming the crossing and an indicator on the kind of crossing ('X', 'L' or 'T' crossing).

For all of these models, the noise distribution in $p(\mathbf{Z}_t|\mathbf{X}_t)$ is chosen as Gaussian, with variance parameters determined empirically. For efficiency reasons, if at one time instant t, several observations are available from the image processing system (in Fig. 3, for example) then only one is used. We used several heuristics for the underlying selection, one being that some observation models are more useful than others at some state, as described in Section 4.2. Also, when several observation corresponding to the same model (e.g., only line crosses) are available, the least ambiguous is selected (i.e. the most different to the others).

5.2 Data Association

The data association is performed on a component basis. This is another benefit of using a mixture-based representation. Instead of performing the data association based on the only filter mean (fast but not robust, since modes could be lost) or based on each particle (very robust but extremely slow), we do it on an intermediate level, i.e. the mixture component level, which gives us the benefits of robustness (by allowing different modes to get their own association) and efficiency (the number of components being typically a one digit number).

Hence, when getting some observation $\mathbf{Y}_t$ that could be coming from Q map elements $q = 1..Q$, we associate the observation to the map element $\hat{q}$ maximizing the likelihood at the center of each component k:

$$\hat{q} = \begin{cases} \arg\max_q p(\mathbf{Z}_t|\bar{\mathbf{X}}^{(k)}, q) \text{ if } \max_q p(\mathbf{Z}_t|\bar{\mathbf{X}}^{(k)}, q) > \epsilon \\ -1 \text{ otherwise.} \end{cases}$$

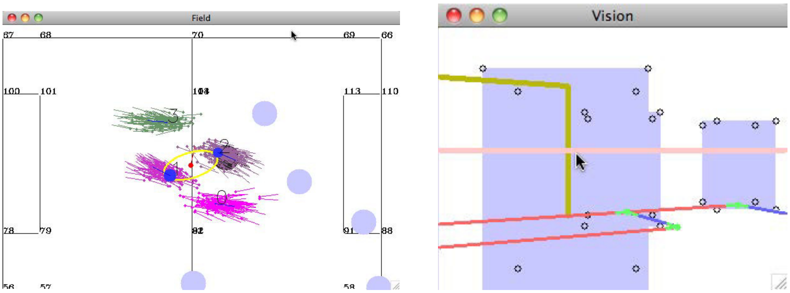
The -1 means that the best likelihood being not confident enough, no association is performed and that the observation is not integrated in the filter.

5.3 Rescue Distributions

The last ingredient of the mixture-based particle filter we implemented for localization purposes is the rescue process. The idea is to detect as soon as possible, from observations collected from the image processing modules, that the quality of the estimation is degrading, We define two indicators for assessing this quality:

- for each observation model o defined in Section 5.1, an indicator is defined that measures how far is the current average likelihood among associated components and the long-term average likelihood; This is an extension of the approach proposed in [9] to the case where several observation models are managed simultaneously. In our system, three indicators are kept, for each observation model, in the form $\frac{a_t^{o,s}}{a_t^{o,l}}$, where $a^{o,s}$ is defined as $a^{o,s} = \frac{1}{T(t_s,t)} \sum_{\tau \in [t_s,t]} \frac{1}{N_\tau^o} \sum_{n,k} \alpha_\tau^{(k)} \tilde{\omega}_\tau^{(n,k)}$. N_τ^o is the number of particles that have been evaluated under the observation model o at time τ,
- a counter of the number of times no component have had observations associated to them is also maintained.

When any of these indicators is activated, a rescue distribution is used to generate samples in a new component, as mentioned above. It is not described in details here for lack of space. See for example [9]. The principle is trivial : for the current observation (goal post, pole, line crosses), we generate a sampling



(a) (b)

Fig. 3. The 2D simulation ambient developed for simulation results. In (a), a bird's eye view of the map we used (with visible mixture components, one per color). (b) The robot subjective view. In blue, we simulate obstacles on the field.

distribution that corresponds to the inverse of the observation. For the goal post model, this means a distribution around two mean positions (computed deterministically from the goal observation).

6 Experiments: Simulations and Port to the Nao Robot

Our localization system has been implemented in C++, in a library that has been used both to generate simulation results (see Fig. 3) and a module running in the NAOQI architecture for NAO robots, the small humanoids used in the Standard Platform League of the ROBOCUP.

Our first series of experiments has been performed in a simulation environment built in C++. We simulate one or several 2D, differential drive robots moving on the plane. They are equipped with a visual system capable of detecting entire goal posts, single poles, or line crosses. The three observation models described in the previous section have been simulated to provide our localization system with observation data. We conducted the following experiment: make the robot follow a circular trajectory with some parts without visible observations. In Fig. 4, we measure the average error of our closest component to the ground truth, in the case of admitting a maximal number of components of 1 (i.e., a classical particle filter, in red), 2 (green), 4 (blue). It can be clearly seen that our filter has one of its mode tracking the exact position, whereas the classical approach loses it half of the time (hence the error of about 1 meter, as the two symmetric modes are distant by 2 meters). Moreover, in this particular experiment, because of the absence of critical parts where the robot could really get lost, the system running with 2 modes at maximum performs better (but to allow rescue components, we would need a higher maximal number of modes).

As commented above, the particle filter-based localization has been also ported to the Nao platform, as a module of the Naoqi framework. The way the filter works is basically the same, except that the observations are real (from an image processing module extracting yellow poles, white lines. . .) and that the motion

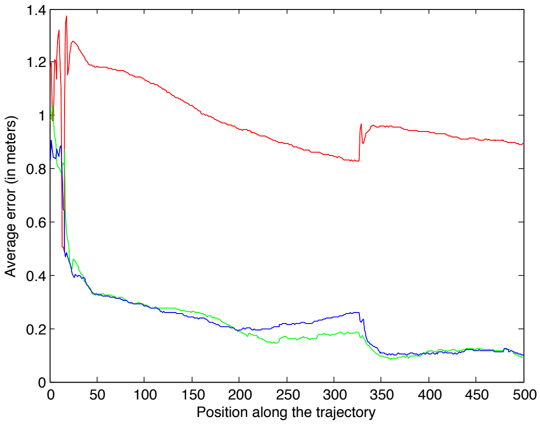


Fig. 4. Evolution of the estimation error on the trajectory in a perfectly symmetric environment, measured by the closest mode to the ground truth. In red, evaluation of a classical particle filter; in green and blue, our improved mixture-based particle filter.

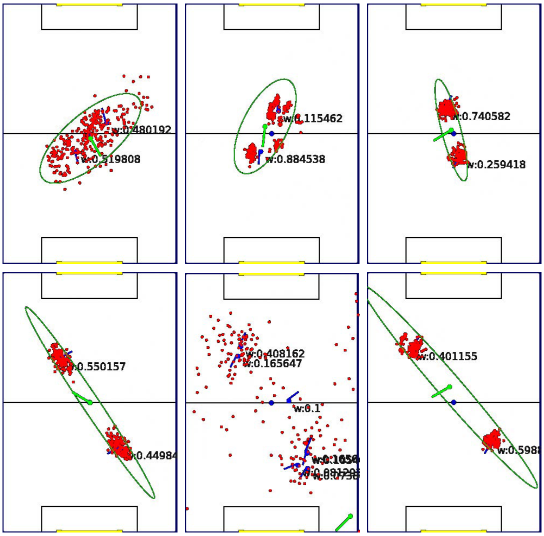


Fig. 5. Visualization of steps 162, 166, 172, 238, 307, 353 of the Nao localization in a symmetric soccer field. Intentionally, we disabled the goal identification system to show the explicit handling of multi-modality. The numbers are the mode weights $\alpha^{(k)}$.

model is derived from the robot odometry. A few snapshots of our localization system is visible in Fig. 5. It corresponds to a similar trajectory as in simulation case, except that much fewer observations are handled (line crosses are not easily detectable). As it can be seen over these steps we included, the robot is initially not localized, at frame 162, but with a gross estimate. After having started its path, it is progressively located, with two symmetric modes (frame 172). When, because if the absence of observations, the localization quality degrades, several new modes appear and the variance (in green) grows considerably. After seeing again a goal post (frame 353), modes get removed and the robot gets its final two symmetric components again.

7 Conclusions and Future Work

We have introduced a mixture-based particle filter for the localization of a humanoid robot evolving in symmetric ambients such as the soccer field in the Robocup soccer challenge. The main contribution of this work is a localization system handling explicitly the management of mixture components, that, because of the clustering technique generating components, correspond generally to modes of the posterior distribution. We defined a set of basic operations based on these components to reinforce the handling of the field symmetry, the situations of getting lost, and the integration of disambiguation information (leading to a mono-modal state). The complexity of our filter stays linear in the number of particles, although slightly higher than the classical approach, because of the component management process. In a near future, we will combine this approach with active strategies for selecting the best observation at a given instant.

References

1. Ido, J., Shimizu, Y., Matsumoto, Y., Ogasawara, T.: Indoor navigation for a humanoid robot using a view sequence. Int. Journal of Robotics Research 28, 315–325 (2009)
2. Gonzalez-Aguirre, D., Asfour, T., Bayro-Corrochano, E., Dillmann, R.: Model-Based Visual Self-Localization Using Gaussian Spheres. In: Geometric Algebra Computing: In Engineering and Computer Science, p. 299. Springer (2010)
3. Thompson, S., Kagami, S.: Humanoid robot localisation using stereo vision. In: Proc. of the IEEE/RAS Int. Conf. on Humanoid Robots, pp. 19–25 (2005)
4. Michel, P., Chestnutt, J., Kagami, S., Nishiwaki, K., Kuffner, J., Kanade, T.: Gpu-accelerated real-time 3d tracking for humanoid autonomy. In: Proc. of the JSME Robotics and Mechatronics Conference (2008)
5. Laue, T., Haas, T.D., Burchardt, A., Graf, C., Röfer, T., Härtl, A., Rieskamp, A.: Efficient and reliable sensor models for humanoid soccer robot self-localization. In: Proc. of the Workshop on Humanoid Soccer Robots, Humanoids 2009 (2009)
6. Strasdat, H., Bennewitz, M., Behnke, S.: Multi-cue Localization for Soccer Playing Humanoid Robots. In: Lakemeyer, G., Sklar, E., Sorrenti, D.G., Takahashi, T. (eds.) RoboCup 2006: LNCS (LNAI), vol. 4434, pp. 245–257. Springer, Heidelberg (2007)
7. Liemhetcharat, S., Coltin, B., Veloso, M.: Vision-based cognition of a humanoid robot in standard platform robot soccer. In: Proc. of Workshop on Humanoid Soccer Robots. In the IEEE-RAS Int. Conf. on Humanoid Robots, pp. 47–52 (2010)
8. Seekircher, A., Laue, T., Röfer, T.: Entropy-Based Active Vision for a Humanoid Soccer Robot. In: Ruiz-del-Solar, J. (ed.) RoboCup 2010. LNCS (LNAI), vol. 6556, pp. 1–12. Springer, Heidelberg (2010)
9. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. MIT Press (2005)
10. Vermaak, J., Doucet, A., Perez, P.: Maintaining multimodality through mixture tracking. In: Proc. of IEEE Conf. on Computer Vision, vol. 2, pp. 1110–1116 (2003)

DeWaLoP In-Pipe Robot Position from Visual Patterns

Luis A. Mateos and Markus Vincze

Automation and Control Institute (ACIN), Vienna University of Technology,
Gusshausstrasse 27 - 29 / E376, A - 1040, Austria
{mateos,vincze}@acin.tuwien.ac.at
http://www.acin.tuwien.ac.at

Abstract. This article presents a methodology to position an in-pipe robot in the center of a pipe from a line matching algorithm applied to the unwrapped omni-directional camera located at the robot's front-end. The advantage of use omni-directional camera inside the pipes is the relation between the cylindrical image obtained from the camera and the position of the camera on the robot inside the pipe, where by direct relation the circular features become linear. The DeWaLoP in-pipe robot objective is to redevelop the cast-iron pipe-joints of the over 100 years old fresh water supply systems of Vienna and Bratislava. In order to redevelop the pipes, the robot uses a rotating mechanism to clean and apply a sealing material to the pipe-joints. This mechanism must be set perfectly in the center of the pipe to work properly. Therefore, it is crucial to set the in-pipe robot in the center of the pipe's horizontal x and y axes.

1 Introduction

The advantage of omni-directional cameras over perspective cameras is the possibility to view 360° on a single image frame. Nevertheless, in order to analyze an omni-directional image it is easier to transform it into a panoramic or perspective image. These types of image are more comprehensible to humans. Besides, image-processing algorithms are mainly designed to work with perspective projection [1].

There are several methods for unwrapping omni-directional image. The most common method used to obtain panoramic images is by directly transforming the polar coordinates into rectangular coordinates [2]. Furthermore, by including the equation of the mirror in the unwrapping process the vertical distortion becomes null [3]. Likewise, the camera must be correctly calibrated to correctly transform the image frames [5]. Moreover, to speed up the process of unwrapping the image in real time, instead of calculating the coordinates at all-time, lookup tables are used to know which pixel from the omni-directional image have a direct correspondence to the panoramic image [1].

Roboticists have been using omni-directional cameras on robot localization, mapping and robot navigation [6][7][8][9]. Omni-directional vision allows the

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 239–248, 2013.

robot to recognize places more easily than with standard perspective cameras [10].

In-pipe robots are commonly used for inspection, in this case, the robot's suspension system sets the robot to maintain its centered position while moving in the pipelines [11] [12]. Other types of in-pipe robots are used for cutting or grinding, these robots are mainly a mobile robot with an attached cutting tool. Nevertheless, these suspension system cannot create a stable structure inside the pipe as required by the DeWaLoP robot.

2 DeWaLoP Robot System

The DeWaLoP Developing Water Loss Prevention robot system objective is to redevelop the cast-iron pipes of the over 100 years old fresh water supply systems of Vienna (3000 km length) and Bratislava (2800 km), by crawling into water canals of about one meter diameter and applying a restoration material to repair the pipe-joints. The DeWaLoP robot is intended to be a low cost robot with high reliability and easiness in use, the robot system includes the conventional inspection of the pipe system, which is carried out using a cable-tethered robot with an onboard video system. An operator remotely controls the movement of the robot systems.

The proposed solution consists of three main subsystems: a control station [13], a mobile robot (similar to a vehicle) and a maintenance unit.

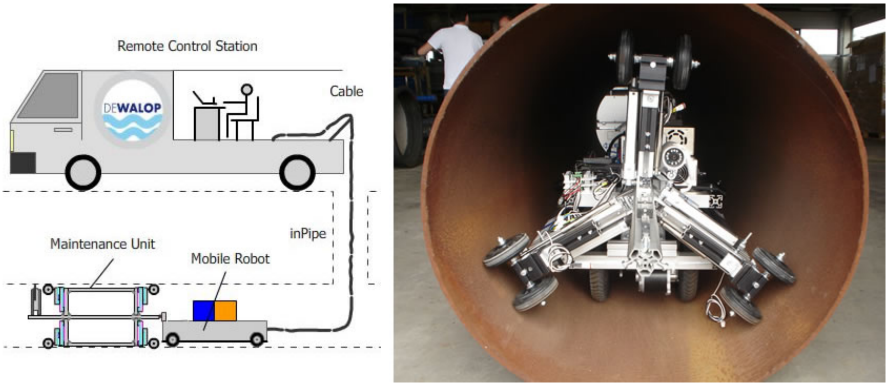


Fig. 1. DeWaLoP in-pipe robot system diagram (on left). DeWaLoP in-pipe robot inside a 900mm diameter pipe

Control Station. The control station is in charge of monitoring and controlling all the systems of the in-pipe robot. The main controller is composed of two processors, a slate computer is in charge of monitoring and displaying the video images from the Ethernet cameras; the second processor is an 8 bits microcontroller with Ethernet capability, in charge of the Robot Systems Remote

Control [14], which receives the information from the physical remote control (the joysticks, buttons and switches), in order to control the robot systems.

Mobile Robot. Mobile platform able to move along the pipe, carrying on board the electronic and mechanical components of the system, such as motor drivers, power supplies, restoration material tank, among others. It uses a differential wheel drive which makes the robot able to promptly adjust its position to remain in the middle of the pipe while moving.

Maintenance Unit. The Maintenance unit is a structure able to expand or compress with a Dynamical Independent Suspension System (DISS) [14]. By expanding its wheeled-legs, it creates a rigid structure inside the pipe, so the robots tools work without any vibration or involuntary movement from its inertia and accurately restore the pipe-joint. By compressing its wheeled-legs, the wheels become active so the maintenance structure is able to move along the pipe by the mobile robot. The structure consists of six wheeled legs, distributed in pairs of three, on each side, separated 120, supporting the structure along the centre of the pipe. The maintenance system combines a wheel-drive-system with a wall-press-system, enabling the system to operate in pipe diameters varying from 800mm to 1000mm. Moreover, the maintenance unit and the mobile robot form a monolithic multi-module robot, which can be easily mounted/dismounted without the need of screws [17].

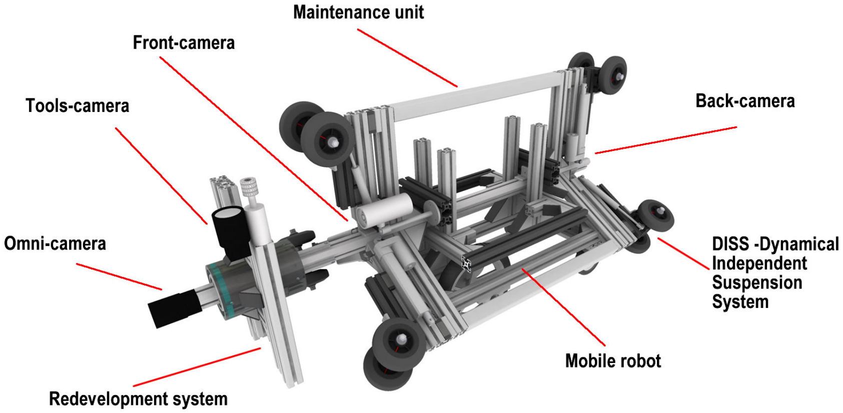


Fig. 2. DeWaLoP in-pipe robot

DeWaLoP Vision System. In order to navigate, detect and redevelop, the in-pipe robot includes in total four cameras [13]. For the navigation stage, two cameras are required, one located at the front, to know the way in the pipe;

the second located at the back, to know the way out. For the detection stage, an omni-directional camera is located at the front-end enabling the pipe-joint detection [14]. Finally, for the redevelopment stage, another camera is mounted on the redevelopment tool system enabling the operator to follow in detail the cleaning and restoration process.

Redevelopment Tool System. The DeWaLoP redevelopment tool system mimics a cylindrical robot able to cover the 3D in-pipe space. Nevertheless, the tool system modifies the standard cylindrical robot into a double cylindrical arm, where both arms are connected to the central axis of the maintenance unit and located 180° each other. Likewise, the cleaning and sealing tools rotate from the center of the pipe. Therefore, it is crucial to set the in-pipe robot perfectly in the 3D center of the pipe.

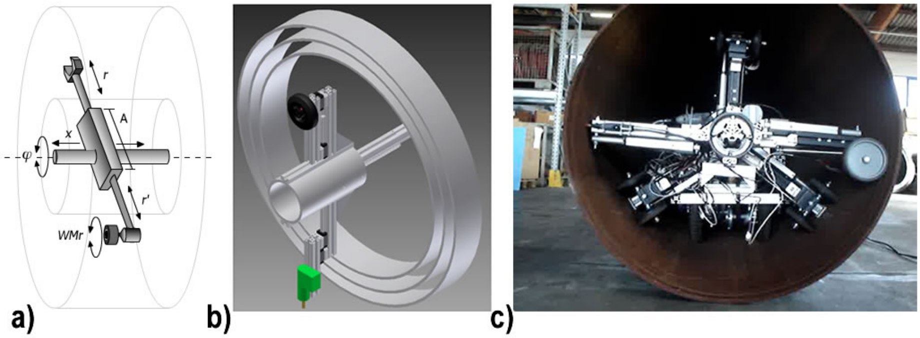


Fig. 3. Redevelopment tools system a) Simplified model. b) 3D model. c) Mounted on the DeWaLoP in-pipe robot.

3 Omni-directional Camera Inside the Pipe

The motivation to use omni-directional camera in the DeWaLoP in-pipe robot is the relation between the cylindrical image obtained from the omni-directional camera and the position of the camera on the robot inside the pipe. The camera is installed on the front end of the maintenance unit, as shown in figure 2, where by direct relation the circular features of the pipe-joints become linear.

The robots omni-directional system consists of a perspective camera adapted to a hyperbolic mirror. It is known that the distribution of resolution from hyperbolic mirrors is better than the one obtained with systems that use spherical mirror or parabolic mirrors without the orthographic lens [18]. The composition of the perspective camera with the hyperbolic mirror is set, so that the camera projection center C coincide with the focal point of the mirror F, enabling the perspective camera to be modeled by an internal camera calibration matrix K, which relates 3D coordinates $X = [x, y, z]^T$ into retinal coordinates $q = [q_u, q_v, 1]^T$.

The azimuthal angle in the original image is mapped into the horizontal axis of the panoramic image, and the radial coordinate in the original image is mapped into the vertical axis of the panoramic image.

The panoramic image is created by doing an inverse mapping of the pixels, for each pixel in the panoramic image a correspondence pixel must exist in the omni-directional camera image. An improved version of this method can be done by including the mirror equation in the unwrapping process, so the vertical distortion is null.

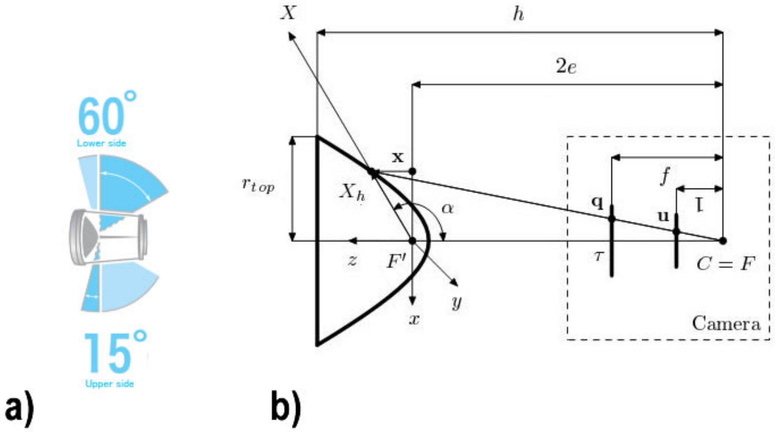


Fig. 4. a) Omni-directional field of view. b) Omni-directional camera model.

Where v_{pn} represents the coordinate of a pixel in a column of the panoramic image and $v_{pn} = 0$ for pixels that correspond to the mirror rim. The parameter r_i represents the radial distance of a pixel in the omni-directional image. The coordinate (x, y) represents a point on the mirror surface. The user defined parameter DV_{pn} represents the height in the panoramic image measured in pixels from the horizontal line that corresponds to the mirror rim, $x = r_{top}$, until the horizontal line that corresponds to $y = 0$. The remaining parameters, mirror eccentricity $2e$, r_{pixel}, y_{top} and r_{top}, are known.

In the same way u_{pn} represents the column in the panoramic image that is being mapped; r_i is the radial distance of a pixel in the omni- directional image that corresponds to a pixel in the panoramic image with vertical coordinate v_{pn}; the parameter u and v represent the coordinate of a pixel in the omni-directional image that corresponds to the pixel with coordinates (u_{pn}, v_{pn}) in the panoramic image.

$$u = r_i cos\left(\frac{2\pi \cdot u_{pn}}{H_{pn}}\right) \tag{1}$$

$$v = r_i sin\left(\frac{2\pi \cdot u_{pn}}{H_{pn}}\right) \tag{2}$$

In order to speed up the process of unwrapping the image in real time, instead of calculating the coordinates at all-time, lookup tables are used to know which pixel from the omni- directional image have a direct correspondence to the panoramic image.

Unwrapping Omni-directional Image from Equation 1 and 2, image (1024x776)

```
function [mtrix_x,mtrix_y]=trans_matrix_base(img_omni,xc,yc,r,R);
r1 = (R+r)/2;
for x1=1:fix((2*pi*r1))
theta=(2*pi*r1-x1)/r1;
for y1=1:R;
xomni=xc + fix(y1*cos(theta));
yomni=yc + fix(y1*sin(theta));
if xomni<776 & yomni<1024 & xomni>0 & yomni>0
matrix_x(x1,y1) = xomni;
matrix_y(x1,y1) = yomni;
end
end
end
end
```

Unwrapping Omni-directional Image from Lookup Matrix

```
function[matrix]=fast(matrix_x,matrix_y,r,R,image_omni);
r1 = (R+r)/2;
for x=1:fix((2*pi*r1))
for y=1:R
xomni=matrix_x(x,y);
yomni=matrix_y(x,y);
if xomni<776 & yomni<1024 & xomni>0 & yomni>0
   matrix(x,y,1) = image_omni(xomni,yomni,1);
   matrix(x,y,2) = image_omni(xomni,yomni,2);
   matrix(x,y,3) = image_omni(xomni,yomni,3);
end
end
end
end
```

The unwrapped omni-directional image will reveal linear features for circular features on the original omni-directional image. If a circular feature is centered to the omni-directional camera, on the unwrapped image it will reveal a perfect line crossing the images horizontal x-axis. On the other hand, if a circular feature is non-centered to the omni-directional camera, then it will reveal a wave-line pattern as shown in figure 5.

From the unwrapped image a template matching is performed to match and analyze the position of the robot inside the pipe. The basic method of template

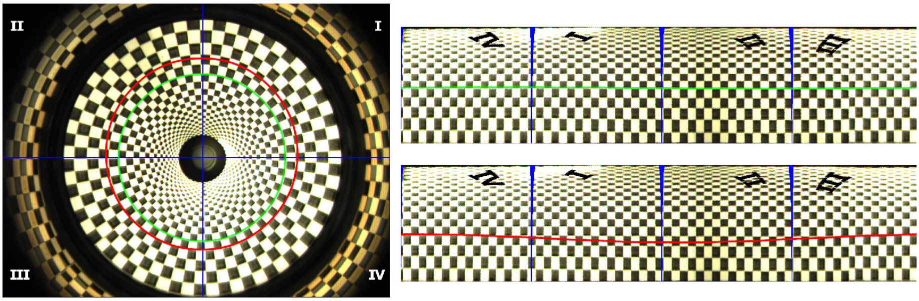


Fig. 5. Checkerboard pattern over the omni-directional camera, green circle centered, red circle translated from center (on left). Unwrapped omni-directional images (on right).

matching uses a convolution mask (template), tailored to a specific feature of the search image. The convolution output will be highest at places where the image structure matches the mask structure, where large image values get multiplied by large mask values [19].

The template matching compares the intensities of the pixels, using the sum of absolute differences (SAD). A pixel in the search image with coordinates (x_S,y_S) has intensity $I_S(x_S,y_S)$ and a pixel in the template with coordinates (x_T,y_T) has intensity $I_T(x_T,y_T)$.

The SAD in the pixel intensities is defined as:

$$Diff(x_S, y_S, x_T, y_T) = |I_S(x_S, y_S) - I_T(x_T, y_T)| \tag{3}$$

$$SAD(x,y) = \sum_{i=0}^{T_{ROWS}} \sum_{j=0}^{T_{COLS}} Diff(x+i, y+j, i, j) \tag{4}$$

Looping through the pixels in the search image as the template is translated at every pixel and takes the SAD measure.

$$\sum_{x=0}^{S_{ROWS}} \sum_{y=0}^{S_{COLS}} SAD(x,y) \tag{5}$$

The lowest SAD value refers to the best position of the template within the search image. A way to speed up the matching process is through the use of an image pyramid. These series of images with different scales generate a sequence of reduced images [20]. These lower resolution images can be searched for the template (reduced in the same way), in order to yield possible start positions for searching at the larger scales. The larger images can then be searched in a small window around the start position to find the best template location.

The mask in this case, is the average of the physical pipe-joint-gap, which is a rectangle with width equal to the width of the unwrapped panoramic image

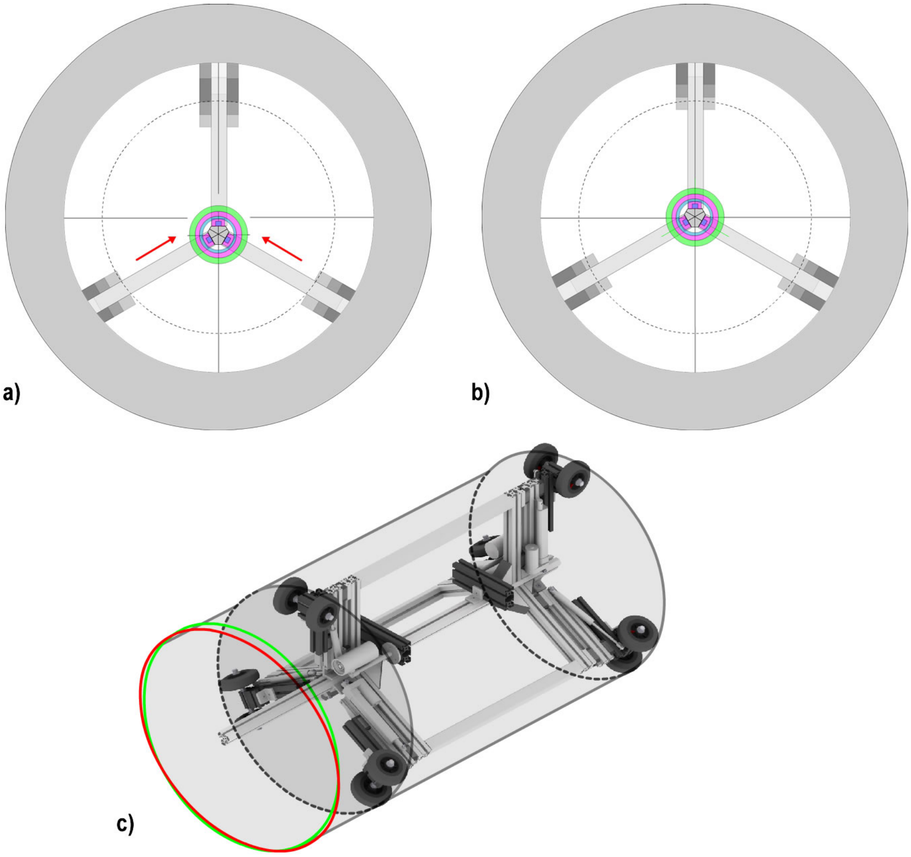


Fig. 6. Virtual DeWaLoP robot inside the pipe. a) Cross-section front-view, at time $t = n$, the robot is misaligned with respect to the pipe's center. b) Cross-section front-view, at time $t = n + 1$, the robot readjust its position to the pipe's center from visual patterns. c) Omni-directional circles detected, red circle misaligned at $t = n$ and green circle centered at $t = n + 1$.

and height equal to 16 mm, which is the average of the biggest possible gap 30 mm and the smallest possible gap 2 mm, from the pipeline specifications.

The sum of absolute differences (SAD) was selected because it provides a simple way to automate the searching for patterns inside an image, if compared to sum of squares (SSD) or cross-correlation.

4 In-Pipe Robot Positioning

In the in-pipe robot simulator, at time $t = n$, the robot was set misaligned with respect to the y cross-section of the pipe. Thus, the circle obtained from the omni-directional camera will be not centered, as shown in figures 5 and 6c. From the misaligned initial position, in figure 6a, the robot update its position, at time $t = n + 1$, by moving its virtual linear actuators to set its position in the pipes horizontal, see figure 6b. This alignment is done by knowing in which quadrant the line starts a wave-pattern. In this simulation, the misalignment was set from the top, so the two linear actuators must push the maintenance unit to move it up, to be centered.

Therefore, by analyzing the linear features from the unwrapped omni-camera image it is possible to detect the position of the in-pipe robot. In the same way, it is possible to adjust its position to be centered with respect to the pipe's x and y cross-section, so the rotating cleaning and sealing tools of the DeWaLoP in-pipe robot work properly.

5 Conclusions

This paper presents a method for centering an in-pipe robot visually, from its omni-directional camera. Instead of processing data from sensors, such as Inertial Measurement Unit (IMU) or pressure sensors. Currently, the system has been tested with a virtual robot from a real omni-directional camera.

Acknowledgement. This work is part-financed by Project DeWaLoP from the European Regional Development Fund, Cross- Border Cooperation Programme Slovakia- Austria 2007-2013.

References

1. Grassi, V., Okamoto, J.: Development of an omnidirectional vision system. Journal of the Brazilian Society of Mechanical Sciences and Engineering (2006) ISSN 1678-5878
2. Benosman, R., Kang, S.: Panoramic Vision: Sensors, Theory, and Applications. Springer, NY (2001)
3. Daniillidis, K., Klette, R.: Imaging Beyond the Pinhole Camera. Springer, New York (2006)
4. Sturm, P., Ramalingam, S., Gasparini, S., Barreto, J.: Camera models and fundamental concepts used in geometric computer vision. Foundations and Trends in Computer Graphics and Vision 6(12) (2010)
5. Scaramuzza, D.: Omnidirectional vision: from calibration to robot motion estimation. PhD thesis n. 17635. PhD thesis, ETH Zurich (2008)
6. Blosch, M., Weiss, S., Scaramuzza, D., Siegwart, R.: Vision based mav navigation in unknown and unstructured environments. In: IEEE ICRA
7. Bosse, M., Rikoski, R., Leonard, J., Teller, S.: Vanishing points and 3d lines from omnidirectional video. In: International Conference on Image Processing (2002)
8. Corke, P., Strelow, D., Singh, S.: Omnidirectional visual odometry for a planetary rover. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (2004)
9. Tardif, J., Pavlidis, Y., Daniilidis, K.: Monocular visual odome-try in urban environments using an omni-directional camera. In: IEEE/RSJ (2008)

10. Scaramuzza, D., Fraundorfer, F., Pollefeys, M.: Closing the loop in appearance-guided omnidirectional visual odometry by using vocabulary trees. Robotics and Autonomous System Journal 58(6) (2010)
11. Nassiraei, A.F., Kawamura, Y.: Concept and Design of A Fully Autonomous Sewer Pipe Inspection Mobile Robot KANTARO. In: 2007 IEEE International Conference on Robotics and Automation (2007)
12. Roh, S., Wan Kim, D., Lee, J., Moon, H., Ryeol Choi, H.: In-pipe Robot Based on Selective Drive Mechanism. International Journal of Control, Automation, and Systems 7(1), 105–112 (2009)
13. Mateos, L.A., Sousa, M., Vincze, M.: DeWaLoP Robot Remote Control for In-pipe Robot. In: ICAR 2011, pp. 518–523 (2011)
14. Mateos, L.A., Vincze, M.: DeWaLoP Robot Dynamical Independent Suspension System. In: ICMET 2011, pp. 287–292 (2011)
15. Mateos, L.A., Vincze, M.: Developing Water Loss Prevention DeWaLoP Robot Vision system. In: CET, pp. 65–68 (2011)
16. Mateos, L.A., Vincze, M.: DeWaLoP Robust Pipe Joint Detection. In: IPCV, pp. 727–732 (2011)
17. Mateos, L.A., Vincze, M.: DeWaLoP-Monolithic Multi-module In-Pipe Robot System. In: Jeschke, S., Liu, H., Schilberg, D. (eds.) ICIRA 2011, Part I. LNCS, vol. 7101, pp. 406–415. Springer, Heidelberg (2011)
18. Svoboda, T., Pajdla, T., Hlavac, V.: Central Panoramic Cameras: Geometry and Design. Research Report K335/97/147, Czech Technical University, Faculty of Electrical Engineering, Center for Machine Perception (1997)
19. Brunelli, R.: Template Matching Techniques in Computer Vision: Theory and Practice. Wiley (2009) ISBN 978-0- 470-51706-2
20. Mateos, L.A., Shao, D., Kropatsch, W.G.: Expanding Irregular Graph Pyramid for an Approaching Object. In: Bayro-Corrochano, E., Eklundh, J.-O. (eds.) CIARP 2009. LNCS, vol. 5856, pp. 885–891. Springer, Heidelberg (2009)

Electric Vehicle Automation through a Distributed Control System for Search and Rescue Operations

Marco Polo Cruz-Ramos, Christian Hassard, and J.L. Gordillo

Center for Intelligent Systems, Tecnológico de Monterrey, Monterrey, México
{mp.cruz.phd.mty,cr.hassard.phd.mty,jlgordillo}@itesm.mx

Abstract. In search and rescue operations, time plays an important role; therefore, an automatic or teleoperated deployment service to bring rescue robots near to or on a target location is desired. We propose to use a distributed control architecture to automate an electric vehicle that, after instrumentation and automation, serves as carrier robot for smaller rescue robots. Distributed control architectures rely heavily on communication networks for information exchange. The control architecture discussed on this paper is based on Controller Area Network (CAN) protocol, to which different nodes have been attached. Three main control loops are closed through the network: speed, steering and deployment mechanism for transported rescue robots. Tests were carried out to prove reliability and effectiveness of the distributed control architecture. Such tests indicated that a distributed control network based on CAN protocol is suitable for controlling different loops at real time and in hostile environments.

Keywords: Distributed control systems, CAN-bus, autonomous vehicles, marsupialism, rescue robots.

1 Introduction

In search and rescue operations, time plays an important role; therefore, an automatic or teleoperated deployment service to bring rescue robots near to or on a target location is desired. A proposal for this issue has been to employ bigger in size robots to function as carriers of smaller rescue robots. This approach is known as *marsupialism* [1]. Examples of marsupialism at different applications are: urban search and rescue operations [1], coal mine rescue missions [2], reconnaissance and surveillance [3], floor characterization [4].

Since marsupialism involves at least two robots, we refer to this Multi-robot System as a Marsupial Robotic Team (MarRT). The main drawback of former MarRTs is the size of the carrier robot. A bigger in size and power carrier robot can cover longer distances through more hostile terrains; and at the same time, a higher payload allows more complex and heavier passenger robots to be transported to a deployment point. Hence, our work proposes to automate an electric vehicle, which after instrumentation and automation, serves as carrier robot of other smaller rescues robots. To that extent, a distributed control system (DCS) is proposed as backbone for such automation process. DCSs are employed in a variety of processes that require input data from

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 249–260, 2013.

different sensors, sometimes located far from the actuators and even from the processing units. These control systems depend on on information sharing among different controlling units all interconnected through a communication channel or bus [5]. Furthermore, in order to interact and assure a correct control, these units rely heavily on a reliable and robust communication protocol.

When implementing a DCS on a vehicle, different aspects must be taken into account: distances between electronic modules vary; sent data might be subjected to electromagnetic interference (EMI); response in real time is needed because delays on communication might endanger the control's performance; among others. Hence, vehicle's instrumentation, the addition of sensors, actuators and/or processing units to replace the human operator commands, requires the use of robust vehicle network technologies in order to ensure a reliable data exchange among the added electronic modules. It must be highlighted, that there are many possible options regarding the communications architecture that can be used between the components.

For automating the proposed utilitarian vehicle, three main control loops are implemented over the DCS: *speed*, to control the main electric motor that provides traction; *steering*, to drive a motor attached to the steering wheel to have control over the steering angle of the front wheels; *deployment mechanism*, a ramp to deploy on-board rescue robots.

The rest of the sections are structured as follows: Section 2 states the basic aspects of a DCS and of CAN protocol. Section 3 outlines the hardware and firmware employed for the implementation of the DCS. Section 4 describes the experiments carried out to validate the robustness and performance of the distributed control loops on the CAN-bus; as well as the results obtained from our experiments. Finally, Section 5 refers to the conclusions obtained and depicts future work on this research line.

2 Network Controlled System

A distributed control system consisting of a series of interconnected controllers has been designed for an electric vehicle in order to achieve speed, steering and deployment mechanism control. The system's architecture defines the information flow between the interacting modules on the vehicle.

A Network Controlled System (NCS) is a type of DCS in which control loops are closed by means of an information network. Hence, the essential feature of a NCS is the data exchanged, command and feedback signals, among interconnected components through the network. This type of setup offers several advantages over traditional point-to-point control solutions, some of which are [6-7]: 1) modularity and flexibility for system design, 2) simpler and faster implementation times of the individual components, 3) ease of system diagnosis and maintenance, 4) decentralized control and, 5) increase in system's agility.

In a NCS, sensors and actuators attached directly to the desired process plant are connected to the real-time network. The controller, while physically separated from the plant, is also connected to the real-time network, closing the control loop.

A NCS can be modeled as "discrete-continuous" with time varying elements induced by the network. Induced time delays on an NCS can be summarized as the controller-actuator delay (τ_{ca}) and the sensor-controller delay (τ_{sc}), as well as the computational time required by each device performing control operations over the network [8]. These induced delays depend on different factors; e.g. network type, cable length, and baud rate. Figure 1 depicts a general representation of NCS showing the delays on sensors and controllers.

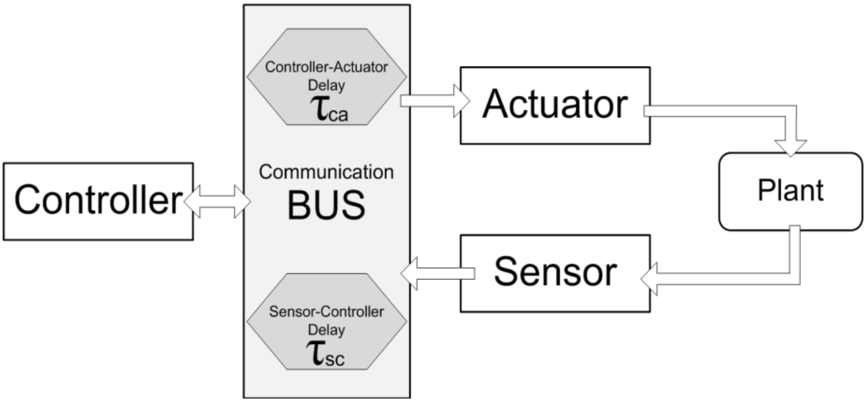


Fig. 1. General representation of a Network Control System (NCS). Command and feedback signals are transmitted through the network. This representation includes delays on the sensor and actuator's side. In addition, processing time must be taken into account.

Proper interconnection of every device on the network depends on the communication protocol used. The selected communication protocol for implementing the NCS for the electric vehicle automation was the Controller Area Network (CAN) protocol. CAN is a reliable and robust communication protocol that was designed for the automotive industry in the early 1980's by Robert Bosch GmbH company to communicate several components among themselves [9-11]. Since CAN is a message-based multimaster broadcast serial bus, it has also been used on industrial applications, as well as autonomous vehicles [12-13], robots [14-15], and on particular sections of commercial vehicles [16-17]. It has been proven suitable for real-time distributed control and to be reliable for modular configuration; and that it is possible to implement a NCS for closed-loop control with relatively low cost and high efficiency [18].

3 Implementation

For the implementation of the NCS on the utilitarian electrical vehicle, High Speed CAN (1Mbit/s baud rate) was chosen. This baud rate proved to be adequate for the information flowing through the bus (sensor data, control commands, etc.), which will be covered with more detail in Section 4.

3.1 Communication Hardware

To connect a device to the CAN bus, a special hardware is required, usually embedded in the device itself. For devices that are not originally capable of CAN communication, modules and interfaces were developed to integrate them into the CAN bus. All implemented nodes in this vehicle share the same basic structure, consisting of a microcontroller unit (MCU) or central processing unit module; a CAN controller-transceiver module capable of Serial Peripheral Interface (SPI) communication; and along with these, a custom interface, consisting of input/output channels, relays or signal conditioning elements specific for the node's application. Figure 2 depicts the general hardware configuration of each CAN-node.

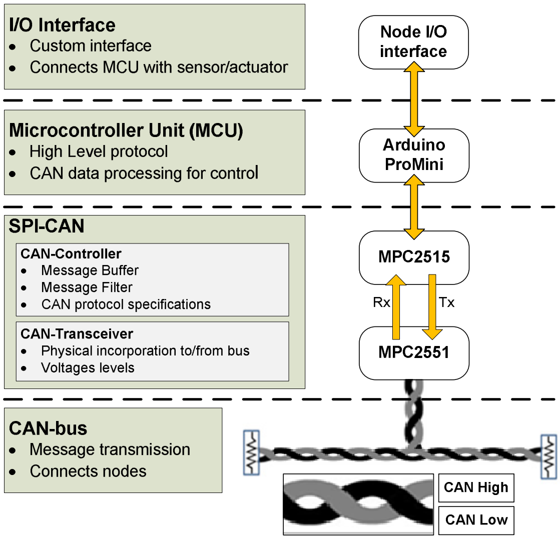


Fig. 2. General hardware configuration of each node implemented on the vehicle. This design allows any MCU capable of SPI communication to be connected to the vehicle's CAN bus. Its modularity permits modules to be interchanged; while maintaining an open possibility of MCUs and peripherals.

The CAN physical bus consists of a pair of shielded twisted cables. A node is connected to the bus through any of the hubs mounted throughout the vehicle; e.g. on the vehicle's front panel and on the rear part of the vehicle. Each hub supports up to five nodes and includes a 120 Ohms resistor as terminal ends of the bus. So far six nodes have been added to the bus; yet, due to its modularity, more nodes can be added, providing more sensing and actuating capabilities.

The controller-transceiver module or SPI-CAN module is in charge of receiving messages from the bus and transferring them via SPI to an MCU module and vice versa. This module was implemented in order to keep the design open (flexible), such that any controller with no CAN interface but with SPI interface can be used on the nodes. An MCP2551 CAN transceiver [19] and an MCP2515 CAN controller [20] were employed in this module. The former converts the bits to actual voltages from/to the bus to receive/transmit messages; while the latter filters received/transmitted messages in the bus and handles all CAN protocol specifications. This module was designed and prototyped in our laboratory.

The MCU module consists of a microcontroller with SPI interface. Programming of the microcontroller varies according to the function of the particular node. The MCU module is in charge of processing, interpreting the data contained in any message received through the SPI-CAN module, sending status or control messages to other nodes in the bus and managing the sensor or actuator it is connected to. Also the MCU module might execute control routines or process raw data given by sensors. In our implementation, an Arduino ProMini [21] (version 5V 16MHz) with an ATmega328 microprocessor was employed as common MCU in all nodes.

Finally, in order for the MCU to interact with the sensor or actuator, an Input/output interface was required. This I/O interface application is specific and might include power drivers, signal conditioning hardware, or other devices, depending on the specific node function.

3.2 Control Loops Hardware

The NCS is currently divided into three main control loops: 1) *speed* control loop, in charge of maintaining a desired RPM set point for the rear wheels; 2) *steering* control loop, in charge of placing the front wheels at a desired steering angle; and 3) *deployment mechanism* control, which drives a ramp from storage to deployment-ready position and vice versa. The first two control loop consist of two CAN-nodes each; one for actuation commands and one for sensing and control processes. In case of the deployment mechanism control, only one node is implemented; in which sensing and actuation commands are processed and generated. Figure 3 depicts the three mentioned control loops with their respective CAN-nodes, along with a sixth node for external wireless communication.

In the *speed* control loop, an actuation node is connected to the motor driver Millipak from Sevcon, which originally came with the vehicle and is in charge of regulating the current and voltage supply to the 5HP electric motor. This node (N_1) receives CAN frames from the network and processes the information received to generate a speed command value for the controller to drive the motor. On the other hand, the sensor node of this control loop (N_2) is connected to the encoder mounted on the brake's drum. This node receives the square pulse signal generated by the encoder, computes the revolutions per hour at which the wheel is turning, executes the controller routine to calculate the error and compute a control signal; finally, generates a CAN message that is broadcasted through the network. Hence, the speed control loop is closed via the CAN network once the CAN messages generated by N_2 are received and interpreted by N_1. Figure 4 (left) shows a picture of the implemented nodes for the speed control loop.

In the *steering* control loop an actuation node is connected to a motor attached to the vehicle's original steering wheel. This node (N_3) receives CAN frames from the network and moves the motor in the desired direction. The motor movement is stopped when the angular position sensor connected to the sensor node (N_4) detects that the front wheels are positioned at the desired steering angle, broadcasting the notification via the CAN network. N_3 monitors the bus, and once the CAN message broadcasted by N_4 is detected, the motor is stopped. As a result, the steering angle control loop is closed via the CAN network once the messages generated by N_4 are received and interpreted by N_3. Figure 4 (right) shows a picture of the implemented nodes for steering control loop.

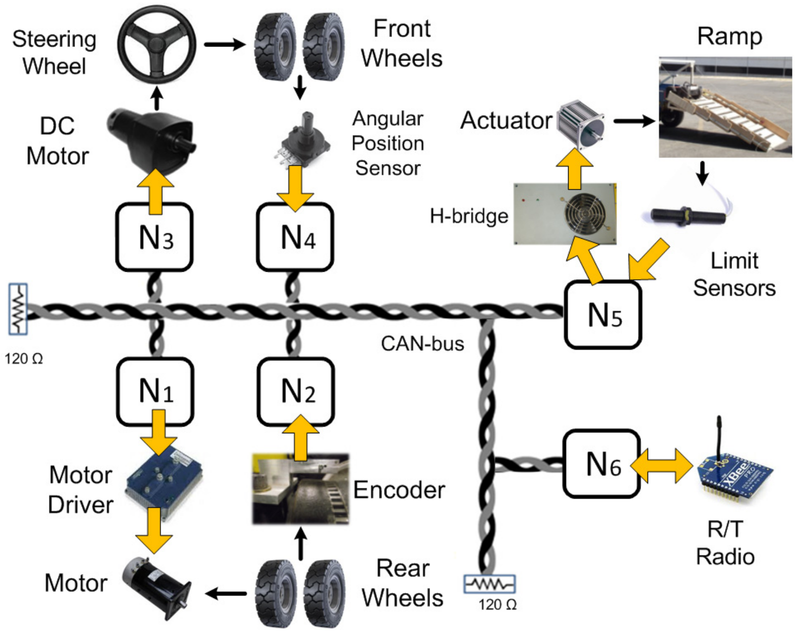


Fig. 3. Three implemented distributed control loops. Control loops send actuation commands and sensing data through a CAN-bus. Thicker arrows represent electrical connections, while thinner arrows refer to direct hardware mounting or physical interaction.

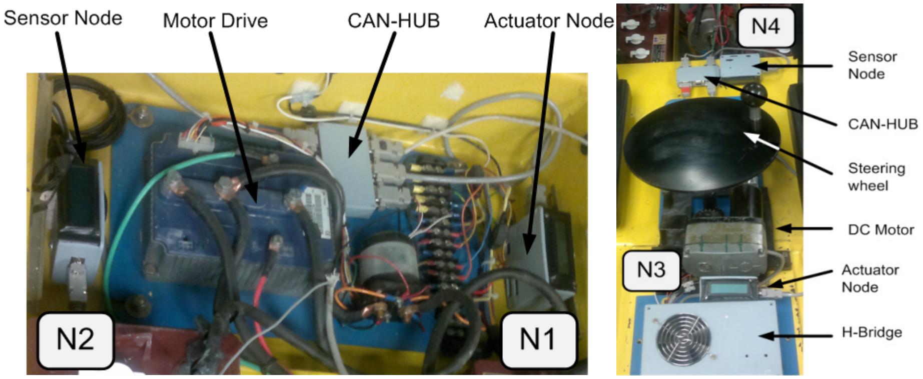


Fig. 4. Actual implementation of the nodes involved in the speed and steering control loop

In the *deployment mechanism* control loop both the actuating and sensing elements are connected to a single node (N_5). This node receives CAN frames from the network to lower or raise the ramp. Although the control loop is not closed via the CAN network, frames of the ramp's state are generated and broadcasted into the network. Figure 5 depicts the node implementation along with the deployment mechanism main hardware components. This ramp allows smaller rescue robots transported by the electric vehicle to descend to ground level once a deployment point has been reached. Thereby, the automated vehicle fulfills its role within the MarRT as carrier robot of smaller rescue robots.

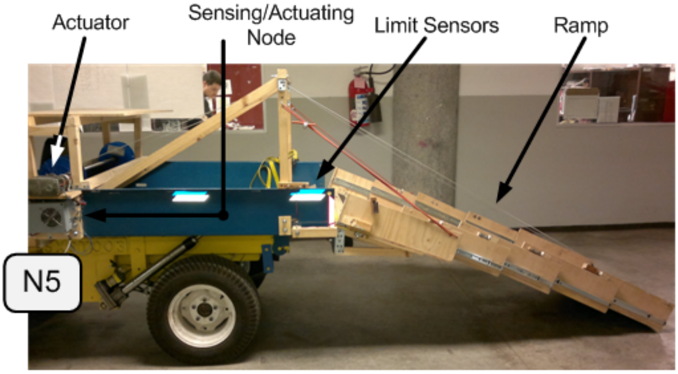


Fig. 5. Actual implementation of deployment platform node. Contrary to other control loops, only one node processes data from sensors and generates actuating commands to control the lowering and rising of the ramp for the deployment of transported rescue robots in the MarRT.

Finally, a node for wireless communication (N_6) was incorporated to the network. Employed CAN-messages were of different lengths, depending on if they were control messages, information messages or acknowledgment messages. Table 1 provides an example of the frames defined for the deployment mechanism control loops.

Table 1. Some of the messages involved with the deployment mechanism control

Receives/sends	ID	Byte 0	Byte 1
Receives (control message)	0x3 F0	0: Raise ramp; 1: Lower ramp.	Not used
Sends (status message)	0x3 F5	0: Ramp is retracted (up); 1: Ramp is deployed (down).	0: No child vehicles in bay; 1: Child vehicles are present in bay.

4 Experiments and Results

The NCS was validated on a real experimental platform. Such experimental platform consists of an electric vehicle made by Johnson Industries, model SuperTruck instrumented with the mentioned prototyped CAN nodes. Figure 6 depicts the computer-aided design (CAD) model and a photograph of the actual MarRT: the carrier robot and two smaller robots for search and rescue operations.

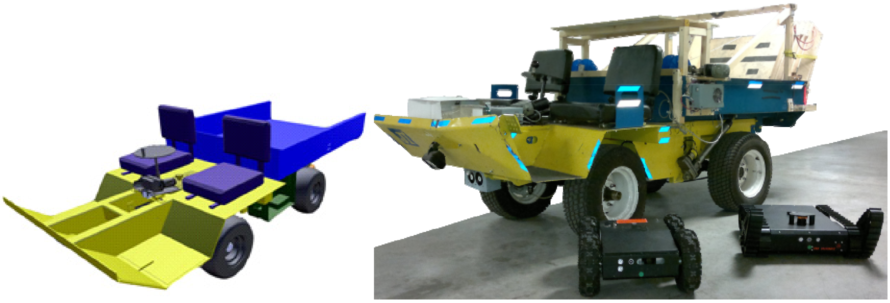


Fig. 6. Marsupial Robotic Team (MarRT). On the left, the CAD model of the carrier robot generated through inverse engineering for simulation and design of mechanical components. On the right, actual electrical vehicle instrumented through a CAN-based network controlled system and two of the search and rescue robots it can transport to a deployment point.

The carrier robot, is an electric vehicle with Ackerman-like configuration, hence the control of the steering angle of the front wheels is needed for the automation of the orientation of the vehicle; a control loop to for the speed is also expected; and a third control loop to lower and raise a mechanism (e.g. ramp) to deploy the transported robots is required.

When the vehicle is powered-up, as default it sets speed values to zero (not moving) and the steering angle to zero degrees (facing straight ahead). The desired speed is coded in a set-point message with ID 0x300, while the desired steering angle is coded in a set-point message with ID 0x350 and the ramp desired state with ID 0x3F0. All set points are defined in an off-board computer with a Human Machine Interface (HMI). To send the set-point messages a RF module connected to the computer is used; then sent/received by an on-board RF module (N$_6$), which messages are interpreted by the MCU located on this node; finally, once codified, the messages are forwarded into the CAN-bus.

Once the corresponding controlling node receives the set-point, either speed (N$_2$), or steering angle (N$_4$), it would automatically try to reach it by comparing the desired set-point with the lecture broadcasted by the sensor nodes, N$_1$ for speed feedback or N$_6$ for measured steering angle. If the node finds a difference between the desired set-point and the actual reading, it generates a control message to be sent to the corresponding actuator node. For the deployment mechanism control, a message to lower or raise the ramp is sent. Once received, the actuator/sensor node (N$_5$) drives the motor either CW to lower the ramp or CCW to raise it. The ramp moves until limit sensors are activated.

To monitor the performance of the NCS, experiments were conducted. On one hand, different set-points were manually given to both, the speed and steering controllers. On the other hand, the deployment mechanism was commanded to lower and rise repeatedly. Figure 7 shows how the vehicle's speed behaved, while Fig. 8 shows the vehicle's steering angle response, both cases at different set-points.

We can notice that the vehicle was actually capable of reaching the desired set-point in all cases, but continued with some oscillations around it. This was due to the simplicity of the control algorithms programmed so we could first test the performance of the network. It is planned to improve these control algorithms with a PID or a Fuzzy Logic based controller in future work to achieve a better control.

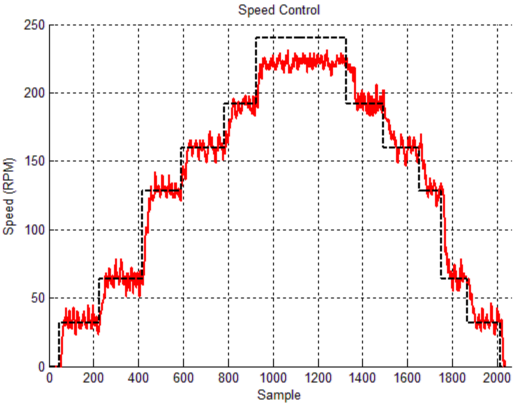


Fig. 7. Vehicle's speed response at different given set-points. Dashed-line represents the given speed set-points, while the continuous line the measured speed. The vehicle was not able to achieve maximum set-point (240 RPM) because this value exceeded the vehicle's maximum speed (220 RPM).

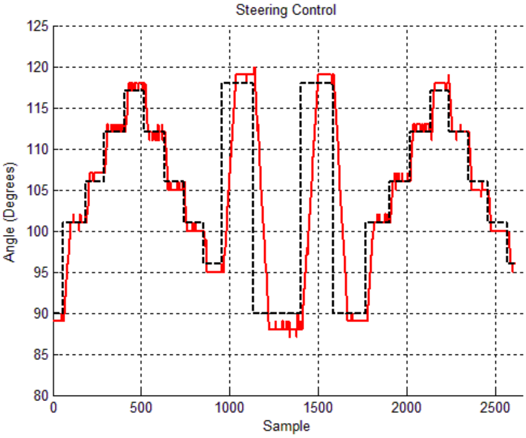


Fig. 8. Vehicle's steering angle response to given set-points. Dashed-line represents the given angle set-points, while the continuous line the measured angle.

Additionally, the bus bandwidth used by the NCS was monitored. Figure 9 shows a snap-shot of one of the control messages sent from one of the sensor nodes to its corresponding actuator node (left); and also shows how often these messages were sent (right). It can also be noted that a typical control message took around 120 microseconds; meaning that the delay in communication between sensor and actuator is actually not significant for this implementation.

Further, Fig. 9 shows that each message of a given control loop was sent with 25 milliseconds between them and approximately 70 milliseconds of time between different control loops messages. We can see that most of the bus bandwidth is still available. Hence, we can safely implement more nodes in the network without compromising the performance of the already installed hardware. Once the main loops were automated, speed, steering, and deployment mechanism, additional sensors, actuators, or controller can be added to the network; e.g IMU, GPS, laser, robotic arm manipulator, data logger, pose estimation unit.

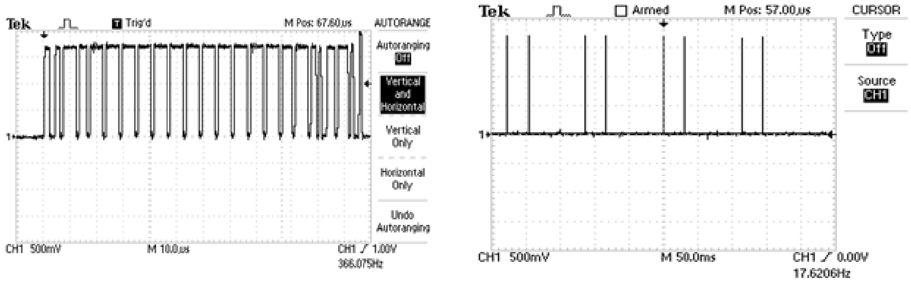


Fig. 9. Example of some of the CAN messages on the vehicle bus. On the left, control message sent from a sensor node to an actuator node; on the right, snapshot of the messages on the CAN bus. Each spike corresponds to a broadcasted message.

Owing to the advantage that CAN protocol handles message collisions and priorities, no message was lost. As a result, the vehicle was able to achieve the desired speed and steering angle and control the deployment mechanism, all through messages sent over the communication bus.

5 Conclusions and Future Work

A carrier robot to transport and deliver rescue robots to a destination site in an autonomously way was presented. Once the deploy point is reached, a ramp is lowered, allowing on-board rescue robot to be deployed. To increase the payload of carrier robot, an electric utilitarian vehicle was instrumented and automated by implementing a distributed control system. Such DCS worked as expected, allowing us to control its speed, steering angle and the state of the deployment mechanism. Communication was effectively established and characteristics of the chosen protocol, Controller Area Network (CAN), such as robustness and reliability were verified by operating the network at maximum speed with control, status and acknowledgment messages transmitted at regular intervals. This was made with physically distributed devices within an electrical vehicle. Additionally, electronic design of PCBs for the modules and interfaces within each node is simple, low cost and reliable.

It was demonstrated with the implementation of two different control loops closed over the bus (speed and steering) and one third controller in charge of lowering and raising the ramp for deployment of on-board robots that the induced delays due on the network do not affect significantly as to mitigate the performance of the controllers.

The use of a standardized SPI to CAN interface made the system more flexible about the microprocessor or electronic control unit that can communicate through the bus. One advantage of implementing a Network Controlled System using bus protocol is the possibility of including new nodes without disruption of the whole network. The robustness of the system permits the removal of current modules and insertion of new ones without sacrificing the existing functionality of the control architecture. This means that the system can grow in complexity without much redesign; hence the network proposed is not limited to the six nodes presented. In this manner, the implemented CAN network can be expanded, adding more nodes to successfully build a completely autonomous vehicle that serves as carrier robot.

As the control algorithms can be modified as long as the controller's output travels through the network with the proper structure to be received by the corresponding actuation node, some of the ongoing work is the design of PID and Fuzzy logic based controllers, implementing them into the currently installed microprocessors to test the system performance and reliability.

In addition, the main advantage of having a utilitarian vehicle as carrier robot is not only that its payload is higher, but also that it has longer travel coverage and it can traverse through rough terrains (better traversability). Finally, other modules are planned to be added to the vehicle's bus, like: path planning module, obstacle avoidance module, bumpers, laser, among others to provide more autonomy to the vehicle.

References

1. Murphy, R.R., Ausmus, M., Bugajska, M., Ellis, T., Johnson, T., Kelley, N., Kiefer, J., Pollock, L.: Marsupial-like Mobile Robot Societies. In: Proceedings of the Third Annual Conference on Autonomous Agents, pp. 364–365. ACM Press (1999)
2. Zhao, J., Liu, G., Liu, Y., Zhu, Y.: Research on the Application of a Marsupial Robot for Coal Mine Rescue. In: Xiong, C.-H., Liu, H., Huang, Y., Xiong, Y.L. (eds.) ICIRA 2008. LNCS (LNAI), vol. 5315, pp. 1127–1136. Springer, Heidelberg (2008)
3. Hougen, D.F., Benjaafar, S., Bonney, J.C., Budenske, J.R., Dvorak, M., Gini, M., Krantz, D.G., Li, P.Y., Malver, F., Nelson, B., Papanikolopoulos, N., Rybski, P.E., Stoeter, S.A., Voyles, R., Yesin, K.B.: A miniature robotic system for reconnaissance and surveillance. In: Proc. of the IEEE Int'l. Conf. on Robotics and Automation, San Francisco, CA, U.S.A, pp. 501–507 (2000)
4. Anderson, M., McKay, M., Richardson, B.: Multirobot automated indoor floor characterization team. In: Proc. of the IEEE Int'l. Conference on Robotics and Automation (1996)
5. El-Sadek, M.Z., Hemeida, A.M., Abelwahab, M.A., Alkosy, A., Younies, S.A.: Distributed control system approach for a unified power system. In: Universities Power Engineering Conference, pp. 947–950 (2004)
6. Huo, Z., Fang, H., Ma, C.: Networked control system: state of the art. In: Fifth World Congress on Intelligent Control and Automation, vol. 2, pp. 1319–1322 (2004)
7. Chow, M., Tipsuwan, Y.: Network-based control systems: a tutorial. In: The 27th Annual Conference of the IEEE Industrial Electronics Society, vol. 3, pp. 1593–1602 (2001)

8. Lihua, Z., Yuqiang, W., Lei, G., Haojie, C.: A Novel Research Approach on Network Control Systems. In: Int. Conf. on Internet Computing in Science and Engineering, pp. 262–265 (2008)
9. Robert Bosch GmbH, Controller Area Network (CAN) Specification, `http://www.bosch-semiconductors.de/media/pdf/canliteratur/can_fd_spec.pdf`
10. Lima, G.M.A., Burns, A.: A Consensus Protocol for CAN-Based Systems. In: Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS 2003). IEEE Computer Society, Washington, DC (2003)
11. Juexiao, C., Feng, L., Zechang, S.: Reliability Analysis of CAN nodes under Electromagnetic Interference. In: IEEE International Conference on Vehicular Electronics and Safety, ICVES 2006, pp. 367–371 (2006)
12. Gurram, S., Conrad, J.: Implementation of can bus in an autonomous all-terrain vehicle. Proceedings of IEEE, 250–254 (2011)
13. Weidong, L., Li'e, G., Yilin, D., Jianning, X.: Communication Scheduling for CAN bus Autonomous Underwater Vehicles. In: Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation, pp. 379–383 (2006)
14. Lesperance, E., Desbiens, A.L., Roux, M.-A., Lavoie, M.-A., Fauteux, P.: Design of a small and low-cost power management unit for a cockroach-like running robot. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2387–2392 (2005)
15. Barker, O., Beranek, R., Ahmadi, M.: Design of a 13 degree-of-freedom biped robot with a CAN-based distributed digital control system. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), pp. 836–841 (2010)
16. Fredriksson, L.: CAN for Critical Embedded Automotive Networks. IEEE Micro 22(4), 28–35 (2002)
17. Guo, H., JieAng, J., Wu, Y.: Extracting Controller Area Network data for reliable car communications. In: Intelligent Vehicles Symposium, pp. 1027–1032 (2009)
18. Del Bosque, J.: Autonomous ground vehicle automation: Can bus implementation and speed control. Masterthesis, Instituto Tecnologico de Estudios Superiores de Monterrey, Mexico (2010)
19. MCP2551 high-speed CAN transceiver, Data Sheet
20. MCP2515 stand-alone CAN controller, Data sheet
21. Arduino Pro Mini, `http://arduino.cc/en/Main/ArduinoBoardProMini`

Determination of the Instantaneous Initial Contact Point on a Parallel Gripper Using a Multi Input Fuzzy Rules Emulated Network Controller with Feedback from Ultrasonic and Force Sensors

César Navarro, Chidentree Treesatayapun, and Arturo Baltazar

Centro de Investigación y Estudios Avanzados - Unidad Saltillo,
Robotics and Advanced Manufacturing Program
cesar.navarro.777@gmail.com,
{chidentree,arturo.baltazar}@cinvestav.edu.mx

Abstract. Many applications of robotic manipulators require a precise applied force control to determine the instantaneous initial contact (force approaching zero) as accurate as possible. Thus, the transient state from free to restricted motion needs to be taken into account. In this work, a multi input fuzzy rules emulated network (MiFREN) control scheme with adaptation is developed to find the first contact position between the fingers of a parallel gripper and a soft object. We propose the use of an ultrasonic sensor (with Hertzian contact) working simultaneously with a high sensitivity load cell. The IF-Then rules for MiFREN controller and a new cost function using both the ultrasonic signal and the contact force are proposed. The results show that the proposed controller is capable of finding the instantaneous initial contact without any knowledge about the object, material properties and/or its location on the working space.

Keywords: Instantaneous initial contact, neuro-fuzzy, adaptive control, transient phase.

1 Introduction

A successful force contact control method requires dealing with the collisions between the manipulator and the contact surface. A collision could generate an overshoot (impulse force) that can destabilize the control [1]. The problem of transient control from free to restricted motion has been studied [2], [3], [4], and [5]. Some authors use the impulse force value to decide when to switch between free motion control and contact force control [2]. However, oscillations in force due the bouncing of the manipulator can be observed. To minimize this phenomenon, the impact mechanics of the manipulator has been modeled [3]. Hence a smooth transition between free and restricted motion is required. This can be achieved by using vision based method [4] or by controlling the manipulator trajectory and using flexible elements [5].

In this work, we propose a control scheme to find the instantaneous initial contact point between the two fingers of an electric parallel gripper and a soft object. To

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 261–272, 2013.

reach the instantaneous initial contact a smooth transition between free and constrained motion is required. To accomplish that, a new multi-input fuzzy rule emulated network (MiFREN) control scheme using ultrasonic and force sensors signals as feedback is proposed.

Recently, Armendaríz *et al.* [6], [7], [8] presented a force contact sensor based on the Hertzian contact and ultrasound. The sensor has a hemispherical body that is attached to a piezoelectric transducer. The transducer induces mechanical waves into the hemispherical body; if the body is not making contact with a surface, all the mechanical waves are reflected back to the transducer; on the other hand, if a contact occurs some energy is transmitted to the touched material and some is reflected back to the transducer. The decrease of the reflected energy is indicative of contact and it is used to find the instantaneous initial contact.

This works is organized as follows: first, the nature of the feedback signals and MiFREN controller design are explained; second, the linear consequence adaptation algorithm is discussed and control stability results are shown; finally, the experimental platform and the results of experiments are shown.

2 Feedback Signals

In this work we propose to use two different feedback signals in the instantaneous initial contact controller; one of the signals is obtained from a force contact sensor and the other one from an ultrasonic sensor.

The force feedback signal alone is not reliable in the transient state from free to restricted motion. The collision of the gripper fingers causes an impulse force ([9]) that is read by the force sensor; this impulse force can affect the controller performance. Fig. 1 shows the read force in collisions with different velocities. However, the ultrasound feedback signal doesn't present any nonlinearity when a collision occurs. These signals are noisy and can have detrimental effects on the controller. Fig. 2 shows the root mean square (RMS) of the reflected mechanical waves provided by the ultrasonic sensor with collisions at different velocities. We propose to use a controller that takes advantage of the two feedback signals by exploiting the invariance of the ultrasonic signal in collisions and the smoothness of the force readings.

3 Design of the Multi Input Fuzzy Rules Emulated Network Controller Used to Find the First Contact Point

3.1 Design of the MiFREN Controller

A control algorithm capable to manage the ultrasound and force feedback signals is necessary. The classic controllers (like PID) are useless because they don't allow using more than one input. We propose to do a modification to the MiFREN controller [10] to control the gripper by using the force ($F(k+1)$) and ultrasonic ($RMS(k+1)$) feedback signals.

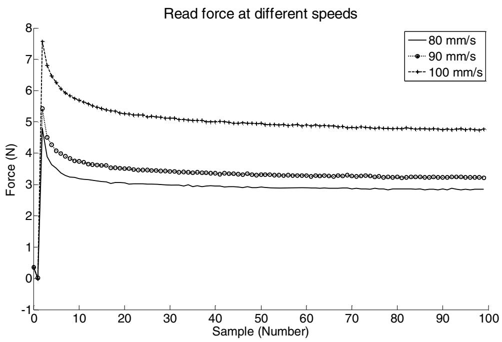


Fig. 1. Read force in a collision with different finger speeds

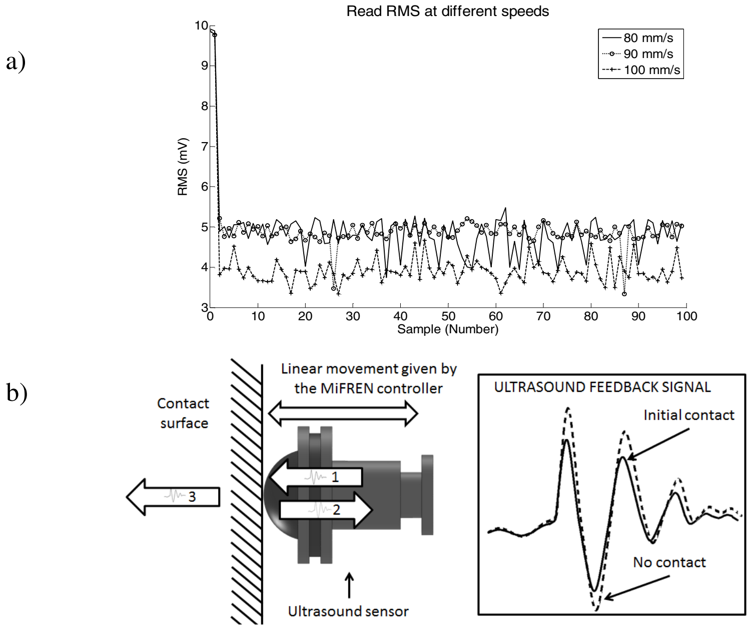


Fig. 2. a) RMS readings during a collision with different finger speeds. b) The ultrasonic sensor: The sensor generates an ultrasound signal (←1); some energy of the signal is transmitted to the contact surface (←3) and the rest is reflected back to the sensor (→2).

Sensors (ultrasonic and load) are attached to each finger (two fingers) of the gripper. The distance between the fingers is controlled to modify the feedback signals. The signal control (Δd) modifies the fingers opening to approach the feedback signals to the expected values ($F_{ex}(k)$ and $RMS_{ex}(k)$). Fig. 3 illustrates the block diagram of the system.

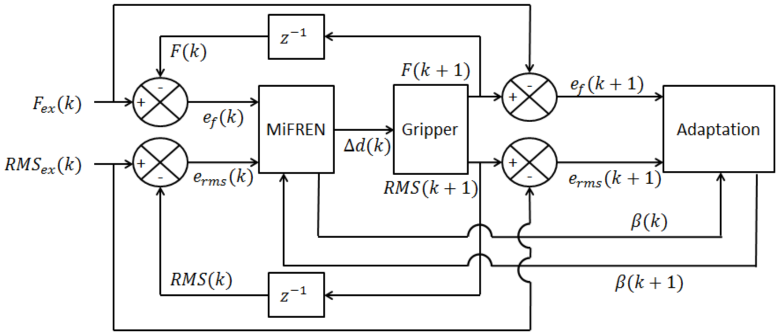


Fig. 3. Block diagram for the controlled system with the adaptation algorithm

Fig.4 illustrates the network topology of the MiFREN controller. Layer 1 corresponds to the input nodes. The layer 2 is formed by the membership function nodes; in this layer there is a node for every linguistic characteristic of the inputs. Layer 3 is the fuzzy inference system; the output of every node is calculated as

$$\Phi_i(k) = \mu_j\left(e_f(k)\right)\mu_k\left(e_{RMS}(k)\right) , \tag{1}$$

where $i = [1, \ldots ,9]$; $\mu_j\left(e_f(k)\right)$ is the j-th linguistic characteristic related to the contact force error; $\mu_k\left(e_{RMS}(k)\right)$ is the k-th linguistic characteristic related to the ultrasonic error. Layer 4 is known as the Linear Consequence layer and is analog to a defuzzification process. The output for every node is calculated as

$$O_i(k) = \beta_i(k)\Phi_i(k) , \tag{2}$$

where $i = [1, \ldots ,9]$ and $\beta_i(k)$ is the i-th linear consequence parameter. Layer 5 is the output, namely the control signal $\Delta d(k)$. The output of this node is calculated as

$$\Delta d(k) = \sum_1^9 O_i\,(k) . \tag{3}$$

The fuzzy rules for the MiFREN controller are designed taking into account all of the possible case scenarios for the interactions between the sensors and the object. The fuzzy rules information is contained in the table 1. The designed fuzzy rules are described below.

Rule 1. If $F(k)$ is zero and $RMS(k)$ is zero then $\Delta d(k)$ is positive large.
Rule 2. If $F(k)$ is zero and $RMS(k)$ is positive small then $\Delta d(k)$ is almost zero.
Rule 3. If $F(k)$ is zero and $RMS(k)$ is positive large then $\Delta d(k)$ is positive medium.
Rule 4. If $F(k)$ is negative small and $RMS(k)$ is zero then $\Delta d(k)$ is positive small.
Rule 5. If $F(k)$ is negative small and $RMS(k)$ is positive small then $\Delta d(k)$ is zero.
Rule 6. If $F(k)$ is negative small and $RMS(k)$ is positive large then $\Delta d(k)$ is negative small.
Rule 7. If $F(k)$ is negative large and $RMS(k)$ is zero then $\Delta d(k)$ is positive negative medium 1.

Rule 8. If $F(k)$ is negative large and $RMS(k)$ is positive small then $\Delta d(k)$ is negative medium 2.
Rule 9. If $F(k)$ is negative large and $RMS(k)$ is positive large then $\Delta d(k)$ is negative large.

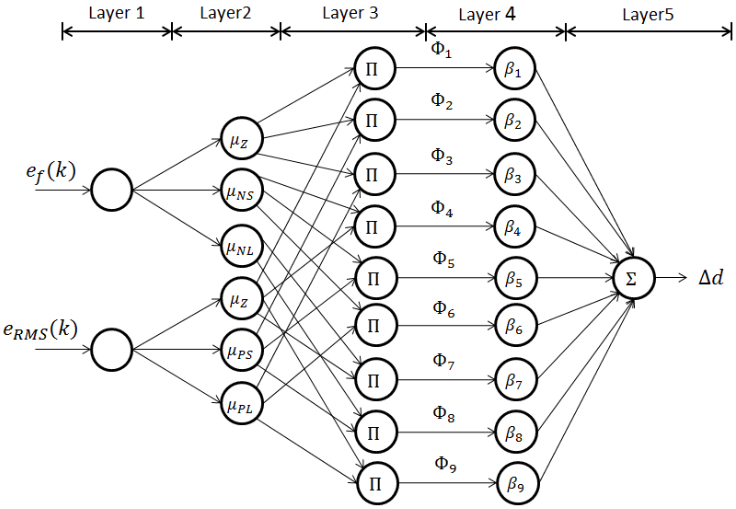


Fig. 4. Network structure of the MiFREN controller

The linguistic features of the feedback signals are evaluated by the membership functions. Fig. 5 and Fig. 6 show the membership functions used for the force and ultrasonic signals. The linear consequence deffuzifies the obtained information from the membership function layer. The linear consequence is set in mm and small values are used because a minimal change in the distance between the fingers causes a large alteration in the contact force. Table 2 lists the linear consequence values.

Table 1. Magnitudes for the control signal $\Delta d(k)$ in function of the force and RMS readings

Force and ultrasonic sensor interactions			
RMS decay Read force magnitude	Zero	Positive small	Positive large
Zero	Positive large	Almost zero	Positive medium
Negative small	Positive small	Zero	Negative small
Negative large	Negative medium 1	Negative medium 2	Negative large

Table 2. List of the values used for the linear consequence; every value of β represent a displacement in mm

Linear consequence parameter	Value	Linguistic consequence
β_1	3 mm	Positive large
β_2	2 mm	Positive medium
β_3	1 mm	Positive small
β_4	0.5 mm	Almost zero
β_5	0 mm	Zero
β_6	-0.5 mm	Negative small
β_7	-1 mm	Negative medium 1
β_8	-2 mm	Negative medium 2
β_9	-3 mm	Negative large

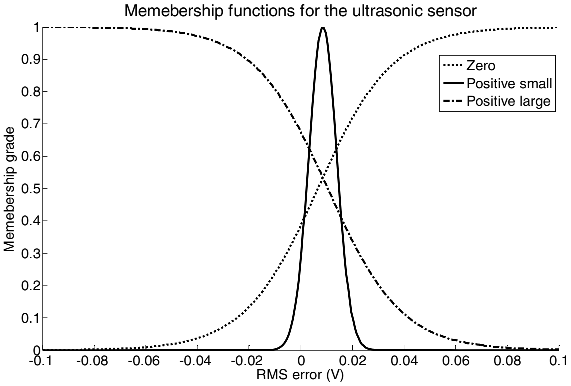


Fig. 5. Membership functions for the ultrasound sensor readings

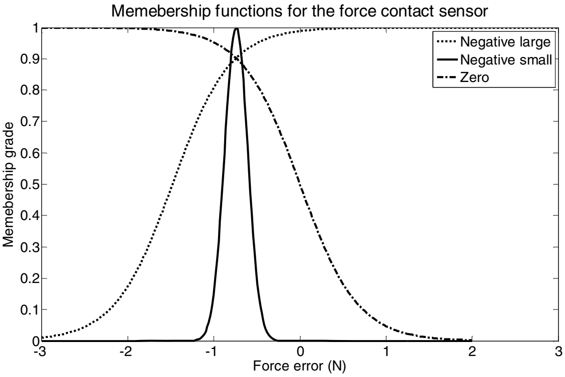


Fig. 6. Membership functions for the force sensor readings

3.2 Adaptation Algorithm for the Linear Consequence in the MiFREN Controller

The criteria used to choose the design parameters is mostly based on the experience or previous knowledge about the system, however there's no guarantee that these parameters are the best ones for a good control performance. In his work, the linear consequence is automatically adjusted by the gradient descent method. The proposed cost function to minimize is

$$J(k+1) = \frac{\gamma_1}{2} e_f{}^2(k+1) + \frac{\gamma_2}{2} e_{rms}{}^2(k+1) \, , \tag{4}$$

where $e_f{}^2(k+1)$ and $e_{rms}{}^2(k+1)$ are the errors obtained by the difference between the expected values and the current ones in an instant $k+1$. The terms γ_1 and γ_2 are constants used to weight the errors. In an instant $k+1$ the linear consequence parameter $\beta_i\ (k+1)$can be calculated as

$$\beta_i(k+1) = \beta_i(k) - \eta\left(\frac{\partial J(k+1)}{\partial\beta_i(k)}\right) , \tag{5}$$

where η is the learning rate for the adjusted parameter. The Leibniz notation for differential term from (5) is

$$\begin{aligned}\left(\frac{\partial J(k+1)}{\partial\beta_i(k)}\right) &= \left(\frac{\partial\frac{\gamma_1}{2}e_f{}^2(k+1)}{\partial F(k+1)}\right)\left(\frac{\partial F(k+1)}{\partial\Delta d(k)}\right)\left(\frac{\partial\Delta d(k)}{\partial\beta_i(k)}\right) \\ &+ \left(\frac{\partial\frac{\gamma_2}{2}e_{rms}{}^2(k+1)}{\partial RMS(k+1)}\right)\left(\frac{\partial F(k+1)}{\partial\Delta d(k)}\right)\left(\frac{\partial\Delta d(k)}{\partial\beta_i(k)}\right) .\end{aligned} \tag{6}$$

The solution of (6) is

$$\left(\frac{\partial J(k+1)}{\partial\beta_i(k)}\right) = \left[-\gamma_1 y_f e_f(k) - \gamma_2 y_{rms} e_{rms}(k)\right](\Phi_i) \, , \tag{7}$$

where y_f is $\partial F(k+1)/\partial\,\Delta d(k)$ and y_{rms} is $\partial RMS(k+1)/\partial\,\Delta d(k)$. Replacing equation (7) in (5) the result is

$$\beta_i(k+1) = \beta_i(k) + \eta\Phi_i(k)\left[\gamma_1 y_f e_f(k+1) + \gamma_2 y_{rms} e_{rms}(k+1)\right] . \tag{8}$$

The linear consequence can be tuned by using (8).

3.3 Stability Analysis and Learning Rate Selection

The value of the learning rate affects directly the control performance. If the learning rate is high, the stability is compromised; moreover, if the learning rate is small the system convergence is slow. The operational range for the learning rate where the control is stable is calculated by Lyapunov. The candidate function is

$$V(k) = \frac{\gamma_1}{2} E_f^2(k) + \frac{\gamma_2}{2} E_{rms}^2(k) \, , \tag{9}$$

where $E_f(k)$ is $F_r - F(k)$ and E_{rms} is $RMS_r - RMS(k)$. The equation (9) is defined positive and is differentiable, to prove the stability system its derivative must be negative definite. From (9) we have

$$\Delta V(k) = \tfrac{\gamma_1}{2} E_f^2(k+1) + \tfrac{\gamma_2}{2} E_{rms}^2(k+1) - \tfrac{\gamma_1}{2} E_f^2(k) - \tfrac{\gamma_2}{2} E_{rms}^2(k) \ , \tag{10}$$

where $E_f(k+1)$ is equals to $E_f(k) + \Delta E_f(k)$ and $E_{rms}(k+1)$ is equals to $E_{rms}(k) + \Delta E_{rms}$. Solving (10) we obtain

$$\Delta V(k) = \gamma_1 \Delta E_f(k) \left[E_f(k) + \tfrac{\Delta E_f(k)}{2}\right] + \gamma_2 \Delta E_{rms}(k) \left[E_{rms}(k) + \tfrac{\Delta E_{rms}(k)}{2}\right] . \tag{11}$$

From (11) the term $\Delta E_f(k)$ can be calculated as

$$\Delta E_f(k) = \tfrac{\Delta E_f(k)}{\Delta \beta_i(k)} \Delta \beta_i(k) \approx \tfrac{\partial E_f(k)}{\delta \beta_i(k)} \Delta \beta_i \ (k), \tag{12}$$

when $\Delta\beta_i(k) = \beta_i(k+1) - \beta_i(k)$. Following the same reasoning for the term $\Delta E_{rms}(k)$ we have

$$\Delta E_{rms}(k) = \tfrac{\Delta E_{rms}(k)}{\Delta \beta_i(k)} \Delta \beta_i(k) \approx \tfrac{\partial E_{rms}(k)}{\delta \beta_i(k)} \Delta \beta_i(k) \ . \tag{13}$$

The partial derivative solution from (12) is

$$\frac{\partial E_f(k)}{\delta \beta_i(k)} = \frac{\partial E_f(k)}{\partial F(k+1)} \frac{\partial F(k+1)}{\partial \Delta d(k)} \frac{\partial \Delta d(k)}{\delta \beta_i(k)} = -y_f \Phi_i(k) \ . \tag{14}$$

By another hand, the solution for the derivative term in (13) is

$$\frac{\partial E_{rms}(k)}{\delta \beta_i(k)} = \frac{\partial E_{rms}(k)}{\partial RMS(k+1)} \frac{\partial RMS(k+1)}{\partial \Delta d(k)} \frac{\partial \Delta d(k)}{\delta \beta_i(k)} = -y_{rms} \Phi_i(k) \ . \tag{15}$$

Since $\beta_i(k+1)$ is equal to $\beta_i(k) + \Delta\beta_i(k)$, from (8) the term $\Delta\beta_i(k)$ is

$$\Delta\beta_i(k) = \eta \Phi_i(k)\left[\gamma_1 y_f e_f(k+1) + \gamma_2 y_{rms} e_{rms}(k+1)\right] . \tag{16}$$

Substituting the equation (13) and (12) into (11) we have

$$\begin{aligned} \Delta V(k) = \gamma_1 \frac{\partial E_{rms}(k)}{\delta \beta_i(k)} \Delta\beta_i(k) \left[E_f(k) + \frac{1}{2} \frac{\partial E_{rms}(k)}{\delta \beta_i(k)} \Delta\beta_i(k)\right] \\ + \gamma_2 \frac{\partial E_{rms}(k)}{\delta \beta_i(k)} \Delta\beta_i(k) \left[E_{rms}(k) + \frac{1}{2} \frac{\partial E_{rms}(k)}{\delta \beta_i(k)} \Delta\beta_i(k)\right] . \end{aligned} \tag{17}$$

Then substituting (14), (15) and (16) into (17) we have

$$\begin{aligned}\Delta V(k) = \eta\Phi_i^2(k)S(k)\Big\{&-\gamma_1 y_f\left[E_f(k) - \frac{1}{2}y_f\Phi_i^2(k)\eta S(k)\right]\\ &- \gamma_2 y_{rms}\left[E_{rms}(k) - \frac{1}{2}y_{rms}\Phi_i^2(k)\eta S(k)\right]\Big\},\end{aligned} \tag{18}$$

where $S(k) = \gamma_1 y_f E_f(k+1) + \gamma_2 y_{rms}E_{rms}(k+1)$. Doing some algebra from (18) we obtain

$$\begin{aligned}\Delta V(k) = &-\eta\Phi_i^2(k)S(k)\big[\gamma_1 y_f E_f(k) + \gamma_2 y_{rms}E_{rms}(k)\big]\\ &+\frac{1}{2}\eta^2 S^2(k)\Phi_i^4(k)\big[\gamma_1 {y_f}^2 + \gamma_2 {y_{rms}}^2\big].\end{aligned} \tag{21}$$

In order to have a stable system, is needed obey the following relation obtained from (21)

$$\begin{aligned}&\eta\Phi_i^2(k)S(k)\big[\gamma_1 y_f E_f(k) + \gamma_2 y_{rms}E_{rms}(k)\big]\\ &\quad> \frac{1}{2}\eta^2 S^2(k)\Phi_i^4(k)\big[\gamma_1 {y_f}^2 + \gamma_2 {y_{rms}}^2\big].\end{aligned} \tag{22}$$

With (22) the appropriate range for η to maintain stability is

$$0 \le \eta < \frac{2\left(\gamma_1 y_f E_f(k) + \gamma_2 y_{rms}E_{rms}(k)\right)}{\Phi_i^2(k)S(k)\left(\gamma_1 {y_f}^2 + \gamma_2 {y_{rms}}^2\right)}. \tag{23}$$

Without loss of generality (23) can be rewritten as

$$0 \le \eta < \frac{2\left(\gamma_1 y_f E_f(k) + \gamma_2 y_{rms}E_{rms}(k)\right)}{\Phi_i^2(k)S(k)\left(\gamma_1 {y_f}^2 + \gamma_2 {y_{rms}}^2\right) + \varepsilon}, \tag{24}$$

with ε denotes the design parameter with small value. The relation obtained in (24) can be used to estimate the learning rate for tuning parameters inside MiFREN.

Regarding this control scheme, the proposed controller does not require a mathematical model of the physical phenomenon. Furthermore, the controller is able to manage two signals - force sensor and ultrasonic sensor- according to the proposed cost function. The cost function can predict correctly the instantaneous initial contact despites the impulse force.

4 Experimental Setup and Results

The experimental platform has an electric parallel gripper WSG50 by Weiss Robotics working together with the proposed controller based on MiFREN. The gripper is manipulated through a PC and is communicated by an Ethernet connection. The gripper fingers have been customized to hold the force and ultrasonic sensors. A piezoelectric transducer generates the ultrasonic feedback signal. The piezoelectric element is connected to a pulser/receiver machine. This machine is connected to a Tektronix TDS1012 oscilloscope and the ultrasound signal is processed. The oscilloscope sends the processed signal to the PC through a GPIB connection. The F/T Mini40 sensor reads the contact force and is connected to the PC through a data acquisition card from National Instruments. The sensors' readings, the manipulation of the gripper and the control program are integrated in LabVIEW 2010. Fig. 7 shows the overall experimental platform. The test material used to find the instantaneous initial contact is a plastic hollow sphere which is free to move.

According to (24), the learning rate η is 0.01 when is $\gamma_1 = 1$ and $\gamma_2 = 1$ as the initial state. The MiFREN controller with the linear consequence adaptation is set to reach an empirical instantaneous initial contact and the ideal instantaneous initial contact. The expected values for the empirical contact were obtained by manually moving the gripper fingers to the contact surface; these values are 0.7 newton and 0.0065 volt. The ideal contact means a zero applied force. Fig. 8, Fig. 9 and Fig. 10 are showing the controller results. From these figures, there are five interest regions.

Region I. This region represents the fingers in free movement ($\gamma_1 = 1$ and $\gamma_2 = 1$).
Region II. This zone corresponds to the initial contact force due $\Delta d(k)$.
Region III. This zone is where the control stabilizes to reach the empirical instantaneous initial contact.
Region IV. In this part the ideal instantaneous initial contact is set (zero force and almost no attenuation and γ_1 is changed to 0.2 and γ_2 to 1.5).
Region V. In this region the ideal instantaneous initial contact is reached.
Once the ideal contact is reached, a force contact control can be applied since the transient phase has been exceeded. Note that Fig. 8 and Fig. 9 show significant change values with small displacements from the gripper fingers. This means that these results can be improved if the resolution of the gripper increases.

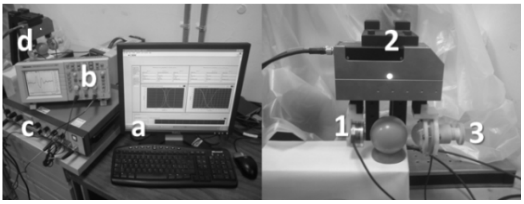


Fig. 7. At the left; a) The PC running the LabVIEW program; b) The oscilloscope; c) The pulser/receiver machine; d) The grip system. At the right: 1) The force contact sensor; 2) The WSG50 gripper; 3) Ultrasound sensor.

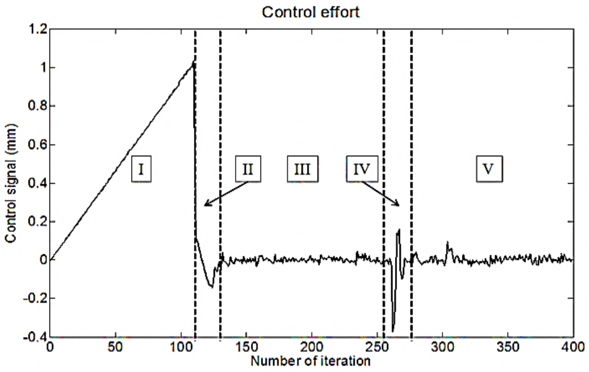


Fig. 8. Control effort of the MiFREN controller with linear consequence adaptation

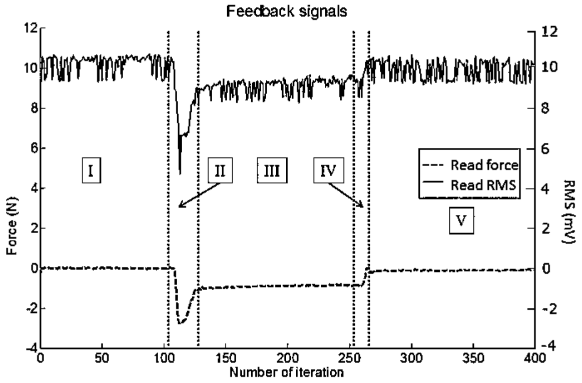


Fig. 9. Read force and RMS using the MiFREN controller with linear consequence adaptation

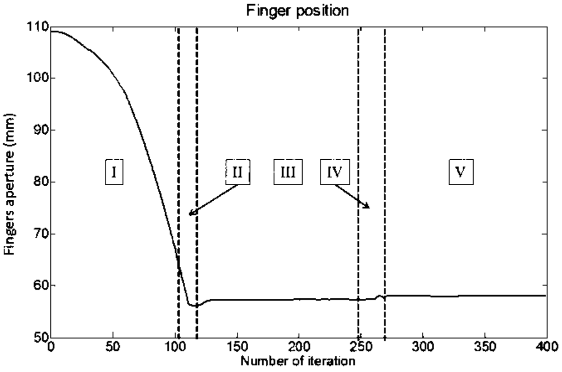


Fig. 10. Gripper fingers position

5 Conclusions

In this work, we have integrated a grip system conformed by a parallel electric gripper, an ultrasonic sensor and a force contact sensor. This system is controlled by a MiFREN controller that uses two different feedback signals. Also, a new cost function is proposed to perform the linear consequence adaptation according to the knowledge based on the system behavior and control stability was mathematically and numerically demonstrated. The system has ability to determine the instantaneous initial contact with no information about the mathematic model of contact or manipulator dynamics. This is done without any information about the object, or the use of computer vision algorithms. The experimental results show that transition between the free motion and the instantaneous initial contact can be obtained without bouncing or losing contact with the object. An important benefit of the MiFREN controller is the fact that the membership functions can be redefined according to the material properties of the test object, such as acoustic impedance, mechanical properties and others.

Acknowledgements. This work was sponsored by CONACYT grants SEP-CONACYT #134564 and #84791.

References

1. Volpe, R., Khosla, P.: Experimental Verification of a Strategy for Impact Control. In: Proceedings of the 1991 IEEE International Conference on Robotics and Automation, vol. 2, pp. 185–1860 (1991)
2. Tarn, T.J., Wu, Y., Xi, N., Isidori, A.: Force Regulation and Contact Transition Control. IEEE Control Systems 16, 32–40 (1996)
3. Youcef-Toumi, K., Guts, D.A.: Impact and Force Control. In: Proceedings of the 1989 IEEE International Conference on Robotics and Automation, vol. 1, pp. 410–416 (1989)
4. Alkkiomaki, O., Kyrki, V., Kalviainen, H., Liu, Y., Handroos, H.: Smooth Transition from Motion to Force Control in Robotic Manipulation Using Vision. In: 9th International Conference on Control, Automation, Robotics and Vision, ICARCV 2006, pp. 1–6 (2006)
5. Book, W.J., Kwon, D.: Contact control for advanced applications of light weight arms. Journal of Intelligent & Robotic Systems 6, 121–137 (1992)
6. Armendariz, J., Treesatayapun, C., Baltazar, A.: Force feedback controller based on fuzzy rules emulated networks and Hertzian contact with ultrasound. Mechanical Systems and Signal Processing 27, 534–550 (2011)
7. Armendariz, J., Treesatayapun, C., Baltazar, A.: Development of an Estimated Force Feedback Controller Based on Hertzian Contact and Ultrasound. In: 2010 IEEE International Robotic and Sensors Environments (ROSE), pp. 1–6 (2010)
8. Armendariz, J., Treesatayapun, C., Baltazar, A.: Determination of Surface Mechanical Properties Using a Hertzian Contact and Ultrasound Sensor. In: 2010 IEEE Ultrasonics Symposium (IUS), pp. 1446–1449 (2010)
9. Walker, I.D.: The use of Kinematic Redundancy on Reducing Impact and Contact Effects in Manipulation. In: Proceedings of the 1990 IEEE International Conference on Robotics and Automation, vol. 1, pp. 434–439 (1990)
10. Treesatayapun, C.: Fuzzy rules emulated network and its application on nonlinear control systems. Applied Soft Computing 8, 996–1004 (2007)

Integration of Directional Smell Sense on an UGV

B. Lorena Villarreal, Christian Hassard, and J.L. Gordillo

Center for Intelligent Systems, Tecnológico de Monterrey, Monterrey, México
{lvillarreal.bg,christian.hassard}@gmail.com,
jlgordillo@itesm.mx

Abstract. The olfaction sense in animals has been exploited on some applications like chemicals detection and during search and rescue operations. There has been considerable attention to bring this capability to mobile robots implementing odor source localization techniques. The objective of this research is to use a robot with a smell sensor inspired by nature, to identify the direction from where the odor is coming. The design of the sensor consists in two bio-inspired nostrils, separated by a septum, integrating a full nose system with the ability of inhalation and exhalation, which helps to desaturate the sensors. It has been implemented with a bus architecture based on Controller Area Network (CAN) to make the integration of the sensor relatively fast and without need of interfering with the existing systems of the vehicle. After several experiments, we conclude that the designed sensor can reach its objective even in an outdoor environment.

Keywords: Unmmaned vehicle, bio-inspired nose, smell sense, odor source localization, CAN bus.

1 Introduction

Smell is an important sense in the animal kingdom as it is used for inspection, recognition, mating, hunting, and others. In humans, our smell capabilities are much less acute than those of dogs, bears and rats for example. Due to this ability, dogs are commonly used in search and rescue operations at disaster areas, at airports or border controls. In addition to their superior smell sense, they can be trained to search for drugs, explosives, some chemicals or hazardous substances or even lost people.

Nowadays the technology is developing very fast, so it is becoming possible to provide a more reliable and resilient solution to these needs. One promising application for a smell sensor would be in rescue robots that have been integrated with sensors, actuators and algorithms that permit them to act with some degree of autonomy and are applied to disaster areas where it would be too risky for a human team safety. The algorithms used by these robots are most of the times based on common sensors like sonars, static or mobile cameras, lasers and some actuators [1], but despite the increasing trend of biologically inspired systems, in mobile robotics olfaction has not been extensively implemented.

The main characteristics of commonly used smell sensors are the discrimination between odors and the sensitivity. This is why most smell sensors are usually in a fixed position for food quality control, environmental analysis, and others, but the

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 273–284, 2013.

most important task of a mobile sniffing robot is the odor source localization. To achieve this, and inspired by nature, the smell system requires the capability of directionality to know from where the odor is coming, which only can be achieved by measurement strategies, in other words, applying artificial intelligence techniques.

In addition to discriminating an odor, analyzing the sensitivity achieved for that odor and implementing the algorithms to find the direction of the source, a problem shared by all sensory systems is that they must function in real world. The intention of this research is to analyze and justify that a single robot can find an odor source in an indoor or outdoor environment by using a biologically inspired stereo nose system.

One of the difficulties of developing chemical sensors versus other sensors is that chemical reactions change the sensor, often in a way that is nonreversible. Therefore the system must be easy to maintain, meaning that the sensors should be exchangeable with minimum effort. In this implementation a ground vehicle instrumented with CAN bus is used to increase the robot adaptability and the robustness of the designed odor sensor modules. Thanks to the reliability and flexibility of CAN bus applied on unmanned vehicles [2], this allows us to easily replace any sensor if needed.

In the following section we present some related work and algorithms, then on the third section the analysis of odor behavior is presented followed by the analysis of the bio-inspired nose system and the implementation design in section 4. The experimental set-up and results are described in sections 5 and 6 respectively. On section 7 we present the proposed algorithmic solution for odor source localization and finally on sections 8 and 9 we present the future work and conclusions.

2 Related Background

There is some work about odor source localization techniques in mobile robots using different types of environments [3, 4, 5]. In presence of an air flow, the particles of odor are carried by it, a behavior we are going to analyze in the following section. When this happens, to follow the odor trail and the air flow is a solution being developed by Lochmatter and Martinoli [3, 4, 6]. They are using a chemical sensor and different algorithms inspired by nature, as well as using the wind measurements to decide the best movement of the robot. However, using local wind measurements is not feasible in an unventilated indoor environment because the wind speeds present are usually too low [7] and can be mistaken as measurement noise. Besides, a sniffing robot preferably should not be dependent on the wind flow especially during search and rescue missions.

On the other hand, when no air flow is present, the use of concentration differences in the ambient determines the movement of the robot as in [5]. In this research, they are using a direct sensor-motor coupling in a differential robot, with a motor on each tire. The speed of each tire is influenced directly by the averaged concentration measured by a chemical sensor array. They are using as a source a device leaking alcohol. Using this reactive algorithm, the major concentration values were not usually reached where the odor source was located [7]. In this case, the leaking process generates different local maximum because the time of exposure is too long, which prevented a real estimate of the odor source position and direction.

A hound robot was presented by Loutfi et. al. [8] which discriminates odors and creates grid maps by convolving of the readings of the robot as it moves using the radial symmetric two dimensional Gaussian functions. They used three different robot movement techniques [9]: spiral, sweeping and instant gradient. Sweeping was the most efficient but the time to complete the task is too long, and a reactive algorithm was a little faster but most of the times gets stuck at a local maximum. A major drawback of this approach could be that the volatility of the odor particles causes them to be quickly distributed in the room so, when the robot finishes its routine, the distribution of the odor could be completely different.

Some other algorithms that include predefined airflow models and obstacle maps are described in [10][11]. The reader can refer to the work of Kowadlo and Rusell [12], to get more in-depth information about odor source localization methods.

3 Diffusion-Advection Property

Odor can be propagated without air flows present by diffusion or by advection depending on the laminar air velocity.

Diffusion is the process by which matter is transported from one part of a system to another as a result of molecular motions [13]. This process is non-reversible and increases the entropy as the chemical reactions take place.

The Fick's law for diffusion says that the mass of a solute crossing a unit area per unit time in one direction is proportional to the solute concentration gradient in that direction.

$$\bar{q} = -D\,\partial C/\partial x = -D\nabla C = \partial C/\partial t \tag{1}$$

Where q is a fluid flow, D is the diffusion coefficient which theoretically is a tensor, C is fluid concentration at certain measurement (x, or t).

The diffusion equation describes the transport of some kind of conserved quantity, in this case, odor concentration. In a two-dimensional x-y space

$$\partial C/\partial t = D(\partial^2 C/\partial x^2 + \partial^2 C/\partial y^2) \tag{2}$$

Advection describes the diffusion process in a fluid moving at a uniform velocity, u, which is constant in time. Now the total flux is composed by two independent flows. The total mass flux q_T crossing a unit area perpendicular to the flow direction will consist of convective uc and diffusive q_D fluxes, resulting on the following advection-diffusion equation:

$$\frac{\partial C}{\partial t} = D\left(\frac{\partial^2 C}{\partial x^2} + \frac{\partial^2 C}{\partial y^2} + \frac{\partial^2 C}{\partial z^2}\right) - \left(u_x \frac{\partial C}{\partial x} + u_y \frac{\partial C}{\partial y} + u_z \frac{\partial C}{\partial z}\right). \tag{3}$$

It is composed by the concentration and wind gradients. When the air flow is high the diffusion process is overwhelmed but it should not be neglected unless the distance was too short. When no air flow is present, the odor propagation will be radial as in a simple diffusion. To solve the odor source localization, considering both behaviors, we propose to use a reactive ascent method, as will be mentioned on section 7: "Proposed algorithm".

4 Nose System

To understand how a bio-inspired nose system works, it is necessary to keep in mind that odors are a mixture of many individual molecular components, that they are volatile and that smell is a single stimulus or sensation that occurs when these compounds chemically react with the olfactory sensorial system [14, 15]. This chemical reaction is then converted into an electrical or physical signal [16].

4.1 Sensor Model

When the odor, in this case alcohol, is detected, the MQ-3 gas sensor reacts; causing a change in the resistance between two terminals, which will give a voltage drop that is dependent on the alcohol concentration. This voltage *Vin* is the one that is measured and analyzed in [17] to obtain the best results. The concentration change (*% C*) was calculated as:

$$\%C = (differential\ \ range)/(Vmax - Reference) * 100 \tag{4}$$

Where *Vmax* is the maximum voltage ever measured, *Reference* is the lower voltage ever measured in absence of alcohol, and *differential range* is the difference between *Vin* and *Reference*.

4.2 Nostril Design

Smell sense in nature is divided in several stages [18, 19], which can be described as follows:

1. Aspiration process. The odor is carried in by an air flow being inhaled
2. Conduction. Then a good sample of the odor is conducted to the sensor.
3. Sensing. All the sensors react and send the signal.
4. Processing. This is the algorithmic stage of the smell sense.
5. Transforming. The air is transformed to clean the sensory system.
6. Exhalation. The expulsion of the air through the nose and ventilation of the system.

The design of the used nostril consists in an array of 3 sensors inside a chamber. All the biological stages are included in the nostril system developed, that is why only an homogenized sample of the environment is introduced into the chamber of the sensors. The complete nostril provides: an efficient and optimized measurement of the odor; an appropriate desaturation level; the homogenization of the air mix; and the sampling of environment.

4.3 Nose System

To complete the nose system we included a septum as in [13]. The odor molecules behave as light and image: the angle at which the wave is incident on the surface equals the angle at which it is reflected. Regardless if air flow is present, the septum represents the same obstacle for both nostrils. The final nose system is composed by a septum and two nostrils, each one with an array of 3 alcohol sensors.

The mechanical prototype of the sensor, shown in Fig. 1, helped to separate the odor molecules between the nostrils, while the inhalation process was useful to concentrate the odor molecules and the exhalation helped to dispose of the already sensed molecules cleaning the chamber to prepare the system for a new sensing cycle. Using this device, a desaturation of the sensors is accomplished, optimizing the measurement and making it more realistic.

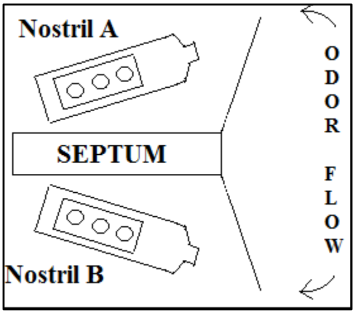


Fig. 1. Bio-inspired sensor system prototype

5 Experimental Set-Up

The nose system has been implemented as a module on a ground vehicle to provide a mobile platform for the odor source localization. The vehicle was originally designed to be driven by a human, so it has been instrumented with CAN bus, thanks to its proven reliability on commercial vehicles and on unmanned vehicles [20], while retaining the capacity of manual control. A microcontroller in the nose system module reads the nostrils voltages and puts the information in the bus while another module, a RF link, takes this information and broadcasts it to a near computer for further processing and analysis of the data. Figure 2 shows the architecture of the vehicle, where the nose system has been added as a new module and a picture of the nose in the vehicle.

To get closer in solving the odor source localization problem, we first need to prove that the implemented nose system will detect the direction from where the odor is coming in relation to the vehicle, due to a difference in the readings of the nostrils, even when at an outdoor environment.

To simulate the advection behavior an odor source was used. This odor source consists in a container with alcohol gel and an air entrance that generates an almost laminar flow with a fixed magnitude.

Several experiments were made to test the odor sensor implemented in the vehicle. Figure 3 shows the experimental set-ups.

First, as a control experiment, we made measurements in a closed environment with the vehicle in a fixed position while a directed odor source was aimed at the vehicle at several angles from the center of the sensor, as shown in Fig. 3a.

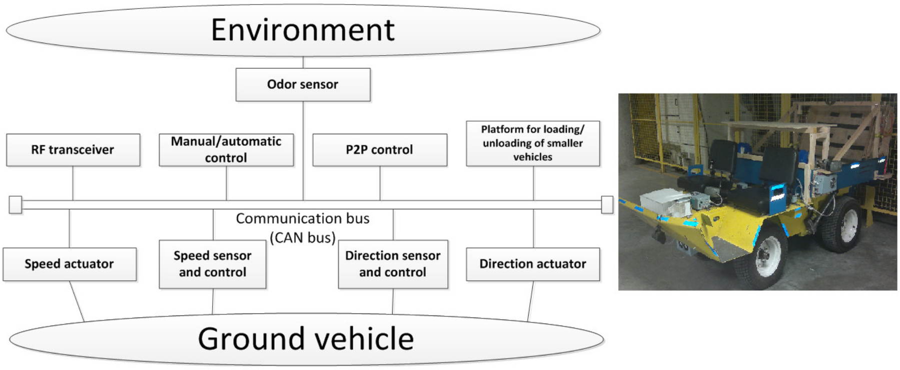


Fig. 2. Left: The vehicle architecture, using a distributed control philosophy through a network on the vehicle. Right: The nose system in the vehicle (white box in the front).

To test the performance of the nose system in a semi-outdoor environment, we took the vehicle to a hall where there was a small but constant natural airflow and repeated the test, using a directed odor source aimed at the vehicle at several angles, as shown in Fig. 3b.

Finally, we took the vehicle to an outdoor environment where we placed the odor source in a fixed position and drove the vehicle at certain orientations relative to the source, as shown in Fig. 3c and Fig. 3d.

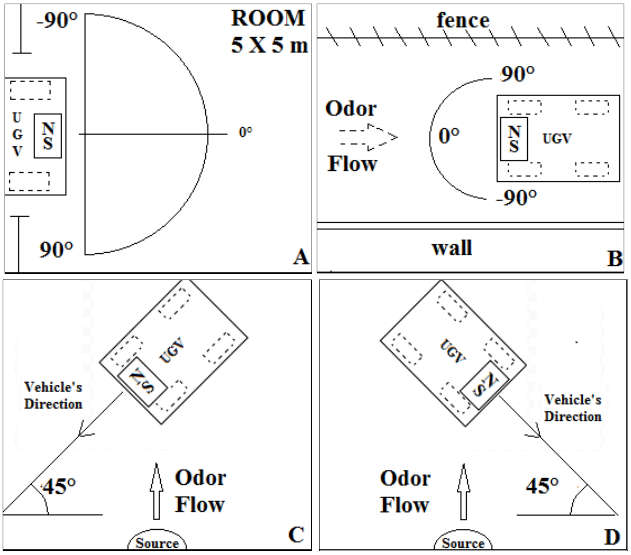


Fig. 3. Experimental set-ups used to test the odor sensor under different conditions: (A) in a closed environment; (B) with a laminar air flow; (C) and (D) outdoor tests at shown angles

6 Experimental Results

The biologically inspired nose system proved to have the ability to detect the direction from where the odor is coming. Three different experiments set-ups were used to prove it: controlled environment, laminar flow and outdoor environment. It is important to notice that when the nose system was not implemented the sensors do not respond with respect to direction of the source as expected and shown in Fig. 4.

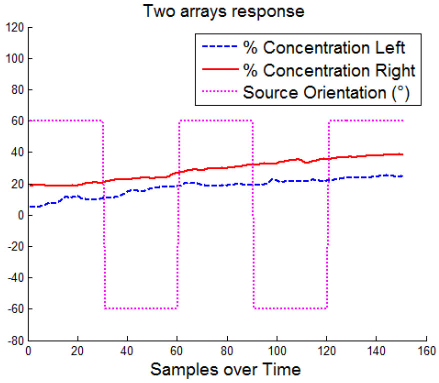


Fig. 4. Response when the biological nose system is not implemented

When the nose system is implemented the difference between nostrils was positive when the robot is facing to the right and negative when the robot is facing to the left in a controlled environment. Fig. 5 shows the measurements obtained from the nose system using the experimental setup shown in Fig. 3a.

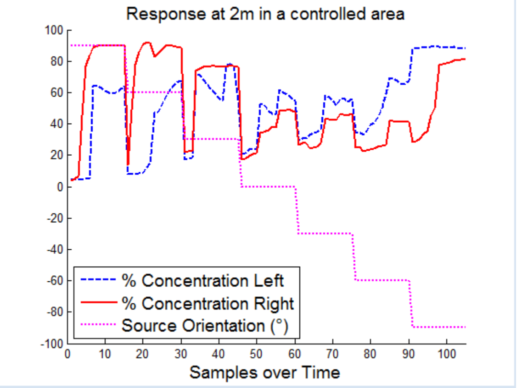


Fig. 5. Measurements obtained at 2m from the source and different angles (theta) during tests in a closed environment with minimal perturbations

We can notice that the nostrils would react with a different magnitude and at different times depending on the angle of the odor source. There is also a clear uncertainty at 0°, but this problem can be solved using a minimum threshold of the magnitude difference between nostrils, where the robot will have to follow a straight line. To compare the results, an error defined as the quantity of wrong perceptions of the source direction was analyzed using the sign of the difference between left and right measurements. In these results the error percentage without considering the uncertainty of the measurements when the robot is facing at 0° was around 3.3%, which means that the performance to discriminate direction was increased from not possible, as showed in Fig. 4, to 96.7% when the biologically inspired nose system was included.

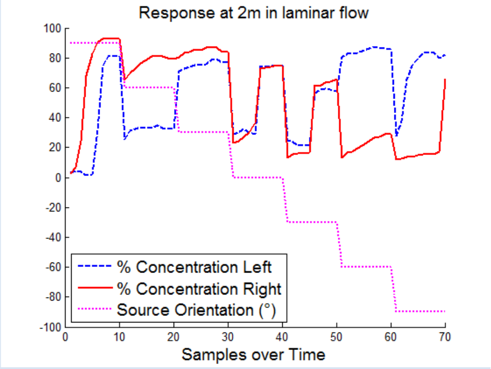


Fig. 6. Measurements obtained at 2m from the source and different angles (theta) during tests in a hall with a small natural air flow

Figure 6 shows the measurements gathered when the vehicle was on the experimental setup showed on Fig. 3b. In this case, as we had some small turbulent air flow, the sensor had more trouble to determine the direction of the odor when this was around 30 and -30 degrees, probably due to the mechanical design of the septum, which is currently being analyzed.

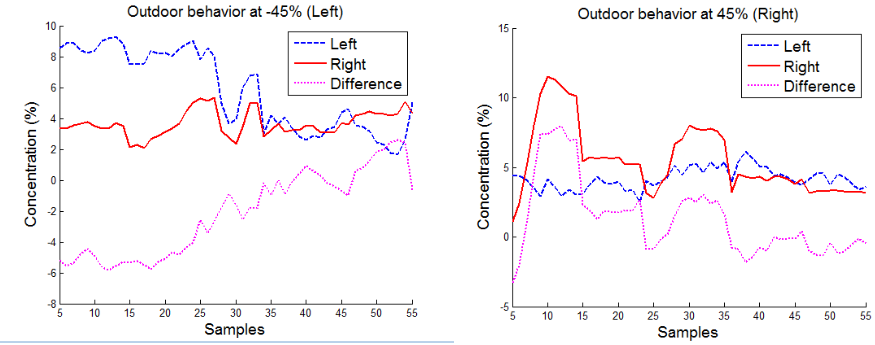


Fig. 7. Measurements gathered in an outdoors environment at 2m from the source and air currents of approximately 23km/h. Figure at left shows the results when the vehicle passed at 45 degrees from left to right, while figure at right shows the results when the vehicle passed at 45 degrees from right to left. The vehicle speed was close to 4km/h.

In the case of the outdoor experiments, even when we had turbulent wind currents of up to 23km/h and a moving vehicle, the sensor was still capable of determinate the direction of the source, although with less difference between the nostrils readings.

As we can see in Fig. 7, the first 35 samples where the most reliable and represent the moment when the vehicle passed over the source. At that point, the algorithm must demand to change the direction of the robot.

7 Proposed Algorithm

We will use a variation of the gradient descend method, which is a first-order optimization algorithm, which we will call reactive ascend method to find the odor source. To find a local minimum of a function using gradient descent, one takes steps proportional to the negative of the gradient (or of the approximate gradient) of the function at the current point [21].

```
given a starting point x ∈ dom $R^2$.
repeat
1. Determine a descent direction △x.
2. Line search. Choose a step size t > 0.
3. Update. x := x + t△x.
until stopping criterion is satisfied.
[Gradient descend method]
```

In reactive ascend method the system can estimate a direction based on the measurement differences between the nostrils, instead of sampling the environment. Using this, we can choose the step size and the direction that the robot needs to take by multiplying $\theta_T * k$, where k is a constant related to the maximum speed of the robot expected. Finally we can use these results to implement reactive ascend method to find the odor source.

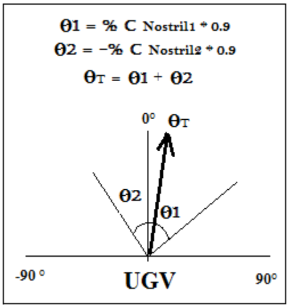


Fig. 8. Estimation of the robot heading using the nostrils measurements

As an example, Fig. 9 shows the direction estimate by the robot using the stereo nose system at different orientations of the source in a controlled environment, as well as the step size the robot would make on its orientation.

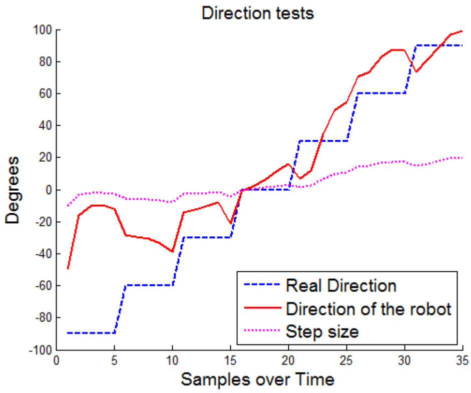


Fig. 9. Direction or heading estimate obtained by the robot perception

From Fig. 9, we can see that the robot would proportionally change its orientation to face the odor source. In this example the difference between left or right due to the prototype rough implementation can be compensated by an experimental *k*, independent for each nostril.

8 Future Work

In future work we will apply artificial intelligence and control techniques to improve the heading estimation of the robot, as well as performing the necessary experiments to prove that the reactive ascend method is good enough to find the odor source, using our bio-inspired physical design. The fundamental approach is to increase the efficiency in odor source localization techniques.

We'll use a gradient function for each nostril with respect to time using a buffer of historic positions and the methodology of mean squares to determine a vector for the next movement of the robot. We can also consider wind measures and subtract both results as in the advection-diffusion equation. These approaches can be reached using AI techniques giving a real treatment to the uncertainties of the environment.

As the vehicle used for these experiments has the capacity of carrying other smaller robots, we have been working in implementing this same nose system on a smaller tracked vehicle. This way, the bigger vehicle can detect an odor by advection while on a given route and then deploy the smaller robot that will finally find the source of the odor exploiting the diffusion of the odor, allowing the bigger robot to continue with its mission.

9 Conclusion

There are many applications where a sniffing robot can be used. This paper demonstrates that diffusion is present and that it can be measured when a constant gas leak is present, just as long as the leak is always incrementing the actual concentration at that point.

Thanks to the inhalation function and the complete nose system, we were able to concentrate the odor molecules near the sensors making the readings between nostrils to be different relative to the direction of the odor. And on the other hand, the exhalation function helps us in the desaturation of the sensors and makes them ready to the next measurement.

After the mentioned, and many other experiments we conclude that the designed odor sensor is capable of discriminating the direction an odor is flowing (right or left) with respect to the direction of the vehicle most of the time.

As we are currently working on a second nose system similar to the exposed in this paper, and thanks to the fact that we adopted the distributed control philosophy for the instrumentation of our vehicles, any nose system can be used on the vehicles that we have already instrumented. Another advantage that arises by instrumenting our vehicles with a network, where the nose system is a module, is that, if one of the nose systems fails in a vehicle and we have another nearby, we can simply unplug the defective module and plug the replacement without disrupting the rest of the modules, thus reducing the down time of the vehicle.

References

1. Shuzi, S.G., Lewis, F.L.: Autonomous Mobile Robots: Sensing, Control, Decision Making and Applications. Taylor & Francis Group, Florida (2006)
2. del Bosque, J., Hassard, C., Gordillo, J.L.: Velocity Control of an Electric Vehicle over a CAN Network. In: 2011 10th Mexican International Conference on Artificial Intelligence (MICAI), pp. 121–126 (2011)
3. Martinoli, A., Lochmatter, T., Raemy, X.: Odor source localization with mobile robots. Bulletin of the Swiss Society for Automatic Control 46, 11–14 (2007)
4. Martinoli, A., Lochmatter, T.: Theoretical analysis of three bio-inspired plume tracking algorithms. In: IEEE International Conference on Robotics and Automation (2009)
5. Lilienthal, A., Duckett, T.: Experimental analysis of smelling braitenberg vehicles. In: 11th International Conference on Advanced Robotics (2003)
6. Martinoli, A., Lochmatter, T.: Simulation experiments with bio-inspired algorithms for odor source localization in laminar wind flow. In: 7th International Conference on Machine Learning and Applications (2008)
7. Wadel, M., Lilienthal, A., Duckett, T., Weimar, U., Zell, A.: Gas distribution in unventilated indoor environments inspected by a mobile robot. In: 11th International Conference on Advanced Robotics (2003)
8. Loutfi, A., Coradeschi, S., Lilienthal, A., Gonzalez, J.: Gas distribution mapping of multiple odour sources using a mobile robot. Robotica 27, 311–319 (2009)
9. Lilienthal, A., Duckett, T.: Building gas concentration gridmaps with a mobile robot. Robotics and Autonomous Systems 48, 3–16 (2004)
10. Kowadlo, G., Russell, R.A.: Naive Physics for Effective Odour Localisation. In: Australian Conference on Robotics and Automation, Brisbane (2003)
11. Cabrita, G., Sousa, P., Marques, L.: Odor guided exploration and plume tracking: Particle Plume Explorer. In: Proc. of European Conf. on Mobile Robotics (ECMR), Örebro, pp. 165–170 (2011)
12. Kowadlo, G., Russell, R.A.: Robot odor localization: a taxonomy and survey. Int. J. Rob. Res. 27, 869–894 (2008)

13. Crank, J.: The mathematics of diffusion. Oxford University Press, New York (1976)
14. Pearce, T.C., Schiffman, S.S., Nagle, H.T., Gardner, J.W.: Handbook of Machine Olfaction: Electronic Nose Technology. Wiley-VCH, Weinheim (2003)
15. Doty, R.L.: Handbook of olfaction and gustation. Marcel Dekker, Inc., New York (2001)
16. Fraden, J.: Handbook of modern sensors physics, designs, and applications. Springer (2003)
17. Villarreal, B.L., Gordillo, J.L.: Directional Aptitude Analysis in Odor Source Localization Techniques for Rescue Robots Applications. In: 2011 10th Mexican International Conference on Artificial Intelligence (MICAI), pp. 109–114 (2011)
18. Craven, B.A., Paterson, E.G., Settle, G.S.: The fluid dynamics of canine olfaction:unique nasal airflow patterns as an explanation of macrosmia. J. R. Soc. Interface 7, 933–943 (2009)
19. Barrett, K.E., Barman, S.M., Boitano, S., Brooks, H.: Ganong's Review of Medical Physiology. McGraw-Hill, New York (2010)
20. Gurram, S.K., Conrad, J.M.: Implementation of CAN bus in an autonomous all-terrain vehicle. Proceedings of IEEE, 250–254 (2011)
21. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, New York (2009)

Automatic 3D City Reconstruction Platform Using a LIDAR and DGPS

Angel-Iván García-Moreno[1], José-Joel Gonzalez-Barbosa[1],
Francisco-Javier Ornelas-Rodríguez[1], Juan-Bautista Hurtado-Ramos[1],
Alfonso Ramirez-Pedraza[1], and Erick-Alejandro González-Barbosa[2]

[1] CICATA, Instituto Politécnico Nacional
Cerro Blanco 141 Col. Colinas del Cimatario, Quéretaro
jgonzalezba@ipn.mx
[2] Instituto Tecnolgico Superior de Irapuato-ITESI

Abstract. In this work an approach for geo-referenced 3D reconstruction of outdoor scenes using LIDAR (Light Detection And Ranging) and DGPS (Diferencial Global Positioning System) technologies is presented. We develop a computationally efficient method for 3D reconstruction of city-sized environments using both sensors providing an excellent base point for high-detail street views. In the proposed method, the translation between consecutive local maps is obtained using DGPS data and the rotation is obtained extracting correspondant planes of two point clouds and matching them, after extracting these parameters we merge many local scenes to obtain a global map. We validate the accuracy of the proposed method making a comparison between the reconstruction and real measures and plans of the scanned scene. The results show that the proposed system is a useful solution for 3D reconstruction of large scale city models.

Keywords: LIDAR, DGPS, 3D reconstruction.

1 Introduction

Terrestrial 3D laser scanning is a powerful tool for the surveyor. 3D laser scanning technology has become beneficial alternative in the collection of as-built data for manufacturing plant and facilities virtual reconstruction and management, as forest inventory characteristics such as vegetation height and volume as well as diameter at breast height. Other applications such as 3D modelling, as-built surveys, documentation, restoration and reconstruction of objects, require automatic processing of massive point clouds to extract surfaces of the recorded objects. In this work we introduces an approach for geo-registered 3D reconstruction of an outdoor scene using LIDAR technology and DGPS. We develop a computational method for 3D reconstruction of city-sized environments using both sensors providing a good base point for high-detail street views. Thus, remote sensing integrated with geospatial procedures and efficient field sampling techniques promises a fundamental data source for ecologically, socially and economically sustainable on-the-ground management.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 285–297, 2013.

The integration of aerial laser wiht GPS / IMU orientation systems has been widely used since the mid-90s because it provides good quality results and take advantage on a comprehensive manner of LIDARs airbone characteristics [14].

However, terrestrial lasers haven't followed the same path and are rarely directly targeted by a GPS / IMU. The integration of terrestrial laser with GPS / IMU sensors was carried out under project Geomobil [19] the mobile mapping system developed in ICC.

Using ground-based laser for tridimensional reconstruction of urban environments has grown considerably, the challenge is to create three-dimensional products with a minimal human intervention in processing information. Many laser scanning systems based on land vehicles have been developed in recent years [19] [9] [10] [12].

In [19] they use a Riegl laser $Z - 210$ capable of collecting up to $10,000$ points / second, intensity and RGB values are collected. The laser has a rotating mirror that allows taking vertical profiles while a servomotor rotates the system horizontally. All these raw data are parameters for a spherical coordinate frame. The GAMS (GPS Azimuth Measurement System) system allows to emulate differential GPS, in this case two antennas are mounted on the vehicle, the correction is made almost instantly achieving a data accuracy of 0.006 m.

Some other works incorporate an inertial measurement unit (IMU) and high precision GPS as in [9], also other sensors to generate high quality 3D views such as high resolution cameras for texturing point clouds [4], by using visual odometry algorithms (e.g. RANSAC or 7-point Hartleys algorithm [11]) they can determine the displacement and orientation of 3D point clouds although GPS are not sending information, this process is called Pose from Video (PfV).

In [12] merging data from different sensors (3 lasers, 2 high resolution cameras, 1 RTK GPS and inertial system) is given in real time, as they mounted the platform. The data of the video cameras are time tagged by the TERRAcontrol system which is synchronized with the laser scanner data. The TERRAcontrol computer gets the actual time from the global navigation satellite system (GNSS) receiver and distributes a time pulse together with a time stamp to the sensors. The accuracy GPS position in the kinematics conditions is 3 cm.

Most recently in [7]authors propose point cloud processing techniques to generate 3D maps from data captured by a system of detection and measurement through light. In this work, the authors presents two principal results: 2D maps for autonomous navigation and 3D maps reconstruction of urban scenes. This method is based on two techniques of segmentation of planes, the first one for a quick extraction of the main plane (the floor) and the second for the extraction of other planes (walls) [20].

2 Multisensorial Data Fusion

To provide our system with an acceptable quality and accuracy, We set up a multisensorial platform which is composed of a LIDAR laser sensor and a centimeter-precision GPS [1] [2]. While the laser sensor will provide the platform with three-dimensional data, the GPS will give for each laser acquisition its location into the world.

The used sensors are:

DGPS The GPS is a ProMark3 with RTK technology (Real Time Kinematic) single frequency and uses a double constellation for performance (GPS + SBAS) which allows GNSS surveys. The accuracy is variable and depends on the type of survey that we are doing, for real-time surveys have fixed RTK horizontal accuracy of 1 cm, post-processed static surveys collect data coordinates with an horizontal accuracy of 0.005 m, 0.01 m vertical and azimuth in arc second, on the other side kinematic works have an accuracy of 0.012 m horizontally, 0.015 m vertically.

LIDAR Velodyne© has developed and produced a High Definition LIDAR (HDL) sensor the HDL-64E, designed to satisfy the demands for autonomous vehicle navigation, stationary and mobile surveying, mapping, industrial use and other applications. The Velodyne HDL unit provides 360-degree azimuth field of view and 26.5-degree elevation field of view, up to 15 Hz frame refresh rate, and a rich point cloud populated at a rate of one million points per second. The HDL-64E operates on a rather simple premise: instead of a single laser firing through a rotating mirror, 64 lasers are mounted on upper and lower blocks of 32 lasers each and the entire unit spins. This design allows for 64 separate lasers to each fire thousands of times per second, providing exponentially more data points per second and a much richer point cloud than conventional designs. The HDL-64E is rated to provide usable returns up to 120 meters.

3 Urban Environments Digitalization Using LIDAR Technology

Our sensor array consisting in a differential GPS and a LIDAR mounted on a vehicle, connected to computer which is synchronized with the internal GPS clock to achieve a straight forward correlation of information from both sensors, as shown in Figure 1. Then using OpenGL libraries through primitive geometrics, in our case points, we are able to produce the 3D scenes by joining several acquisitions taken in different positions of the same capturing path.

3.1 Data Processing

After uprising, we end with gross data of the two sensors, we the process the data from the GPS and we assign these coordinates to the archives of the point clouds acquired by laser sensor. The procedure is done through the collation of the acquisition time between both sensors [15]. Seen otherwise its take the time of acquisition of a LIDAR file and look in the concentrate of the GPS coordinates and assigned to the file for further processing. Using the Cristian algorithm [6] we sync the time of two sensors.

Through C++ developed program we attach the respective GPS coordinates to each point cloud. Sometimes the required time stamp is not in the database,

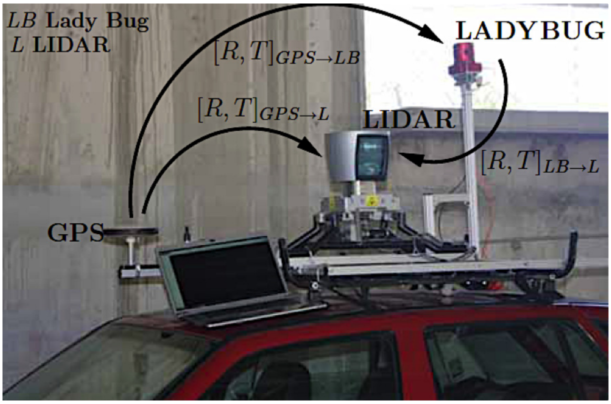


Fig. 1. Multisensor mobile platform

therefore we proceed to interpolate the positions that we do not have using a weighted interpolation:

$$\begin{aligned} \Delta_{gps} &= Gps_{n+1} - Gps_{n-1} \\ \Delta t_1 &= Gps_n - Gps_{n-1} \\ \Delta t_2 &= Gps_{n+1} - Gps_n \\ Missing\ Coordinate &= \frac{Coord_1 \ * \ t_1 \ - \ Coord_2 \ * \ t_2}{\Delta_{gps}} \end{aligned} \tag{1}$$

where Δ_{gps} is the weighting, Δt_1 y Δt_2 are the difference between the GPS acquisition time and missing time. This procedure is repeated for the longitude, latitude and elevation, which are independent data.

One of the algorithms used for data transformations from geographical coordinates to Euclidean coordinates, was the Coticchia - Surace algorithm [5]. The precision is one centimeter when using more than 5 decimals for all operations.

After coordinates are correlated with their respective acquisition of LIDAR sensor, we proceed to obtain the vectors of translation and rotation. Because the coordinates are in the UTM system, we obtain the translation vector with respect to a reference (x_0, y_0, z_0), this reference may be the position of an acquisition.

$$\underbrace{\begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}}_{\text{Translation vector}} = \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} \tag{2}$$

Where (x, y, z) denote the coordinates of the acquisitions, and (t_x, t_y, t_z) the translation vector between acquisitions and reference. So we get all translations for all acquisitions of the path to rebuild. To get the rotation first we define planes between two acquisitions and then rotate the second cloud on the first until matching the planes.

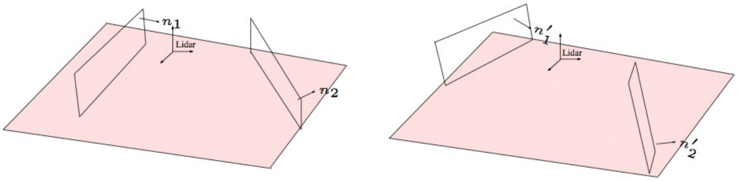


Fig. 2. Two acquisitions in different positions modeled by planes

Figure 2 shows two point clouds where their point of reference has been translated and rotated to match the environment planes. In [11], a plane referenced in two different coordinate systems is defined by the following equation:

$$\begin{bmatrix} n_i' \\ D' \end{bmatrix} = H^{-T} \begin{bmatrix} n_i \\ D \end{bmatrix} \tag{3}$$

Where $H = \begin{bmatrix} R & T \\ \mathbf{0}^T & 1 \end{bmatrix}$, R and T represents the rotation and translation matrix of 3×3 and 3×1 respectively. $\mathbf{0}$ denotes the null vector of 3×1. n_i and n_i' denote the normal unit vector of the planes referenced to the LIDAR coordinated system, which is not steady. D and D' denote the distance from the LIDAR point of reference to a plane in each point cloud. Therfore, the equation 3 is redefined as follows:

$$\begin{bmatrix} n_i' \\ D' \end{bmatrix} = \begin{bmatrix} R & \mathbf{0}^T \\ -(R * T)^T & 1 \end{bmatrix} \begin{bmatrix} n_i \\ D \end{bmatrix} \tag{4}$$

The resulting equations of two consecutive point clouds are used to calculate the rotation and translation between them. The equation 4 is redefined as:

$$F = \begin{bmatrix} n_i' \\ D' \end{bmatrix} - \begin{bmatrix} R & \mathbf{0}^T \\ -(R * T)^T & 1 \end{bmatrix} \begin{bmatrix} n_i \\ D \end{bmatrix} \tag{5}$$

R is given by Euler angles:

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_x) & -\sin(\theta_x) \\ 0 & \sin(\theta_x) & \cos(\theta_x) \end{bmatrix} \begin{bmatrix} \cos(\theta_y) & 0 & \sin(\theta_y) \\ 0 & 1 & 0 \\ -\sin(\theta_y) & 0 & \cos(\theta_y) \end{bmatrix} \begin{bmatrix} \cos(\theta_z) & -sin(\theta_z) & 0 \\ \sin(\theta_z) & \cos(\theta_z) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The *Levenberg-Marquardt's* algorithm is used to compute the rotation and relation from the equation 5 and is given by:

$$\frac{\partial F}{\partial \Gamma} = 0 \ ;$$

Where $\Gamma = \{\theta_x, \theta_y, \theta_z\}$.

3.2 Filtering

An inherent problem when working with lasers is the data acquisition noise, many factors such as environment, surface reflectance and the same sensor calibration are responsible for the digitization of an urban environment are noisy, that's why the information must be further processed to reduce this problem. The issue of filtering in 3D point clouds is fairly addressed in the state of art of the area [16] [13]. We used a version of principal component analysis (PCA) to maintain fine details without shrinking the data. The variant of PCA employed in this work makes a distribution of weights inversely proportional to the sum of the distances at which each data is from the average neighborhood ($V(p)$). Thus, more data are attacked far from the mean, so that outliers do not generate trend in this technique, we implemented the Eq. 6

$$W_i = \frac{1}{g_i \cdot \sum\limits_{j=1}^{n} \frac{1}{g_i}} \tag{6}$$

Where W_i is the weight factor for each point, g_i is the average distance from each point of the neighborhood and n the number of neighborhood data. A weighted averange ($\bar{p}_w$) of the points was calculated with the equation 7, once having the points and the weighted averange og their neighborhood we placed them in the covariance matrix for to the PCA defined by Eq. 8.

$$\bar{p}_w = \frac{\sum W_i \, p_i}{\sum W_i} \tag{7}$$

$$MC_w = \frac{1}{n-1} \sum_{i=1}^{n} (p_1 \; - \; \bar{p}_w)(p_1 \; - \; \bar{p}_w)^t \; W \tag{8}$$

where $W = \left\{\sqrt{W_1}, ..., \sqrt{W_n}\right\}$ are the weights associated with each point p_i the neighborhood $V(p)$. After the envelope to prevent aliasing of the data was applied to a moving average of the points $\bar{p}$ in direction to normal of the tangent plane to $V(p)$. The normal n_m is calculated using the third eigenvector of the covariance matrix MC_w [8]:

$$\bar{p'}_w = \bar{p}_w + t_{min} \, n_m \tag{9}$$

where $\bar{p'}_w$ is the new mean position, $\bar{p}_w$ original mean, n_m is the normal to tangent plane of neighborhood in $\bar{p}_w$ and t_{min} is a displacement calculated by Eq. 10.

$$t_{min} = \sum_{p_i \, \in \, V(p)} n_w \, \|p_i - \bar{p}_w\| \tag{10}$$

4 Results

The previous sections described the three-dimensional reconstruction platform, a Lidar Velodyne and a differential GPS provide the system of necessary data to generate urban scenes. The LIDAR manufacturer specifies an accuracy of ± 5 cm in the collected data, however a home-made calibration of the scanner and the development of our own capture software allow us to obtain an accuracy of ± 1.6 cm [8]. Laser generates point clouds of local urban scenes that represent a portion of the total path, acquiring approximately a million points per second, these clouds are the ones we need to merge at the end of post-processing to generate global maps. At the same time, GPS records its position at a rate of one recording per second.

Our tests were carried out in the urban area close the research center in which work is developed, in Querétaro, México. We have chossen this place because it contains a variety of suitable of urban scenes, such as buildings, parking lots, shopping centers, areas without building, etc., which allow us to have an appropriate feedback appropriate all possible environments that form an area urbanized cities.

Geodesic points were acquired in the WGS84 (World Geodetic System 84) system because is a standard for mapping and has been also defined by the INEGI as standard in the Mexican Federation. In addition to allowing us to maintain the accuracy of the data to manage the information, either to interpolate the positions or switch from geodetic coordinates to flat through the conversion of spherical coordinates to flat through Coticchia-Surace algorithm could know the displacement and calculate the translation vector necessary to merge the local maps and generate the global view.

One of the major obstacles in this work was the computational performance, as shown in Figure 3 each of the marks indicate a capture position of the LIDAR sensor, if each of them is composed by around a million points, we are talking about a large amount of information to be processed, although the virtual 3D allocation of point clouds is not the problem, its manipulation is a big one, since the algorithms must be repeated thousands of times to generate the desired global scene.

In a parallel way to reduce the system works data, also would be eliminating of the maps much of sensor noise captures and the ground points, therefore to find appropriate distribution between time and distances of acquisition will provide our system with the speed and fidelity of data appropriate. Figure 4 shows three point clouds merged implementing the transformation algorithm Coticchia - Surace for obtaining the displacement, difference in acquisition times between each of them is 14 *second* at a speed of 40 km/h approximately . Notice the *blind spot* of the LIDAR in the center of the figure, also the considerable amount of information around this.

Merging several local acquisitons, our global map grows in vision depth, but obviously the final file size containing all data transferred also grows, in Figure 5 seen as being defined shapes and details merge grow as clouds, filtering these clouds, increase details because the noise of the clouds is decreased and shapes become more clear. Using a top view details are not seen properly, in Figure 6 can be seen at higher resolution the objects in the environment, a long trajectorie allows almost a complete rebuild of all objects, this is because data blocked in one local map can be obtenined in the next, or two or more cloud map later,

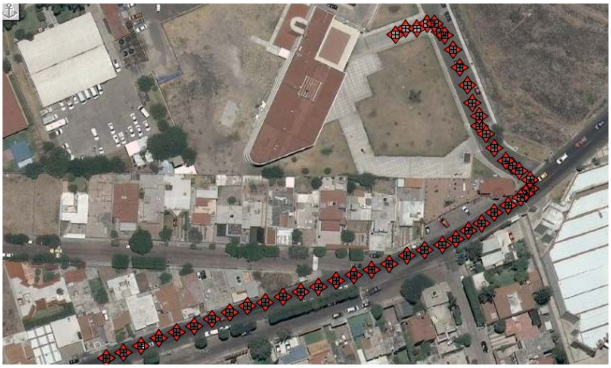


Fig. 3. LIDAR acquisitions

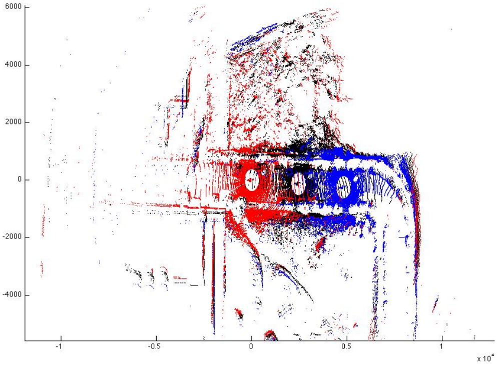


Fig. 4. Three merged data clouds, the displacement of the platform is left - right

mainly because the whole system is moving, allowing for multiple points of view of the same object. The previuos makes it possible to increase object definition and having a robust global map.

There are many other considerations in construction of global maps, for example in Figure 6, the material of certain objects tends to be relevant in the reconstruction, while dark colored objects absorb the laser intesity, light colors reflect almost the same intensity, and on the other hand, metallic objects tend to increase the intensity of the laser pulse and objects such as walls reduces the intensity, in this way the three-dimensional reconstruction with this technology denote implicitly these characteristics of the materials.

These results offer a clear perspective that terrestrial LIDAR technique is a viable option for the construction of three-dimensional urban scenes, where the external physical characteristics of the objects surface are not lost. We hope that by improving capture and fusion algorithms results can be also improved and, as a consequence precision can also be taken to better levels. As mentioned

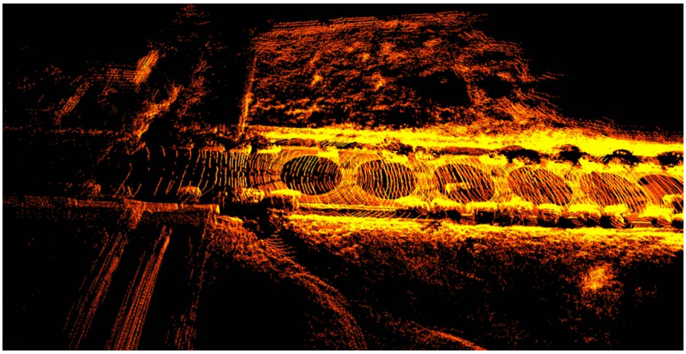


Fig. 5. Several local maps merged. The vision depth grows and objects resolution increase.

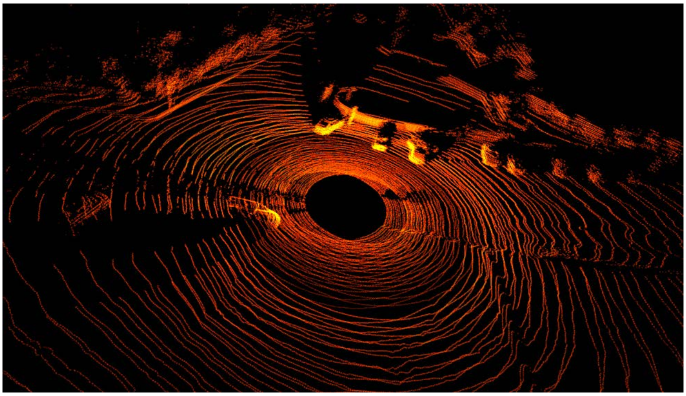


Fig. 6. Metal objects such as cars have higher reflectivity and perceived more defined

above, precision is now around 1.56 *cm* and its improvement will allow us for a closer-to-reality and more accurate global map.

4.1 Error Estimation Procedure

A controlled experiment was carried out using a closed-loop circuit in the shape of a parallelogram (Fig. 7). The idea of the path traced in this circuit is to have a good way for comparing results between our GPS dynamic acquisition and analysis system and reality. In addition a data filtering was carried out to basically erase all data referring the ground, which reduced the amount of procesed information.

It is important to know the error of our GPS system, to do so we used the closed-loop circuit as follows. We measured by hand the distances between each control point (see Fig. 7. Next we mounted the system on the vehicle and followed the circuit for a period of 30 minutes. Each control point (or *waypoint*)

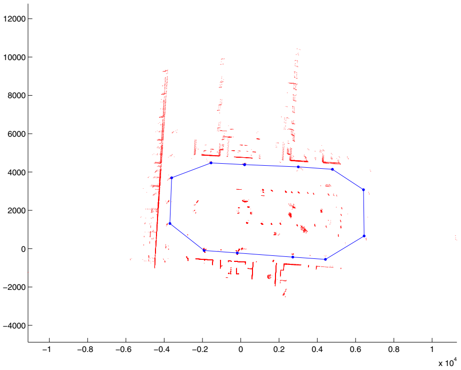


Fig. 7. Path control points for analysis accuracy

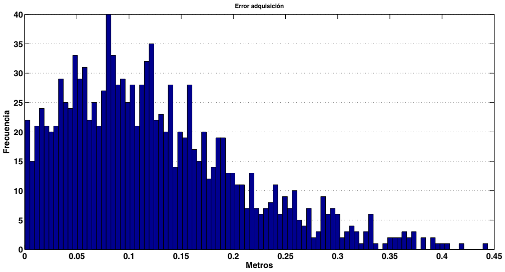


Fig. 8. Acquisition error, path vs. GPS data

location was acquired dynamically along the route. The final *waypoint* count was of around 2000 data points. Because the path described a parallelogram it was possible to obtain a set of equations describing each of its sides, notice that each line was defined by two consecutive *waypoints*. With this equations it was also possible to calculate the distances separating the *waypoints*. Finally, by comparing this calculated distances with those taken by hand (assuming the lasts were true) we were able of obtaining a good estimation of the average error in the dynamic capturing system. The average error ranged from 0 to 50 cm which we considered quite acceptable given the capture conditions (our capturing system is always in motion). See for example [18] and [3] where a traslation error of 30 and 12 cm was obtained with a static system.

Once a set of individual point clouds has been merged, error of fused data is also estimated. This estimation is necessary because the acquisition methodology induces a cumulative error from one capture to the next. By using a closed-path it is ensured that some objects and surfaces are captured in all or several 3D scenes, using this objects as references an error evaluation is possible. Error estimation procedure is described next. We started with the definition of some specific straight lines belonging to some walls, not all the lines but only those visible from an aerial view (we called them reference lines) as they will be seen in an architectural plane, to create this reference lines we made a line fitting of all points defining a wall used as reference. The fitting is needed because the Lydar has an intrinsic error an not all points assigned to a flat object lie in the same plane. The next step is to trace a perpendicular line to each of the reference lines, this perpendicular must also pass through a reference point (those points on parallelogram vertices). Finally we compare the perpendicular position resulting from one capture to the perpendicular position of a different capture. Remember that we have obtained multiple captures by doing the same path several times. After comparison we are in the position of obtaining a good estimation of the cumulative error induced in our readings due to the error on GPS positioning. Cumulative error is shown in Fig. 9 and 10 were one can see that the error is bigger towards the end of the loop.

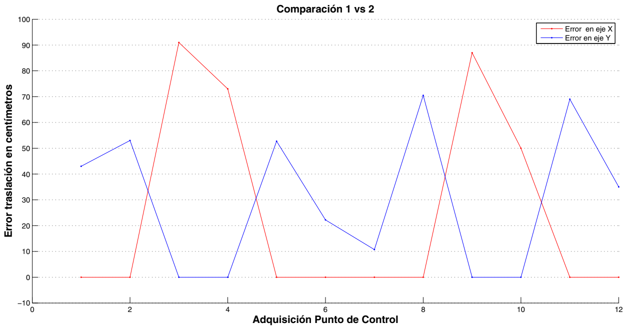


Fig. 9. Accumulated error between the first and second acquisition in the control circuit

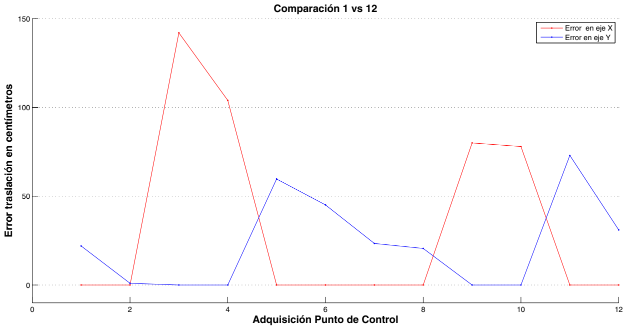


Fig. 10. Accumulated error between the first and last acquisition, completed loop

5 Conclusions and Future Work

An automatic 3D building system requires the least extra effort by users and the equipment used during the procurement field. The accuracy of the information processing should be the most optimal, but now with terrestrial systems, 3D reconstruction is rarely enough by many factors such as noise, viewing angles, speed of which is mounted mobile platform acquisition and performance of the hardware used.

We have presented experimental results of 3D city construction of a system consisting of a Velodyne LIDAR 64E and a differential GPS. Our system can acquire data for several kilometers in real time and fast processing of information to generate accurate three-dimensional scenes.

Among the additional work presented in this paper are debugging and generate georeferencing algorithms and interpolation of data faster and robust, as well as conversion to other flat coordinates by more elaborate processes [17]. Improving the quality of global scenes generated by a most complete calibration of the sensors and improve the automation of data processing tasks. We plan to implement a 3D SLAM algorithm. It is also in our plan to extend partially the Orthogonality to outdoor city-like environments where the constraint of vertical planes holds for most buildings. Alternate work this would complement the view of the scene, as the segmentation of objects not belonging to the environment and the texturing of the point cloud.

Commonly used methods in this field depend of video elements and photogrammetric that provides accurate works but with long time post-processing, that is why the integration of our system will drastically reduce the time to obtain efficient 3D scene point clouds to obtain quickly and a low cost of operation.

This demonstrates that a terrestrial technique for a LIDAR technology can be considered a new alternative rather than traditional methods such as LIDAR and aerial photogrammetry for three-dimensional reconstruction of urban environments.

References

1. Amoureus, L., Bomers, M.P.H., Fuser, R., Tosatto, M.: Integration of lidar and terrestrial mobile mapping technology for the creation of a comprehensive road cadastre. In: The 5th International Symposium on Mobile Mapping Technology, pp. 29–31 (2007)
2. Böhm, J., Haala, N.: Efficient integration of aerial and terrestrial laser data for virtual city modeling uusing lasermaps. Proceedings of the ISPRS Workshop, Laser scanning (2007)
3. Bok, Y., Hwang, Y., Kweon, I.S.: Accurate motion estimation and high-precision 3d reconstruction by sensor fusion. In: Optimization, pp. 4721–4726 (April 2007)
4. Chen, L.C., Teo, T.A., Rau, J.Y., Liu, J.K., Hsu, W.C.: Building reconstruction from lidar data and aerial imagery. In: International Geoscience and Remote Sensing Symposium, vol. 4, pp. 2846–2849. IEEE (2005)
5. Coticchia, A., Surace, L.: La trasformazione delle cordinate dal sistema UTM. Laboratorio di Topografia e Fotogrammetria (1984)

6. Cristian, F.: Probabilistic clock synchronization. Distributed Computing, 146–158 (1989)
7. de Jesus Rico-Jiménez, J.: Construcción de mapas 3d a partir de la extracción de primitivas geométricas obtenidas de datos de un lidar. Tesis de Maestría (2009)
8. González-Barbosa, J.-J., Atanacio-Jiménez, G., et al.: Lidar velodyne hdl-64e calibration using pattern planes. International Journal of Advanced Robotics Systems (2011)
9. Grinstead, B., Koschan, D.: Vehicle-borne scanning for detailed 3d terrain model generation. In: SAE Commercial Vehicle Engineering Congress (2005)
10. Haala, N., Peter, M.: Mobile lidar mapping for 3d point cloud collection in urban areas - a performance test. In: 21 ISPRS Congress (2008)
11. Hartley, R., Zisserman: Multiple view geometry in computer vision. Cambridge University Press (2000)
12. Kremer, J., Hunter, G.: Performance of the streetmapper mobile lidar mapping system in "real world" projects. Photogrammetric Week (2007)
13. Leal, E., Leal, N.: Point cloud denoising using robust principal component analysis. In: International Conference on Computer Graphics Theory and Applications, pp. 51–58 (2006)
14. Lindenberger, J.: Laser-Profilmessung zur topographischen Geländeaufnahme. PhD thesis, Deutsche Geodätische Kommission bei der Bayerischen Akademie der Wissenschaften (1993)
15. Mettenleiter, M., Obertreiber, N., Härtl, F., Ehm, M., Baur, J., Fröhlich, C.: 3d laser scanner as part of kinematic measurement systems. In: 1st International Conference on Machine Control and Guidance, pp. 24–26 (2008)
16. Mitra, N.J., Nguyen, A., Guibas, L.: Estimating surface normals in noisy point cloud data. Special Issue of International Journal of Computational Geometry and Applications 14, 261–276 (2004)
17. Snyder, P.: Usgs no. 1532. Map Projections used by the United States Geological Survey
18. Sprickerhof, J., Lingemann, K., Hertzberg, J.: A heuristic loop closing technique for large-scale 6d slam. Automatika, 199–222 (2011)
19. Talaya, J., Bosch, E.: Geomòbil: the mobile mapping system from the icc. In: 4th International Symposium on Mobile Mapping Technology (2004)
20. Verma, V., Kumar, R., Hsu, S.: 3d building detection and modeling from aerial lidar data. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2213–2220. IEEE (2006)

Towards the Creation of Semantic Models Based on Computer-Aided Designs

Nestor Velasco Bermeo[1], Miguel González Mendoza[1], Alexander García Castro[2], and Irais Heras Dueñas[3]

[1] ITESM Campus Estado de México, Computer Science Department
[2] Florida State University
[3] ITESM Campus Monterrey, Design and Technology Innovation Center
{nestorvb,mgonza,iraisheras}@itesm.mx, alexgarciac@gmail.com

Abstract. Nowadays one of the biggest problems many manufacturing companies face is the loss of knowledge from the information it possesses. Whether it tries to make business or improves the exchange of information within its different areas, valuable knowledge does not reach all stakeholders due to abstraction and ambiguity. A clear example in which both problems have a clear effect in terms of knowledge loss occurs during the interpretation of Computer-Aided Designs (CAD). If there is no experience doing such task the only data extracted will be limited to the elements contained on the drawing. By creating a semantic model we are able to know the contents specific details of a CAD without the use of a graphical tool, also ambiguity problems disappear as the terms used on the semantic model are based on a controlled vocabulary derived from an ontology.

Keywords: Semantic Design, Ontologies, Survey, CAD, Computer-Aided Design.

1 Introduction

Large or medium sized companies face a wide and well known problem nowadays, loss of knowledge among the company. Data is spread across different sources, consolidated databases, Enterprise Resource Planning, Customer Relationship Management, Human Capital Management, Supply Chain Management, etc. Every time a new product is about to be launched on the market a lot of information is generated, and so a need of having a coherent and seamless organizational system is essential to avoid the loss of data. All stakeholders or departments might have access to the information available but that doesn't imply that they have the expertise or background to interpret properly what's stored. And such problems related to the proper understanding of the meaning and implications of such data are big tradeoffs for the company.

All above considerations might be translated into interoperability problems. As stated by the National Institute of Standards and Technology (NIST) “Interoperability is essential to the productivity and competitiveness of many industries. Premised on a

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 298–310, 2013.

reliable digital representation of product and process data coordinated by many different participants and processes, interoperability is necessary for efficient design, production and Supply Chain Management (SCM)" In terms of supply chain costs, poor information visibility translates in unnecessary inventory. Wilson (2006) wrote that logistics costs increased 15.2% from 2004 to 2005, such increase was mainly because of the cost of carrying inventory (which increased by 61 billion dollars in the United States).

In the 2004 NIST's planning report "Economic Impact of Inadequate Infrastructures for Supply Chain Integration" the total costs estimated of inadequacies in supply chain infrastructure where of 5 billion dollars per year for the automotive industry, as of almost of 3.9 billion dollars per year for the electronics industry [1]. The inability to seamlessly and automatically integrate business information is the main aspect of all issues; a $1 reduction in costs from supply chain deficiencies is equivalent to a $12 increase in sales revenues [2].

The data shared across the different systems may be complete but interpretation problems arise when people try to understand the information provided. This is due to the different terminology and representations each particular domain possesses, misinterpretations occur easily when the semantics associated with the terms are not properly defined. If we add both effects of inefficient interoperability and the lack of semantics between the different departments it is easy to understand how the data exchange process can easily get complicated and have considerable negative effects and elevated costs. Whenever a company works on the creation of a new product the data exchange is essential and implicates the participation of different departments (sales, engineering, manufacturing, support, etc).

If a company creates a new product one of the most cost effect stage is the design phase, most of the costs are due to a vast number of design iterations before the product is placed on the stores for its sell. The most common problem behind such iterations is based on the fact that the product design doesn't fulfill all the requirements depicted on the Product Design Specification (PSD) document. The PSD describes how the device will be marketed, produced, distributed, how to be maintained and disposed of. According to Magrab "*the PDS contains all the facts relating to the product's outcome. It is a statement of what the product has to do and is the fundamental control mechanism and basic reference source for the entire product development activity*" [3]

The PDS is a way to ensure that every member of a design team works to reach the same goal [4]. The engineering team in order to create a detailed model of the product generates a series of elements to meet the PSD high level requirements. The model created includes information about the parts and assemblies as well as how they work together in order to satisfy the requirements of the PSD but there is no evidence of how the elements of the model satisfy the PSD requirements.

In order to be more competent on the product development process companies use collaborative product development as its core strategy because it increases efficiency, reliability and innovation, all by ensuring that the right knowledge is available at the right time [5]. Integrated and concurrent design eases the interaction and agreement between engineers but most important the proper share of knowledge and product information is guaranteed reducing the cost of design as well as an increase in reliability. But this strategy is not error-proof since the most common factors affecting collaborative product development are related to the expressiveness and clearness of the documents that the team create and share (product descriptions sometimes are partially created and independently done by different designers as well as engineers and tools interpret the same product description in different ways), shared representations are one of the several areas affecting collaborative engineering, including long term knowledge retention. [6]

Ambiguity and Abstraction are two of the most common problem a company has to deal with (whether it tries to make business with another company or improves the share of information within the different areas), abstraction is the intentional omission of information that the users don't explicitly need, whereas in case of ambiguity the omission is unnoticed and the information omitted is useful. [7]

In terms of engineering drawings the above conflicts occur repeatedly, while working or having a final drawing of a part some information is obvious for the designer and engineers with design knowledge but for a common user there are key aspects omitted based on the abstraction the drawing possesses. In this same example ambiguity comes in hand if the drawing does not fulfill the proper drawing standards[1], for example if the notation of a thread is not properly described production problems occur such as improper fit and hence damages in the part and the counterpart or the need to rebuild a whole assembly.

In order to avoid the above mentioned problems a semantic model is proposed to be able to get extra information beyond the geometrical and graphical representation of a design. The fundamental idea is to generate a semantic model using a controlled vocabulary from an ontology and to be able to know the contents of a CAD drawing without a formal CAD tool to visualize and analyze the file.

In this article we present the implementation of a tool to extract the contents of a CAD drawing in order to create a semantic model that uses terms derived from an ontology. The article is organized as follows: Section 2 presents related work on semantic tools used to extract information of CAD files. In Section 3 use cases and a case example is presented. Section 4 presents the approach; and finally in Section 5 we discuss the findings and the future work.

2 Related Work

This section summarizes related work on extraction tools from CAD files mainly focusing on feature recognition and reuse and ontology use and development.

[1] The governing agency responsible in setting the standards is the American Society of Mechanical Engineers having the following standards referring to mechanical drawings; ASME Y14.100, ASME Y14.4M, ASME Y14.3M, ASME Y14.1, ASME Y14.5M, ASME Y14.13M.

2.1 Feature Recognition and Reuse

Min Li, Y. F. Zhang and J. Y. H. Fuh proposed a novel method to retrieve reusable CAD models using knowledge-driven partitioning based on modeling dependency graphs. In order to retrieve reusable results, two dependency-graph partitioning are given based on design knowledge. The first is a horizontal partitioning scheme to simplify CAD models and preserve their essential shapes. In order to support partial shape retrieval and reuse, another vertical partitioning segments sub-parts in a meaningful way. The main contributions include; graph-based modeling knowledge representation, knowledge-driven graph partitioning strategies for improving reusability of retrieved results and redesign supports by utilizing modeling expertise.

Žáková, Železný et al. focused on contemporary product design based on 3D CAD tools which aim at improved efficiency using integrated engineering environments with access to databases of existing designs, associated documents and enterprise resource planning (ERP). The SEVENPRO project aimed to achieve design process improvements through the utilization of relational data mining (RDM), utilizing past designs and commonly agreed design ontologies. The discussion centers on the applicability of state-of-the-art ILP systems to the RDM tasks in this application domain as well as the implied challenges to ILP research and system implementations.

2.2 Ontology Use and Development

Dokken, Quak and Vibeke discuss about how NURBS (Non Uniform Rational B-Spline) curves and surfaces have been used extensively in Computer-Aided Design in the last two decades but not in Finite Element Analysis. Isogeometric analysis, the main element centers on how traditional Finite Elements may be replaced with volumetric NURBS and shows how although mathematically this seems to be a minor adjustment, it will drastically change the model life-cycle in finite element analysis. Consequently product design ontologies addressing FEA, such as AIM@SHAPE Product Design Ontology have to be revised to include the concepts of isogeometric analysis. The work presents the analysis and recommendations in how to adapt such ontology to fit the new proposal of the inclusion of NURBS.

Catalano, Camossi et al. discuss on how essential effective and efficient information management has become in terms of the Product Development Process. The use of ontologies how they offer new possibilities for representing, handling and retrieving product related knowledge, and for online collaboration. A Product Design Ontology (PDO) is presented; such ontology especially covers the need of sharing industrial product design and engineering analysis as well as shape data in order to develop software tools. Specifically they formalized the task-specific information associated to a shape, and the functionality and usage of shape processing methods in specific tasks of the design workflow. The PDO presented, based on the ontology-driven metadata of shapes and tools, and seems promising for large or medium sized companies that face interoperability problems mainly when trying to retrieve shape-related information and for supporting benchmarking of processing tools and gathering the knowledge about shape processing workflows.

Leimaignan, Siadat et al. present a proposal for a manufacturing upper ontology, aimed to draft a common semantic net in manufacturing domain. They present the usefulness of ontologies for data formalization and sharing, especially in a manufacturing environment. Details are given about the Web Ontology Language (OWL) and its adequation for ontologies in the manufacturing systems. A concrete proposal named MASON (MAnufacturing's Semantics ONtology) is presented and two applications of this ontology are exposed: automatic cost estimation and semantic-aware multiagent system for manufacturing.

Decelle, Gruninger et al., present a view of the current status of manufacturing information sharing using light-weight ontologies and discuss the potential for heavyweight ontological engineering approaches such as the Process Specification Language (PSL). It explains why such languages are needed and how they provide an important step towards process knowledge sharing. Machining examples are used to illustrate how PSL provides a rigorous basis for process knowledge sharing and subsequently to illustrate the value of linking foundation and domain ontologies to provide a basis for multi-context knowledge sharing.

If a company A belongs to a Supply Chain (SC) and is in charge of the design of a part or a complete product there must be a way to make sure the information exchange among all stakeholders is seamless and clear. That will avoid costs within the Supply Chain and reduce production times. The mere sharing of information between engineering design systems and manufacturing systems within a SC does not represent a real integrated system. Supply Chain Integration, must focus on exchange of info between logistics and providers as well between designers and fabricators.

As mentioned before every time a new product is created a PSD is generated as well, but such document does not go beyond the design stage and the further areas involved in the creation of such product hardly have access to it so they rely completely on the CAD file they received and the printed versions of such designs. The exchange of information is reduced to merely transferring CAD files within the participants and involved areas (Design and Manufacturing mainly).Whenever there's a "problem" or question related to the design they have to call the designer and ask questions such as "why did you chose that material?, Is there any particular reason part n is oriented as stated on the drawing?, can we change that?"; the answers given sometimes lack of context and use terms that might not really mean what its intended; "they told me that the cover had to be hard and scratch resistant" but that doesn't give us all the proper information for further use or to solve any future conflict. In the end there is no additional information or description about a part or assembly other than the annotations, notes or explicit information shown in the drawing.

In order to be able to express concepts used in any engineering application or document in an effective way such concepts must be expressed in a language that is highly expressive but at the same time free from imprecision and ambiguity problems (common problems associated with natural language). Here is where the use of ontology becomes essential.

According to Thomas Gruber's definition [13], an ontology is the statement of a logical theory. Even though an ontology can define and specify the meaning of terms there are a lot of ontologies that can be classified based on their degree of formality in the specification of meaning. Based on such criteria we may have an ontology that may be a simple taxonomic hierarchy of classes without any constraints on the meaning of the terms it has.

3 Use Cases and Approach

3.1 Use Case 1: Simple Engineering Drawing

Engineering drawings range from simple part representations (Figure 1) with mere the minimal elements such as the main views (top, front, side view) and the notations of a section view but lacks of information such as dimensions, orientation, material, units, etc. to complex ones.

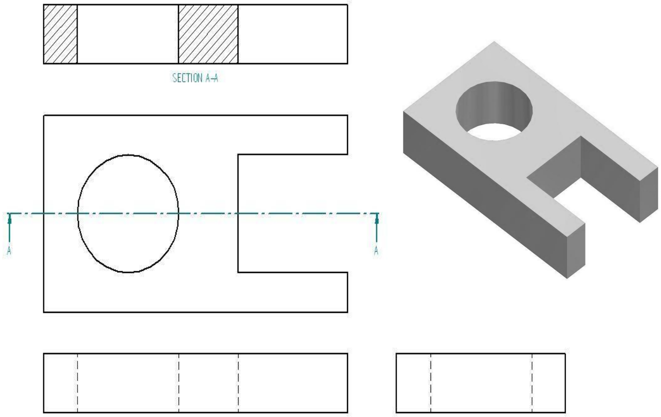


Fig. 1. Simple CAD drawing of a part

In this case the user (a mill operator) faces the problem of having in front of him a paper design that as noted on Figure 1, lacks of critical details such as dimensions, materials, scale, etc. and has the need to know details that are embedded within the CAD file but doesn't have a CAD software at hand to access the file nor the name of the file referring to the drawing in front of him since the graphical and geometrical representation does not explicitly answers his doubts because it's not possible to use a measuring tool (ruler, vermier, etc).

3.2 Use Case 2: Complex But Partially Annotated Drawing

In Figure 2 we see a full engineering drawing of a four cylinder, four cycle Morton M42 [14] [15] motor, with a clear description of each of the components of the motor and noted using the proper standards for the names and details of the drawing. But still important information is omitted as well as the part list is noted in a confusing way.

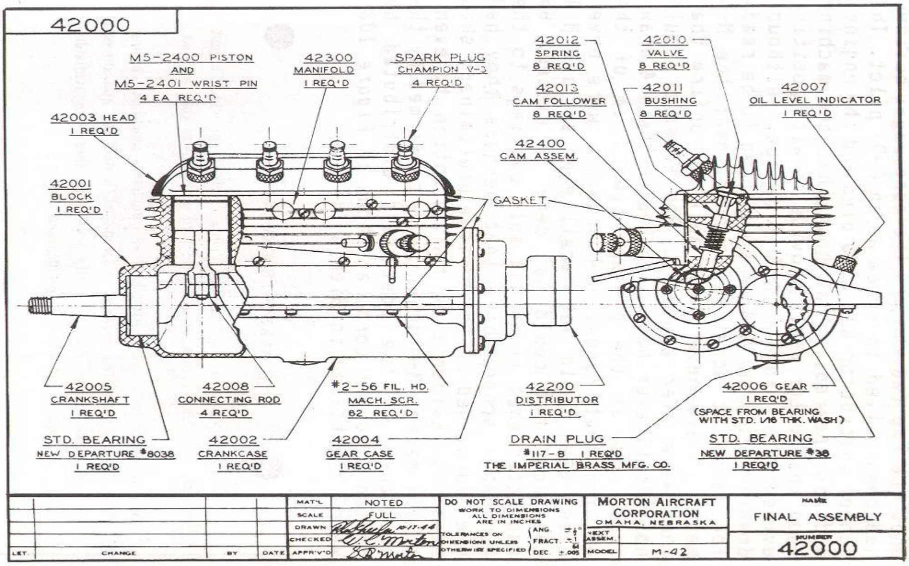

42000
M5-2400 PISTON AND M5-2401 WRIST PIN 4 EA REQ'D
42300 MANIFOLD 1 REQ'D
SPARK PLUG CHAMPION V-3 4 REQ'D
42012 SPRING 8 REQ'D
42010 VALVE 8 REQ'D
42013 CAM FOLLOWER 8 REQ'D
42011 BUSHING 8 REQ'D
42007 OIL LEVEL INDICATOR 1 REQ'D
42003 HEAD 1 REQ'D
42400 CAM ASSEM.
42001 BLOCK 1 REQ'D
GASKET
42005 CRANKSHAFT 1 REQ'D
42008 CONNECTING ROD 4 REQ'D
#2-56 FIL. HD. MACH. SCR. 62 REQ'D
42200 DISTRIBUTOR 1 REQ'D
42006 GEAR 1 REQ'D (SPACE FROM BEARING WITH STD. 1/16 THK. WASH)
STD. BEARING NEW DEPARTURE #8038 1 REQ'D
42002 CRANKCASE 1 REQ'D
42004 GEAR CASE 1 REQ'D
DRAIN PLUG #117-B 1 REQ'D THE IMPERIAL BRASS MFG. CO.
STD. BEARING NEW DEPARTURE #38 1 REQ'D
LET.
CHANGE
BY
DATE
MAT'L NOTED
SCALE FULL
DRAWN 10-17-44
CHECKED W.C. Morton
APPR'V'D
DO NOT SCALE DRAWING WORK TO DIMENSIONS ALL DIMENSIONS ARE IN INCHES
TOLERANCES ON DIMENSIONS UNLESS OTHERWISE SPECIFIED ANG. ±1/2° FRACT. ±1/64 DEC. ±.005
MORTON AIRCRAFT CORPORATION OMAHA, NEBRASKA
NEXT ASSEM.
MODEL M-42
NAME FINAL ASSEMBLY
NUMBER 42000

Fig. 2. Engineering Drawing of an 4 cylinder Morton M42 motor (Source[14])

From figure 2 we are able to understand the importance of knowledge preservation in addition of the use of technology that can be extensible and is able to preserve the essence of a drawing. The motor presented is known as a Morton M42 designed by Mill Morton from the Morton Air Company in 1944, it was checked and approved but there are no details or documents to refer about neither its parts nor its dimensions.

The Morton Company declared they never produced a prototype nor documented any of their designs in an interview with Bob Knutson. All the knowledge and expertise contained in the drawing was lost because there weren't any documents to support the drawing and so neither other company nor individual will have the chance to learn from the motor's design this situation is radical but true and happens to repeat on our days in every company that has a Computer-Aided Design Solution. The designs are stored on plain CAD or Computer-Aided Manufacturing (CAM) files but the real knowledge resides on the person or people involved on the creation of such part. If for any reason the people involved leave the company the expertise and information leave with them and the only thing left behind are empty drawings (that may be as poor as the one presented in figure 5 or as somehow illustrative as the one of Figure 2).

In this case a motor company bought the installations of Morton Company and found the engineering drawing of the Morton M42 motor and finds it practical and economical for their next product line so they hand in the paper design to a CAD expert who replicates what's on paper, later they hand the drawing to the people in their manufacturing team to do as needed to produce such motors. If the designer replicated the drawing at an approximate or similar scale it is possible to deduce a series of key aspects. By knowing the type of screw used it can be deduced what size are the holes, depth and might give a clue about the block's material, by tracking the sparkplug model and the drain plug another piece of information about the block can be known.

3.3 Approach

In order to be able to extract details and useful information from a CAD file we have chosen to work with a neutral file format for two main reasons, first every CAD software has their proprietary APIs and sometimes they lack of documentation; secondly, in order to achieve interoperability we decided to work with a neutral file format that any design tool can save and read from. Currently in the market there are two main “neutral formats”; STandarized Exchange of Product (STEP) which is based on an ISO standard but has a considerable drawback since the full documentation is broken down into “Application Protocols” (AP) with around 523 AP's in total and some more “Application Modules” (AM). On the other hand we have the Initial Graphics Exchange Specification (IGES) started as an effort of the Department of Defense of the U.S. as well as the NIST and other companies to create a neutral format in order to ease the process of exchanging and sharing CAD drawings and it’s important to mention that it is a fully accredited American National Standards Institute (ANSI).

The neutral format for our work is IGES, since its documentation comes in one document and at no cost whereas the documentation available of the Application Protocols of STEP is not free and it requires of at least 3 of them to be able to describe the geometry and contents. The version used is the one available for download which is 5.3.

According to the IGES standard an “*entity*” is the basic unit used among all the documentation, the naming convention of the entities is made up of two main elements; A Unique Name (Line Font, LEP Drilled Hole, etc.) plus a Number identifier, for instance we have the “*View Entity*” (Type 410) or the “*Property Entity*” (Type 406).

In the second case the Property entity includes 28 Forms, which derive from the Property Entity and consider specific cases used to define an element, for instance there is the “*Drawing Units Property*” (Form 17) which declares the drawing units used in the drawing, the full Name of such property will be Property Entity, *Drawing Units Property* (Type 406, Form 17). The IGES standard is made up of 180 entities (including the properties with forms).

The use case for this work is based on the part presented on Figure 1 it has complex geometry due to the circular cavity and a the rectangular cut as well, based on such particular geometry forms the resulting IGES file is in total of 1041 lines. It is important to note that to following our approach described in section 3.3 we will just replicate the figure with approximate dimensions. The competency questions in order to test our approach are:

- It is possible to extract and organize the contents of an IGES file in a human readable way?

- From the information extracted, will we be able to know basic aspects of the drawing such as; which software was used to create the original file? In what units it was drawn? And the date the file was created?
- Can we know how many total entities were used to create the model?

4 Implementation

Parting from the competency questions defined we created a tool to be able to extract all that's inside an IGES file following the standard's documentation (both aligning the file structure as well as the terminology used).

The tools segments the whole IGES files according to the extension defined on the Standard for each section and then lists all the entities found, the output of the extraction and categorization process are the three separated XML files.

The output XML files are based on the three main sections that make up an IGES file; *Directory, Global and Parameter Section*. Each section reports a critical part of the model. The Global Section contains a human-readable report of the drawing some of the key aspects stored on this section are: File Name, Native System (CAD tool used to generate the drawing), Units, Date of creation, author's name and organization, etc. The Directory Entry stores all the entities used to create the model in pairs of lines. The Parameter Data Section holds all the values of the parameters needed for every entity stored on the previous section.

The following segments are extracted from the native IGES file based on the part from Figure 1presented in Section 3.1:

a) Global Section:

```
1H,,1H;,10HSolid Edge,9HPart1.igs,17H By Spatial Corp.,22HXPlus GENERIC/G    1
IGES 3.0,32,38, 6,308,15,7HUnknown,1.000,1,2HIN,1,1.000,15H20120621.1116G  2
19,3.9e-007,0.00,6HAuthor,5HTitle,11,0,15H20120621.111619;             G   3
```

b) Directory Section:

```
314,68.6274528503418,76.86275243759155,86.66667342185974,10HMixe       1P   1
dColor;                                                                1P   2
128,3,1,3,1,0,0,0,0,0,-1.,-1.,-1.,-1.,0.,0.,0.,0.,0.,0.,1.,1.,         3P   3
1.,0.33333333333333,0.33333333333333,1.,1.,0.33333333333333,           3P   4
0.33333333333333,1.,-0.19685039370079,0.,-0.29527559055118,            3P   5
```

c) Parameter Section:

```
314,68.6274528503418,76.86275243759155,86.66667342185974,10HMixe     1P    1
dColor;                                                              1P    2
128,3,1,3,1,0,0,0,0,0,-1.,-1.,-1.,-1.,0.,0.,0.,0.,0.,0.,1.,1.,       3P    3
1.,0.33333333333333,0.33333333333333,1.,1.,0.33333333333333,         3P    4
0.33333333333333,1.,-0.19685039370079,0.,-0.29527559055118           3P    5
-0.19685039370079,-1.18110236220472,-0.29527559055118,               3P    6
```

Now after our tool has extracted and categorized the information of the IGES file, a fragment of the Global Section on our XML output file looks something like this:

```
<ParameterDelimiterCharacter>1H</ParameterDelimiterCharacter>
<ProductIdentificationFromSender>1H;</ProductIdentificationFromSender>
<FileName>10HSolid Edge</FileName>
<NativeSystemID>9HPart1.igs</NativeSystemID>
<PreprocessorVersion>17H By Spatial Corp.</PreprocessorVersion>
```

From the above segment we can easily identify what's the file name, which specific CAD tool was used to originally create the design and what pre-processor was used to generate the corresponding IGES file.

The same happens for the Directory section, which is the largest one since counts and enlists all the entities used along with the 20 corresponding elements that define each one of the entities used.

```
<TypeNumber>314</TypeNumber>
<LineFontPattern>1</LineFontPattern>
<TransformationMatrix>0</TransformationMatrix>
<LineWeightNumber>1</LineWeightNumber>
<ColorNumber>0</ColorNumber>
```

In this case the entity 314 refers to the Color Definition Entity, as stated on the standard it "specifies the relationship of the primary (red, green, and blue) colors to the intensity level of the respective graphics devices as a percent of the full intensity range" here we see that even the colors used on the design are enlisted.

Following the same entity but now looking at its Parameter Data Section we have a dynamic amount of data fields that are directly related to the entity it represents, in this case for entity 314 we have 5 parameter values.

```
<Entity ID="1">
<PS1>314</PS1>
<PS2>68.6274528503418</PS2>
<PS3>76.86275243759155</PS3>
<PS4>86.66667342185974</PS4>
<PS5>10HMixedColor;</PS5>
</Entity>
```

To get an idea of amount of parameters an entity may need to be fully represented, let us mention the case of entity 128 "*Rational B- Spline Surface Entity*" which uses a total count of 58 parameters.

The reason we decided to break down the original IGES file into separate files relies on two main reasons; For any person it's easier to identify and look up for a specific element or entity by looking into a specific file instead of looking up a 1034 line file, on the other hand it's easier to group and manage the different XMLs since they are named after the IGES file of origin and so the loss of context or knowledge is diminished. Both reasons obey the principle that the information must be available for a human or a system (computer) to consume the data it holds and the original file remains untouched so it can be used for visualization purposes having the original file corrupted or altered.

4.1 Results and Discussion

We were able to run our tool on the IGES test file and successfully extracted the contents within it and save such contents into three XML files. We created three sets of figures based on the design from Figure 1; each design was created using three different CAD Software: NX8, Solidworks 2011 and SolidEdge 18. We wanted to know if the information extracted was coherent among all different versions.

All versions were similar to each other, but there were some particular differences:

- The size of the IGES files from the CAD tools differed in file size and line extension
- The amount of entities used by each tool was slightly different based on the fact that each CAD tool uses different entities to create a solid.
- The resulting XML's from each CAD tool successfully extracted all the information from each IGES file but they differed in size and line extension.

It is important to note that the XML files can be easily analyzed and there is no need to jump from line to line as in the original IGES file in order to track down the properties and parameter of an entity as for the original IGES file. By creating XML files from an IGES file we are able to answer each of the competency questions from section 3.3.

- It is possible to extract, visualize and save the contents of any IGES file in XML files without any restriction related to the CAD tool that was used to create the original design.

- From the information extracted we are able to know; which software was used to create the original file, what units were used to create the design and the date the file was created.

- We are able to count and identify how many entities were used to create the model as well as their parameters and specific properties.

5 Conclusions

Even though the resulting files from the extraction process are split up in three separate files, we think the contribution lies on the possibility of extracting, categorizing and identifying key elements from an IGES file without using formerly a CAD tool to obtain such information. We strongly believe that being able to map the contents of such files into a structured XML (based entirely on the IGES documentation) decreases the ambiguity conflicts whenever reading or interpreting the contents of the output files.

5.1 Future Work

The next little step on our work is to be able to replace the number of the entities enlisted from the resulting XML files into their verbal counterpart, based on the terms used by the IGES standard, instead of having Entity 514, 510, 504 have their associated term (Shell, Face and Edge). Once we are able to translate all numerical entities into their Verbal meanings we will create a formal XML schema for the resulting files.

By having the extracted information from the IGES file based on a controlled vocabulary (since it's purely taken from the IGES documentation) we are creating a rough but useful semantic model. This is the prelude of a formal semantic model based on a CAD drawing, but still a lot of work is yet to be done (the creation of an ontology to properly provide a clear and unambiguous meaning to the terms used in the XML and state the relations each entity follow within the IGES scope). Once the ontology is created we will look to integrate other domain ontologies in order to enrich the semantic model with relations deduced by the inclusion of such ontologies.

If we succeed in the creation of semantic models based on CAD drawings we will be able to breach the knowledge loss and ensure that all the information available is unambiguous and the abstraction from a geometric model of a CAD design will not mean that a decision taker has to wait for an expert to understand what's in front of him/her and so new business and operation possibilities will arise.

References

1. National Institute of Standards and Technology (NIST), http://www.nist.gov/director/planning/upload/report04-2.pdf
2. Ryder Center for Supply Chain Management, http://rydercenter.fiu.edu/why-supply-chain-management.html
3. Magrab, E.: Integrated Product and Process Design and Development: The Product Realization Process. CRC Press, New York (1997)
4. Duimering, P.R., et al.: Requirements engineering in new product development. Communications of the ACM 51(3), 77–82 (2008)
5. Sriram Ram, D.: Distributed and Integrated Collaborative Engineering Design. Sarven Publishers (2002)
6. Sriram, R.D., SZykman, S., Durham, D.: Special Issue on Collaborative Engineering. Transactions of the ASME Journal of Computing and Information Science in Engineering 6(2), 93–95 (2006)
7. Bock, C., Gruninger, M.: PSL: A semantic domain for flow models. Software and System Modeling 4(2), 209–231 (2005)
8. Li, M., Zhang, Y.F., Fuh, J.Y.H.: Retrieving Reusable 3D CAD Models Using Knowledge Driven Dependency Graph Partitioning. In: Computer-Aided Design (2010)
9. Žáková, M., Železný, F., Garcia-Sedano, J.A., Masia Tissot, C., Lavrač, N., Křemen, P., Molina, J.: Relational Data Mining Applied to Virtual Engineering of Product Designs. In: Muggleton, S.H., Otero, R., Tamaddoni-Nezhad, A. (eds.) ILP 2006. LNCS (LNAI), vol. 4455, pp. 439–453. Springer, Heidelberg (2007)

10. Dokken, T., Quak, E., Skytt, V.: Requirements from Isogeometric Analysis for Changes in Product Design Ontologies. In: Proceedings of the Focus K3D Conference on Semantic 3D Media and Content, Sophia Antipolis, France (2010)
11. Catalano, C.E., Camossi, E., Ferrandes, R., Cheutet, V., Sevilmis, N.: A product design ontology for enhancing shape processing in design workflows. Journal of Intelligent Manufacturing (2008)
12. Lemaignan, S., Siadat, A., Dantan, J.-Y., Semenenko, A.: MASON: A Proposal For An Ontology Of Manufacturing Domain. In: Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS 2006), pp. 195–200. IEEE Computer Society, Washington, DC (2006)
13. Young, R.I.M., Gunendran, A.G., Cutting-Decelle, A.F., Gruninger, M.: Manufacturing knowledge sharing in PLM: a progression towards the use of heavy weight ontologies. International Journal of Production Research (2007)
14. Morton M42 engine, description and facts, http://modelenginenews.org/cardfile/m42.html
15. Knutson, R.: The Morton Story Part 8: The Morton Fours. Engine Collectors' Journal 14(5-77), 1 (1984)
16. STandarized Exchange of Product (STEP), Industrial automation systems and integration – Product data representation and exchange – Part 21: Implementation methods: Clear text encoding of the exchange structure, http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=33713
17. Initial Graphics Exchange Specification: IGES 5.3, N. Charleston, SC: U.S. Product Data Association, http://www.uspro.org/product-data-resources

Process of Concept Alignment for Interoperability between Heterogeneous Sources

María de Lourdes Guadalupe Martínez-Villaseñor[1,2] and Miguel González-Mendoza[2]

[1] Universidad Panamericana Campus México, Augusto Rodin 498,
Col. Insurgentes-Mixoac, México, D.F., México
[2] Tecnológico de Monterrey, Campus Estado de México
Carretera Lago de Guadalupe Km 2.5,
Atizapán de Zaragoza, Estado de México, México
lmartine@up.edu.mx, mgonza@itesm.mx

Abstract. Some researchers in the community of user modeling envision the need to share and reuse information scattered over different user models of heterogeneous sources. In a multi-application environment each application and service must repeat the effort of building a user model to obtain just a narrow understanding of the user. Sharing and reusing information between models can prevent the user from repeated configurations, help deal with application and services' "cold start" problem, and provide enrichment to user models to obtain a better understanding of the user. But gathering distributed user information from heterogeneous sources to achieve user models interoperability implies handling syntactic and semantic heterogeneity. In this paper, we present a process of concept alignment to automatically determine semantic mapping relations that enable the interoperability between heterogeneous profile suppliers and consumers, given the mediation of a central ubiquitous user model. We show that the process of concept alignment for interoperability based in a two-tier matching strategy can allow the interoperability between social networking applications, FOAF, Personal Health Records (PHR) and personal devices.

Keywords: Concept alignment, Semantic interoperability, user model interoperability.

1 Introduction

Sharing profile information among distributed user models from heterogeneous sources has been envisioned in the community of ubiquitous user modeling researchers [1]. Gathering profile information integrating a ubiquitous user model will provide a better understanding of the user in order to deliver more personalized and proactive services (web applications and services). Sharing and reusing information between models can prevent the user from repeated configurations, help deal with application and services' "cold start" problem, and provide enrichment to user models to obtain a better understanding of the user

The integration of distributed user information from heterogeneous sources and making sense of it to enable user model interoperability, entails handling the semantic

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 311–320, 2013.

heterogeneity of the user models [2].Two mayor approaches for user modeling interoperability are the definition of standard ontologies that provide a common understanding among multiple systems or conversion approach using mediation techniques to provide mappings between different user models [3]. In recent years, it has become clear that a static representation for the ubiquitous user model does not provide the flexibility needed to adapt to new profile suppliers and consumers. So profile suppliers and consumers must not adhere to a standard user model representation and automatic mediation must be provided.

In this paper, we present a process of concept alignment to automatically determine semantic mapping relations that enable the interoperability between heterogeneous profile suppliers and consumers, given the mediation of a central ubiquitous user model. From the process of concept alignment, recommendations are obtained for the inclusion of new concepts, sub collections and collections in the ubiquitous user model concept scheme allowing it to evolve over time.

The rest of the paper is organized as follows. Our approach in the integration of ubiquitous user model is presented in section 2. A brief state of art and related work in schema matching is presented in section 3.The process of concept alignment is explained in section 4. A proof of concept example, results and evaluation are shown in section 5. Conclusions are given in section 6.

2 Ubiquitous User Model Integration

In order to integrate a ubiquitous user model gathering information from heterogeneous sources and be able to reuse it, alignment between similar concepts must be done. The ubiquitous user modeling sources considered for the construction of the ubiquitous user model are: social networking application, personal health records, RDF documents published using Friend of a Friend (FOAF) [4] vocabulary and personal devices with or without attached sensors. All of these profile suppliers provide a source document in XML, JSON or RDF format which can be expressed as Simple Knowledge Organization System (SKOS)[5] concept schemes. These profile suppliers can also be considered as profile consumers for user model enrichment. We also selected Web Services as profile consumers because the user model can be used to customize different steps of interaction with services. User profile information from heterogeneous sources can be reused as hard constrains in the steps of service discovery, selection and composition, and/or as soft constrains identifying useful concepts from the user model that can satisfy web service input parameters requirements.

We can conceive the problem of integrating a ubiquitous user model as the aggregation of profile suppliers and consumers concept schemes. Each of these concept schemes are expressed as SKOS concept schemes. The ubiquitous user model concept scheme is the central mediator in order to enable interoperability between heterogeneous profile suppliers and consumers. We present the interrelations between profile supplier/consumers' concept schemes and the ubiquitous user model concept scheme in figure 1. Note that all concept schemes together constitute a ubiquitous user modeling ontology (U2MIO) presented in [6].

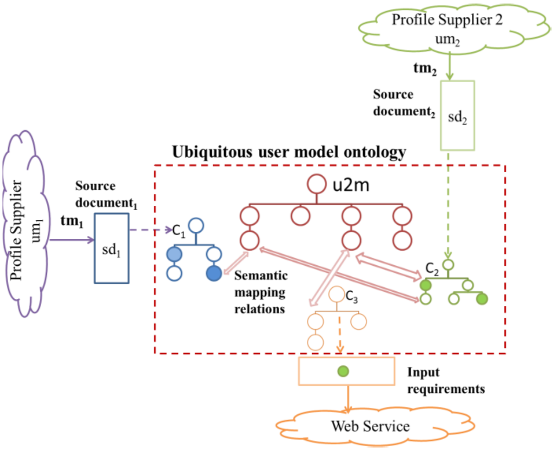


Fig. 1. Interrelations between profile supplier/consumer and the ubiquitous user modeling ontology

3 Schema Matching

“Schema matching is the problem of generating correspondences between elements of two schemas”[7]. So finding one-to-one relations between two concept schemes relies in schema matching techniques. There are surveys [8] [9] and two books [10][11] that reflect the research in this area. One of the schema matching applications is schema integration of independently develop schemas. In particular, we are interested in the integration of ubiquitous user model ontology from heterogeneous and independently developed user profiles. In fact, future trends in data integration suggest that, as new data sources become available, they can be mapped to a single mediated schema [7]. Most data integration implies therefore matching algorithms to produce correspondences and semantic mappings.

Strategies have been proposed that combine matching algorithms in order to better matching results. In 2009, Hamdi et al [12] presented TaxoMap, an alignment tool to discover correspondence between concepts of two ontologies. Their technique is based in TreeTaagger [13] for tagging text and considers label equivalence, high lexical similarity, label inclusion, reasoning on similarity values and reasoning on structure. They evaluated the alignment process with different large scale ontologies with the purpose of ontology enrichment. They suggested taking into account concept properties and not only the hierarchical ones and the use of WordNet [14] as synonym dictionary for future improvements. The structural heuristic was based on Semantic Cotopy[15] measure of a concept which is defined based on the intentional semantics. Bellström et al. [16] presented a three-tier matching strategy consisting of element level matching, structural level matching and taxonomy-based matching which integrates matching methods. As the authors put it, this approach represents “one step towards a semi-automatic method for modeling language independent schema integration”.

4 Process of Concept Alignment

In the previous section we see that a strategy combining several schema matching techniques has been a good choice in the problem of integration of data from heterogeneous sources. So we present a two-tier strategy process of concept alignment to align concept schemes developed from heterogeneous sources in order to integrate a central ubiquitous user model. In this section, we present the formal problem statement and explain the schema matching techniques and alignment strategy used to determine the similarity and mappings between the ubiquitous user model concept schemes and a given interoperability profile supplier or consumer.

4.1 Problem Statement

Following the work in [9] we define a mapping element as a triple: $<c_s,c_t,R>$ where

- c_s is a source concept expressed as skos:Concept from a source concept scheme *X*
- c_t is a target concept expressed as skos:Concept from the ubiquitous user model scheme *U*
- R is a semantic relation (e.g. *equivalent* (=)*; related* ($\subseteq$, $\supseteq$, $\sqcap$); *independent*: ($\perp$) holding between the entities c_s and c_t

The matching operation determines the alignment (*A'*) for the pair of schemas *X* and *U* where *X* can be any concept scheme constructed from a profile supplier/consumer document and *U* will always be the ubiquitous user model concept scheme. See figure 2.

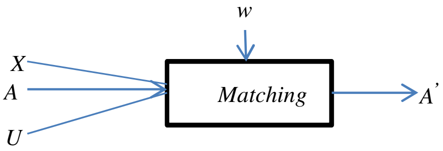


Fig. 2. The matching process

A denotes a previous alignment for this pair of concept schemes (when available) which is to be completed or modified, *w* is an external resource used by the matching process (e.g. WordNet in our case).

We have two basic elements to find mappings between concepts: concept labels and hierarchical structures. Concept values can be used in conflict resolution when a concept is consumed and to determine if further transformation is necessary for interchangeability.

A concept scheme as *(C, $H_{C,}$ V_C)* where C is a set of concepts arranged in a subsumption hierarchy $H_{C.}$ V_C is the set of corresponding concept values if available.

Each concept c_s in a set of concept source C_S is defined by: a label *string1:string2* where *string1* is optional and *string2* is mandatory, and subclass relationships. When namespaces, other than default, are specified in the source document *string1* is used. The attribute or element identifier corresponds to *string2* and it is typically described as a simple or compound word in natural language. A subclass relationship sets up a link with other concepts in the source document. The hierarchical structure most be described as a skos:ConceptScheme.

A concept on the target side C_T is described by set of labels included in the target skos:Concept consisting of lexical labeling, notation and documentation SKOS properties.

4.2 Used Matching Techniques

Different similarity measures [17] have been used in schema matching problems. To determine the similarity between a concept c_s in C_S and a concept c_t in C_T we use the equation 1 which combines three matching techniques.

$$sim(c_s, c_t) = \max(sim_{Dice}(c_s, c_t), d_{lcs}(c_s, c_t), sim_{wordnet}(c_s, c_t)) \forall c_s \in C_S, \forall c_t \in C_T \qquad (1)$$

We chose to include three types of similarity measures: Similarity based in Dice coefficient [18] ($sim_{Dice}(c_s, c_t) \in [0,1]$) has the purpose of finding if concept c_s in X and a concept c_t in U have a high lexical similarity. Equation 2 calculates the longest common substring distance similarity ($d_{lcs}(c_s, c_t) \in [0.1]$) which tries to find if one label is subsumed in the other. The semantic similarity ($sim_{wordnet}(c_s, c_t) \in [0,1]$) is based in Wu and Palmer path lengths method [19] using WordNet as an external resource and tries to find the semantic similarity of the labels. Other languages similarity can be also included if the proper lexical resource is available.

$$d_{lcs}(c_s, c_t) = \frac{Lenght\ (LCS\ (c_s, c_t))}{\min\ (Lenght\ (c_s), Lenght\ (c_t))} \qquad (2)$$

The main methods used are shown in table 1.

Table 1. Used Matching Techniques

Type of similarity	Purpose	Similarity measure
String similarity	High lexical similarity	Dice coefficient
	Label inclusion	Longest common substring (eq.2)
Semantic similarity	Label equivalence	WordNet

The highest of the three similarity measures determines relation R in the triple $<c_s, c_t, R>$ of a mapping element according of the following criteria:

1. *Equivalent* (=): Two concept elements c_s and c_t are *equivalent* iff $sim_O(c_s, c_t) \geq 0.9$.
2. *Related* ($\sqsubseteq, \sqsupseteq, \sqcap$): Two concept elements c_s and c_t are *related* iff $0.9 > sim_O \geq 0.5$.
3. *Independent* ($\perp$):Two concept elements c_s and c_t are i*ndependent* iff $0.5 > sim_O$.

The similarity equation 1 is used in a two-tier matching strategy to determine the mappings of two concept schemes explained in the next section.

4.3 Two-Tier Matching Strategy

The process of concept alignment is based on a two-tier matching strategy that consists in two phases: element level matching and structure level matching.

The purpose of the process of concept alignment is to determine the mappings between two concept schemes at the granularity of concept elements. This means finding the alignment *A'* given the pair of schemas *X* and *U* where *X* can be any concept scheme constructed from a profile supplier/consumer document and *U* will always be the ubiquitous user model concept scheme (as explained in 4.1). We want to determine the all the semantic relations R from the triplet <c_s,c_t,*R*> from all concepts in X to all concepts in U as shown in equation 3.

$$R(c_s, c_t) \forall c_s \in C_S, \forall c_t \in C_T \tag{3}$$

The concepts are directly compared to each other without considering the hierarchy structure and values in the element level matching step. The goal of element level matching is given concept c_s of the source concept scheme X_s, finding the best concept label c_{tb} from a set of concept candidates for alignment in the target concept scheme X_T.

The contexts of the source and target concepts (neighbors of the concepts in the hierarchy) are considered in the structure level matching step. The ultimate goal of this process is determine the one-to-one mappings between the concept c_s of the source concept scheme X_s and the best concept c_{tb} from the set of labels C_t of the target concept scheme X_T. We can also obtain decision recommendations for the inclusion of new concepts, sub collections and collections in the ubiquitous user model concept scheme allowing it to evolve over time.

Element Level Matching Phase

The element level matching phase provides reasoning on the similarity of the isolated concept element. From this stage we can obtain the relation R_0 of every triplet <c_s,c_t,R_0> $\forall c_s \in C_S, \forall c_t \in C_T$ which is a first attempt in finding the semantic mapping relation if we consider only the concept labels without context. The element level matching phase consists in three steps:

1. Calculates the similarity matrix between C_s and C_T with equation 1. The concepts with the highest similarity scores in the target are considered the best suited candidates for alignment (c_{tb}).
2. Determines determine the target collection to which the source document is most related, and to which sub collection each concept in the source c_S belong. The most related collection is determined by calculating which target collection has the higher relative frequency of membership of best suited concepts (Equation 4).

$$r.f.(Collection_i) = \frac{f(c_{bt})}{total\ number\ of\ c_{bt}} \quad \forall i \in dimensions\ of\ X_t \tag{4}$$

3. Determines the relation R_0 between c_s and c_T according to the criteria presented in section 4.2 given the similarity results of the similarity matrix calculated in step 1.

The element level matching is not enough to determine the mapping given the following possible outcomes: homonyms, more than one concept label of the target has *equivalent* relation, no concept label of the target is classified as *equivalent*. So, for every case in the previous outcomes, reasoning on structure is needed.

Structure Level Matching Phase
The structure level matching provides reasoning on structure in order to verify or decline the results obtained in the element level matching phase.
This phase consists in three steps:

1. Select the sets of neighbors N_s and N_T from H_s and H_t which define the context of each concept in the source c_s. The ancestors of c_s and siblings that are terminal elements and share the same parent are considered for N_s. The set of neighbors N_T from the target hierarchy H_t is populated with the labels of the target concepts the sub collection in which the best suited concept for alignment c_{tb} is a member. If more than one c_{tb} is chosen for c_s in the previous element level matching, the labels of all corresponding sub collections are selected for the N_T set. It is important to note that neither c_s nor c_{tb} are included in the sets of neighbors, with the aim of evaluating only the context of these concepts by itself.
2. Calculate a new similarity matrix between the two sets of neighbors N_s and N_T using similarity equation 1determining $R_1(c'_s, c'_{tb})\ \forall c'_s \in N_s, \forall c'_{tb} \in N_t$
3. Given the highest relation $R_0(c_s,\ c_{tb})$obtained from the element level matching phase and the highest relation $R_1(c'_s, c'_{tb})$ resulting of the structure level matching similarity calculation, the next step is apply IF THEN rules to determine the one-to-one mappings and decision recommendation for the inclusion of new concepts, sub collections and collections in the ubiquitous user model concept scheme allowing it to evolve over time.

5 Proof of Concept Example

For our proof of concept demonstration, the ubiquitous user modeling ontology was set-up with Facebook, FOAF, and one profile of a specialized web application to monitor person's diet and physical activity of one user. This demands the design of four concept schemes, one for each profile provider and the ubiquitous user model concept scheme. Semantic mapping relations were established with SKOS properties. We used Protégé ontology Editor for the set-up process.

We considered the integration of the training sessions gathered with a Polar RS300X watch in addition with a Polar S1 foot pod, to the ubiquitous user model. The XML document of the training sessions mentioned was modeled with a concept scheme *P* containing 31 source concepts to be aligned with the concepts of the target ubiquitous user model concept scheme *U*. The outcomes were evaluated by a human expert who decided if the semantic relations found were correct and recommendations make sense. The evaluation and results of the matching process done for the alignment of the concepts in the concept schemes *P* and *U* are presented in section 5.2.

5.1 Process of Concept Alignment Evaluation Criteria

In order to measure the efficiency and effectiveness of the matching/mapping systems, different metrics have been proposed in the literature [20].

In this work, we focus the evaluation of the process of concept alignment in:

- The human effort required by the mapping designer to verify the correctness of the mappings, which is quantified with the metric *overall* [20] and partially measures the efficiency of our process.
- The quality of the generated mappings quantifying the proximity of the results generated by the process of concept alignment to those expected with four known metrics: *precision*, *recall*, *f-measure* and *fall-out* [11].With these metrics we partially measure the effectiveness of our process.

These metrics are based on the notions of *true positives* (TP), *false positives* (FP), *true negatives* (TN) and *false negatives* (FN).

A human expert provided a list of expected matches for the proof of concept example and evaluated the outcomes deciding if the semantic mapping relations found were correct and recommendations make sense. Exact match relations correctly found by the process and good recommendations for concept or collection addition were considered as *TP*. Wrong exact matches were listed as *FP*. When a relevant exact match was not found by the process, a concept was improperly discarded or a wrong recommendation was made, it was registered as *FN*. Properly discarded concepts were recorded as *TN*.

5.2 First Results

The results in terms of the previously mentioned criteria, metrics and conditions are presented in tables 2-3. Table 2 shows the resulting confusion matrix of the matching process between P and U.

Table 2. Resulting confusion matrix of the matching process between P and U

		Expected matches	
		positive	negative
Process of concept alignment outcome	positive	TP=18	FP=0
	negative	FN=4	TN=9

In table 3, we present the efficiency and effectiveness measuring results of the matching process between P and U.

The *precision* calculates the proportion of relevant matches discovered. A 100% *precision* means that all the discovered exact matches found are relevant. *Recall* computes the proportion of matches discovered by the process with respect to all relevant matches determined by the expert. *F-measure* is the trade-off between *precision* and *recall*, in this case we considered the same influence of *precision* and *recall*. *Fall-out* calculates the rate of incorrectly discovered matches out of the number of those not expected by the expert. Since the process did not detect irrelevant matches (wrong exact matches) the *fall-out* is 0%.

Table 3. Efficiency and effectiveness measuring results

Measure	Metric	Result
Quality of generated mappings (Effectiveness)	*Precision*	100%
	Recall	82%
	F-measure	90%
	Fall-out	0%
Human effort (Efficiency)	*Overall*	82%

The *overall* evaluates the amount of work done by a human expert to remove irrelevant exact matches (FP) and add relevant exact matches. 82% *overall* is good considering that the greater this value is, the less effort the expert has to provide.

6 Conclusions and Future Work

We presented a process of concept alignment to allow the interoperability between social networking applications, FOAF, Personal Health Records (PHR) and personal devices. We showed that the process of concept alignment for interoperability based in combined matching techniques and a two-tier matching strategy can enable the integration of source profile information to the ubiquitous user model. This process also permits the evolution of the ubiquitous user model from the U2MIO ontology providing addition recommendations of concepts, sub collections and collections.

The evaluation of our first results show that the quality of generated individual mappings is good because all the discovered exact matches found were classified as relevant by a human expert (100% *precision*) and the process did not detect irrelevant matches (0% *fall-out*) . However, as future work, it is necessary to evaluate whether the combination of concept scheme matches is effective.

Our first results show that the process of concept alignment will ease the pain of a human expert of manually determining the individual mappings to enable the interoperability between heterogeneous sources (82% *overall*).

These results are encouraging but further experimentation and validation of the proposed approach is needed.

References

1. Berkovsky, S., Heckmann, D., Kuflik, T.: Addressing Challenges of Ubiquitous User Modeling: Between Mediation and Semantic Integration. In: Kuflik, T., Berkovsky, S., Carmagnola, F., Heckmann, D., Krüger, A. (eds.) Advances in Ubiquitous User Modelling. LNCS, vol. 5830, pp. 1–19. Springer, Heidelberg (2009)
2. Carmagnola, F.: Handling Semantic Heterogeneity in Interoperable Distributed User Models. In: Kuflik, T., Berkovsky, S., Carmagnola, F., Heckmann, D., Krüger, A. (eds.) Advances in Ubiquitous User Modelling. LNCS, vol. 5830, pp. 20–36. Springer, Heidelberg (2009)
3. Viviani, M., Bennani, N., Egyed-zsigmond, E., Liris, I.: A Survey on User Modeling in Multi-Application Environments. In: Third International Conference on Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services, pp. 111–116 (2010)
4. FOAF Project, `http://www.foaf-project.org/` (accessed on May 22, 2011)
5. Miles, A., Pérez-Agüera, J.: SKOS: Simple Knowledge Organization for the Web. Cataloging & Classification Quarterly 43(3), 69–83 (2007)
6. Martinez-Villasenõr, M.L., González-Mendoza, M.: Towards a Ontology for Ubiquitous User Modeling Interoperability. In: Proceedings of the International Conference on Knowledge Engineering and Ontology Development (KEOD 2012), Barcelona, Spain, October 4-7, pp. 239–244 (2012)
7. Bernstein, P.A., Madhavan, J., Rahm, E.: Generic Schema Matching, Ten Years Later. PVLDB, 695–701 (2011)
8. Rahm, E., Bernstein, P.A.: A Survey of Approaches to Automatic Schema Matching. VLDB J. 10(4), 334–350 (2011)
9. Shvaiko, P., Euzenat, J.: A Survey of Schema-Based Matching Approaches. In: Spaccapietra, S. (ed.) Journal on Data Semantics IV. LNCS, vol. 3730, pp. 146–171. Springer, Heidelberg (2005)
10. Bellahsene, Z., Bonifati, A., Rahm, E. (eds.): Schema Matching and Mapping. Springer (2011)
11. Euzenat, J., Shvaiko, P.: Ontology Matching. Spinger (2007)
12. Hamdi, F., Safar, B., Niraula, N., Reynaud, C.: TaxoMap in the OAEI 2009 alignment contest. In: Ontology Alignment Evaluation Initiative (OAEI) 2009 Campaign - ISWC Ontology Matching Workshop, Westfields Conference Center, Washington, DC (2009)
13. Schmidt, H.: Probabilistic Part-of-Speech Tagging Using Decision Trees. In: International Conference on New Methods in Language Processing (1994)
14. Fellbaum, C.: WordNet and wordnets. In: Brown, K. (ed.) Encyclopedia of Language and Linguistics, 2nd edn., pp. 665–670. Elsevier, Oxford (2005)
15. Maedche, A., Staab, S.: Measuring Similarity between Ontologies. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) EKAW 2002. LNCS (LNAI), vol. 2473, pp. 251–263. Springer, Heidelberg (2002)
16. Bellström, P., Vöhringer, J.: A Three-Tier Matching Stategy for Predesign Schema Elements. In: Proceedings of the Third International Conference on Information, Processing, and Knowledge Management (eKNOW 2011) (2011)
17. Pedersen, T., Patwardhan, S., Michelizzi, J.: WordNet::Similarity-Measuring the Relatedness of Concepts. In: Proceedings of the North American Chapter of the Association for Computer Linguistics (NAACL 2004), Boston, MA, May 3-5, pp. 38–41 (2004)
18. Dice, L.R.: Measures of the amount of ecologic association between species. Ecology 26, 297–302 (1945)
19. Wu, Z., Palmer, M.: Verb semantics and lexicalselection. In: 32nd Annual Meeting of the Associationfor Computational Linguistics, Las Cruces, New Mexico, pp. 133–138 (1994)
20. Bellahsene, Z., Bonifati, A., Duchateau, F., Velagrakis, Y.: On Evaluating Schema Matching and Mapping. In: Bellahsene, Z., Bonifati, A., Rahm, E. (eds.) Schema Matching and Mapping, pp. 253–291. Springer (2011)

A Defeasible Logic of Intention

José Martín Castro-Manzano

Escuela de Filosofía
Universidad Popular Autónoma del Estado de Puebla
21 sur 1103 Barrio de Santiago, Puebla, México 72410
Instituto de Investigaciones Filosóficas
Universidad Nacional Autónoma de México
Circuito Mario de la Cueva s/n Ciudad Universitaria, México, D.F., México 04510
josemartin.castro@upaep.mx

Abstract. Intentional reasoning is a form of logical reasoning with a temporal-intentional and defeasible nature. By covering conditions of material and formal adequacy we describe a defeasible logic of intention to support the thesis that intentional reasoning is *bona fide* reasoning.

Keywords: Defeasible logic, temporal logic, BDI logic, intention.

1 Introduction

Philosophy and computer science have a very deep and unique relation [23]. Not only because these disciplines share some common historical roots —like Leibniz's *mathesis universalis* [8]— and interesting anecdotes —like the correspondence between Simon and Russell [11]—, but more importantly because from the constant dialog that occurs within these disciplines we gain useful hypotheses, formal methods and functional analysis that may shed some light about different aspects of the nature of human behavior, specially under a cognitive schema. The cognitive schema we follow is the BDI model (that stands for *Beliefs*, *Desires* and *Intentions*) as originally exposed by Bratman [4] and formally developed by Rao and Georgeff [21,22]. The general aspect we study is the case of the non-monotonicity of intentional reasoning.

There is no doubt that reasoning using beliefs and intentions during time is a very common task, done on a daily basis; but the nature and the status of such kind of reasoning, which we will be calling intentional, are far from being clear and distinct. However, it would be blatantly false to declare that this study is entirely new, for there are recent efforts to capture some of these ideas already [13,16,19]. But, in particular, we can observe, on one side, the case of BDI logics [22,24] in order to capture and understand the nature of intentional reasoning; and on the other side, the case of defeasible logics [20] to try to catch the status of non-monotonic reasoning.

The problem with these approaches, nevertheless, is that, in first place, human reasoning is not and should not be monotonic [18], and thus, the logical models should be non-monotonic, but the BDI techniques are monotonic; and

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 321–333, 2013.

in second place, intentional states should respect temporal norms, and so, the logical models need to be temporal as well, but the non-monotonic procedures do not consider the temporal or intentional aspect. So, in the state of the art, defeasible logics have been mainly developed to reason about beliefs [20] but have been barely used to reason about temporal structures [14]; on the other hand, intentional logics have been mostly used to reason about intentional states and temporal behavior but most of them are monotonic [7,21,24].

Under this situation our main contribution is a brief study of the nature and status of intentional reasoning to advance the thesis that intentional reasoning is logical reasoning. In particular, this study is important by its own sake because defeasible reasoning has certain patterns of inference and therefore the usual challenge is to provide a reasonable description of these patterns because, if monotonicity is not a property of intentional reasoning and we want to give an adequate description of its notion of inference, then we must study the metalogical properties that occur instead of monotony: once monotonicity is given up, a very organic question about the status of this kind of reasoning emerges: why should we consider intentional reasoning as an instance of a logic *bona fide*?

This paper is organized as follows. In Section 2 we briefly expose what is understood as intentional reasoning. In Section 3 is our main contribution and finally, in Section 4 we sum up the results obtained.

2 Material Adequacy

While developing a logical framework we have to check material and formal adequacy [1]. Material adequacy is about capturing an objective phenomenon. Formal adequacy has to do with the metalogical properties that a notion of logical consequence satisfies. The nature of intentional reasoning is related to a material aspect, while its status is directly connected with a formal one. Concerning material adequacy we will argue how to represent intentions within a BDI model [4]; as for formal adequacy, we will claim that intentional reasoning can be modelled in a well-behaved defeasible logic that satisfies conditions of Consistency, Supraclassicality, Cut and Cautious Monotony [12].

Intentions have certain features that allow us to distinguish them from other cognitive fragments of the BDI model. Proactivity, inertia and admisilibity guarantee that intentions need, respectively, precise notions of commitment, defeasibility and consistency that justify the non-mononotonicity of intentional reasoning. The models of intentional reasoning are built in terms of what we call a bratmanian model. A bratmanian model is a model that *i)* follows general guidelines of Bratman's theory of practical reasoning [4], *ii)* uses the BDI architecture [21] to represent data structures and *iii)* configures notions of logical consequence based on relations between intentional states. There are several logics based upon bratmanian models, but we consider there are, at least, two important problems with the usual logics [7,22,24].

For one, such logics tend to interpret intentions as a unique fragment —usually represented by an operator of the form $\mathsf{INT}(\phi)$—, while Bratman's original theory distinguished three classes of intentions: deliberative, non-deliberative and policy-based. In particular, policy-based intentions are of great importance due to their structure and behaviour: they have the structure of complex rules and behave like plans. This remark is important for two reasons: because the existing formalisms, despite of recognizing the intimate relationship between plans and intentions, seem to forget that intentions behave like plans; and because the rule-like structure allows us to build a more detailed picture of the nature of intentional reasoning since it provides us with the logical structure of intentions.

Let us consider the next example for sake of explanation: assume there is an agent that has an intention of the form $on(X,Y) \leftarrow put(X,Y)$. This means that, for such an agent to achieve $on(a,b)$ it typically has to put a on b. If we imagine such an agent is immersed in a dynamic environment, of course the agent will try to put, *typically*, a on b; nevertheless, a *rational* agent would only do it as long as it is *possible*; otherwise, we would say the agent is not rational. Therefore, it results quite natural to talk about some intentions that are maintained typically but not absolutely if we want to guarantee some level of rationality. And so, it is reasonable to conclude that intentions –in particular policy-based intentions [4]–, allow some form of defeasible reasoning [13] that must comply with some metalogical properties.

However, the bigger problem is that —perhaps due to this structure problem— these systems do not quite recognize that intentional reasoning has a temporal-intentional and defeasible nature. Intuitively, the idea is that intentional reasoning is temporal-intentional because it employs intentions and beliefs as dynamic data structures, i.e., as structures that change during time; but it is also defeasible, because if these data structures are dynamic, their consequences may change. Of course, while defeasible logics capture the second feature, they miss the first one; and while BDI logics catch the former, they avoid the latter.

The bratmanian model we propose tries to respect this double nature by following the general guidelines of Bratman's theory of practical reasoning [4], so we distinguish structural (plan-like structure), functional (proactivity, inertia, admissibility), descriptive (partiality, dynamism, hierarchy) and normative (internal, external consistency and coherence) properties to configure the notion of inference. To capture this notion of inference in a formal fashion the next theory is proposed in terms of $AgentSpeak(L)$[3] (see Appendix for more details):

Definition 1. *(Non-monotonic intentional theory) A non-monotonic intentional theory is specified by a tuple* $\langle bs, ps, F_{bs}, F_{ps}, \vdash, \mid\!\sim, \dashv, \sim\!\mid, \succ \rangle$ *where:*

- bs *denotes the set of beliefs*
- ps *denotes the set of intentions*
- $F_{bs} \subseteq bs$ *denotes the basic beliefs*
- $F_{ps} \subseteq ps$ *denotes the basic intentions*
- $\vdash$ *and* $\dashv$ *are strong consequence relations*
- $\mid\!\sim$ *and* $\sim\!\mid$ *are weak consequence relations*
- $\succ \subseteq ps^2$ *s.t.* $\succ$ *is acyclic*

The item bs denotes the beliefs, which are literals. F_{bs} stands for the beliefs that are considered as basic; and similarly F_{ps} stands for intentions considered as basic. Each intention $\phi \in ps$ is a structure $te : ctx \leftarrow body$ where te represents the goal of the intention —so we preserve *proactivity*—, ctx a context and the rest denotes the body. When ctx or $body$ are empty we write $te : \top \leftarrow \top$ or just te. Also it is assumed that plans are *partially* instantiated.

Internal consistency is preserved by allowing the context of an intention denoted by $ctx(\phi)$, $ctx(\phi) \in bs$ and by letting te be the head of the intention. So, *strong consistency* is implied by internal consistency (given that strong consistency is $ctx(\phi) \in bs$). *Means-end coherence* will be implied by *admissibility* —the constraint that an agent will not consider contradictory options— and the *hierarchy* of intentions is represented by the order relation, which we require to be acyclic in order to solve conflicts between intentions.

With this theory we will arrange a notion of inference where ϕ is strongly (weakly) derivable from a sequence Δ if and only if there is a proof of $\Delta \vdash \phi$ ($\Delta \mid\!\sim \phi$). And also, that ϕ is not strongly (weakly) provable if and only if there is a proof of $\Delta \dashv \phi$ ($\Delta \sim\!\mid \phi$), where $\Delta = \langle bs, ps \rangle$.

2.1 A Defeasible Logic

Initially our approach is similar to a system defined after $B^{KD45}D^{KD}I^{KD}$ with the temporal operators *next* ($\bigcirc$), *eventually* ($\Diamond$), *always* ($\Box$), *until* (U), *optional* (E), *inevitable* (A), and so on, defined after $CTL*$ [6,10].

Since the idea is to define BDI^{CTL} semantics in terms of *AgentSpeak*(L) structures, we need a language able to express temporal and intentional states [15]. Thus, we require, in first place, some way to express these features:

Definition 2. *($BDI^{CTL}_{AS(L)}$ syntax) If ϕ is an AgentSpeak(L) atomic formula, then $\mathsf{BEL}(\phi)$, $\mathsf{DES}(\phi)$ and $\mathsf{INT}(\phi)$ are well formed formulas of $BDI^{CTL}_{AS(L)}$. Every $BDI^{CTL}_{AS(L)}$ formula is a state (s) or path (p) formula:*

- $s ::= \phi | s \wedge s | \neg s$
- $p ::= s | \neg p | p \wedge p | \mathsf{E}p | \mathsf{A}p | \bigcirc p | \Diamond p | \Box p | p \; \mathsf{U} \; p$

As for semantics, initially the meaning of BEL, DES and INT is adopted from [2]. So, we assume the next function:

$$\begin{aligned} agoals(\top) &= \{\}, \\ agoals(i[p]) &= \begin{cases} \{at\} \cup agoals(i) & \text{if } p = +!at : ctx \leftarrow h, \\ agoals(i) & \text{otherwise} \end{cases} \end{aligned}$$

which gives us the set of atomic formulas (at) attached to an achievement goal ($+!$) and $i[p]$ denotes the stack of intentions with p at the top.

Definition 3. *($BDI^{CTL}_{AS(L)}$ semantics) The operators BEL, DES and INT are defined in terms of an agent ag and its configuration $\langle ag, C, M, T, s \rangle$:*

$$\mathsf{BEL}_{\langle ag,C,M,T,s \rangle}(\phi) \equiv \phi \in bs$$

$$\mathsf{INT}_{\langle ag,C,M,T,s\rangle}(\phi) \equiv \phi \in \bigcup_{i \in C_I} agoals(i) \vee \bigcup_{\langle te,i\rangle \in C_E} agoals(i)$$

$$\mathsf{DES}_{\langle ag,C,M,T,s\rangle}(\phi) \equiv \langle +!\phi, i\rangle \in C_E \vee \mathsf{INT}(\phi)$$

where C_I denotes current intentions and C_E suspended intentions.

And now some notation: we will denote an intention ϕ with head g by $\phi[g]$. Also, a negative intention is denoted by $\phi[g^c]$, i.e., the intention ϕ with $\neg g$ as the head. The semantics of this theory will require a Kripke structure $K = \langle S, R, V\rangle$ where S is a set of agent configurations, R is an accessibility relation defined after the transition system of *AgentSpeak*(L) and V is a valuation function that goes from agent configurations to true propositions in those states:

Definition 4. *Let $K = \langle S, R, V\rangle$, then:*

- *S is a set of agent configurations $c = \langle ag, C, M, T, s\rangle$*
- *$R \subseteq S^2$ is a total relation s.t. for all $c \in R$ there is a $c' \in R$ s.t. $(c, c') \in R$*
- *V is valuation s.t.:*
  - $V_{\mathsf{BEL}}(c, \phi) = \mathsf{BEL}_c(\phi)$
  - $V_{\mathsf{DES}}(c, \phi) = \mathsf{DES}_c(\phi)$
  - $V_{\mathsf{INT}}(c, \phi) = \mathsf{INT}_c(\phi)$
- *Paths are sequences of configurations $c_0, \dots, c_n$ s.t. $\forall i(c_i, c_{i+1}) \in R$. We use x^i to indicate the i-th state of path x. Then:*

S1 $K, c \models \mathsf{BEL}(\phi) \Leftrightarrow \phi \in V_{\mathsf{BEL}}(c)$
S2 $K, c \models \mathsf{DES}(\phi) \Leftrightarrow \phi \in V_{\mathsf{DES}}(c)$
S3 $K, c \models \mathsf{INT}(\phi) \Leftrightarrow \phi \in V_{\mathsf{INT}}(c)$
S4 $K, c \models \mathsf{E}\phi \Leftrightarrow \exists x = c_1, \ldots \in K | K, x \models \phi$
S5 $K, c \models \mathsf{A}\phi \Leftrightarrow \forall x = c_1, \ldots \in K | K, x \models \phi$
P1 $K, c \models \phi \Leftrightarrow K, x^0 \models \phi$ *where ϕ is a state formula*
P2 $K, c \models \bigcirc\phi \Leftrightarrow K, x^1 \models \phi$
P3 $K, c \models \Diamond\phi \Leftrightarrow K, x^n \models \phi$ *for $n \geq 0$*
P4 $K, c \models \Box\phi \Leftrightarrow K, x^n \models \phi$ *for all n*
P5 $K, c \models \phi \,\mathsf{U}\, \psi \Leftrightarrow \exists k \geq 0$ *s.t.* $K, x^k \models \psi$ *and for all j, k,* $0 \leq j < k | K, x^j \models \phi$ *or* $\forall j \geq 0 : K, x^j \models \phi$

The notion of inference comes in four flavors: if the sequence is $\Delta \vdash \phi$, we say ϕ is strongly provable; if it is $\Delta \dashv \phi$ we say ϕ is not strongly provable. If is $\Delta \mathrel{|\!\sim} \phi$ we say ϕ is weakly provable and if it is $\Delta \mathrel{\sim\!|} \phi$, then ϕ is not weakly provable.

Definition 5. *(Proof) Let ϕ be a $BDI^{CTL}_{AS(L)}$ formula. A proof of ϕ from Δ is a finite sequence of $BDI^{CTL}_{AS(L)}$ formulas satisfying:*

1. *$\Delta \vdash \phi$ iff*
   1.1. $\Box\mathsf{A}(\mathsf{INT}(\phi))$ *or*
   1.2. $\Box\mathsf{A}(\exists\phi[g] \in F_{ps} : \mathsf{BEL}(ctx(\phi)) \wedge \forall\psi[g'] \in body(\phi) \vdash \psi[g'])$
2. *$\Delta \mathrel{|\!\sim} \phi$ iff*

2.1. $\Delta \vdash \phi$ *or*
2.2. $\Delta \dashv \neg\phi$ *and*
2.2.1. $\Diamond\mathsf{E}(\mathsf{INT}(\phi)\ \mathsf{U}\ \neg\mathsf{BEL}(ctx(\phi)))$ *or*
2.2.2. $\Diamond\mathsf{E}(\exists\phi[g] \in ps : \mathsf{BEL}(ctx(\phi)) \wedge \forall\psi[g'] \in body(\phi)\ \mathrel{|\!\sim}\ \psi[g'])$ *and*
2.2.2.1. $\forall\gamma[g^c] \in ps$*,* $\gamma[g^c]$ *fails at* Δ *or*
2.2.2.2. $\psi[g'] \succ \gamma[g^c]$
3. $\Delta \dashv \phi$ *iff*
3.1. $\Diamond\mathsf{E}(\mathsf{INT}(\neg\phi))$ *and*
3.2. $\Diamond\mathsf{E}(\forall\phi[g] \in F_{ps} : \neg\mathsf{BEL}(ctx(\phi)) \vee \exists\psi[g'] \in body(\phi)\ \dashv \psi[g'])$
4. $\Delta \mathrel{\sim\!|}\ \phi$ *iff*
4.1. $\Delta \dashv \phi$ *and*
4.2. $\Delta \vdash \neg\phi$ *or*
4.2.1. $\Box\mathsf{A}\neg(\mathsf{INT}(\phi)\ \mathsf{U}\ \neg\mathsf{BEL}(ctx(\phi)))$ *and*
4.2.2. $\Box\mathsf{A}(\forall\phi[g] \in ps : \neg\mathsf{BEL}(ctx(\phi)) \vee \exists\psi[g'] \in body(\phi)\ \ \mathrel{\sim\!|}\ \psi[g'])$ *or*
4.2.2.1. $\exists\gamma[g^c] \in ps$ *s.t.* $\gamma[g^c]$ *succeds at* Δ *and*
4.2.2.2. $\psi[g'] \not\succ \gamma[g^c]$

3 Formal Adequacy

We now focus our attention on the formal aspect. We will argue that this model of intentional reasoning is well-behaved.

For consistency, we start with a square of opposition in order to depict logical relationships of consistency and coherence.

Proposition 1. *(Subalterns$_1$) If* $\vdash \phi$ *then* $\mathrel{|\!\sim}\ \phi$*.*

Proof. Let us assume that $\vdash \phi$ but not $\mathrel{|\!\sim}\ \phi$, i.e., $\mathrel{\sim\!|}\ \phi$. Then, given $\vdash \phi$ we have two general cases. Case 1: given the initial assumption that $\vdash \phi$, by Definition 5 item 1.1, we have that $\Box\mathsf{A}(\mathsf{INT}(\phi))$. Now, given the second assumption, i.e., that $\mathrel{\sim\!|}\ \phi$, by Definition 5 item 4.1, we have $\dashv \phi$. And so, $\Diamond\mathsf{E}(\mathsf{INT}(\neg\phi))$, and thus, by the temporal semantics, we get $\neg\phi$; however, given the initial assumption, we also obtain ϕ, which is a contradiction.

Case 2: given the assumption that $\vdash \phi$, by Definition 5 item 1.2, we have that $\exists\phi[g] \in F_{ps} : \mathsf{BEL}(ctx(\phi)) \wedge \forall\psi[g'] \in body(\phi) \vdash \psi[g']$. Now, given the second assumption, that $\mathrel{\sim\!|}\ \phi$, we also have $\dashv \phi$ and so we obtain $\Diamond\mathsf{E}(\forall\phi[g] \in F_{ps} : \neg\mathsf{BEL}(ctx(\phi)) \vee \exists\psi[g'] \in body(\phi) \dashv \psi)$, and thus we can obtain $\forall\phi[g] \in F_{ps} : \neg\mathsf{BEL}(ctx(\phi)) \vee \exists\psi[g'] \in body(\phi) \dashv \psi)$ which is $\neg(\exists\phi[g] \in F_{ps} : \mathsf{BEL}(ctx(\phi)) \wedge \forall\psi[g'] \in body(\phi) \vdash \psi[g'])$. ■

Corollary 1. *(Subalterns$_2$) If* $\mathrel{\sim\!|}\ \phi$ *then* $\dashv \phi$*.*

Proposition 2. *(Contradictories$_1$) There is no* ϕ *s.t.* $\vdash \phi$ *and* $\dashv \phi$*.*

Proof. Assume that there is a ϕ s.t. $\vdash \phi$ and $\dashv \phi$. If $\dashv \phi$ then, by Definition 5 item 3.1, $\Diamond\mathsf{E}(\mathsf{INT}(\neg\phi))$. Thus, by proper semantics, we can obtain $\neg\phi$. However, given that $\vdash \phi$ it also follows that ϕ, which is a contradiction. ■

Corollary 2. *(Contradictories$_2$) There is no ϕ s.t. $\mid\!\sim \phi$ and $\sim\!\mid \phi$.*

Proposition 3. *(Contraries) There is no ϕ s.t. $\vdash \phi$ and $\sim\!\mid \phi$.*

Proof. Assume there is a ϕ such that $\vdash \phi$ and $\sim\!\mid \phi$. By Proposition 1, it follows that $\mid\!\sim \phi$, but that contradicts the assumption that $\sim\!\mid \phi$ by Corollary 2. ■

Proposition 4. *(Subcontraries) For all ϕ either $\mid\!\sim \phi$ or $\dashv \phi$.*

Proof. Assume it is not the case that for all ϕ either $\mid\!\sim \phi$ or $\dashv \phi$. Then there is ϕ s.t. $\sim\!\mid \phi$ and $\vdash \phi$. Taking $\sim\!\mid \phi$ it follows from Corollary 1 that $\dashv \phi$. By Proposition 2 we get a contradiction with $\vdash \phi$. ■

These propositions form the next square of opposition where c denotes contradictories, s subalterns, k contraries and r subcontraries.

$$\begin{tikzcd}
\vdash \phi \arrow[d] & k \arrow[l] \arrow[r] & \sim\!\mid \phi \arrow[d] \\
s \arrow[d] & c \arrow[ul] \arrow[ur] \arrow[dl] \arrow[dr] & s \arrow[d] \\
\mid\!\sim \phi & r \arrow[l] \arrow[r] & \dashv \phi
\end{tikzcd}$$

Proposition 1 and Corollary 1 represent Supraclassicality; Proposition 2 and Corollary 2 stand for Consistency while the remaining statements specify the coherence of the square, and thus, the overall coherence of the system.

Consider, for example, a scenario in which an agent intends to acquire its PhD, and we set the next configuration Δ of beliefs and intentions: $F_{bs} = \{\top\}$, $bs = \{scolarship\}$, $F_{ps} = \{research : \top \leftarrow \top\}$, $ps = \{phd : \top \leftarrow thesis, exam;$ $thesis : scolarship \leftarrow research; exam : \top \leftarrow research\}$. And suppose we send the query: *phd?* The search of intentions with head *phd* in F_{ps} fails, thus the alternative $\vdash \phi[phd]$ does not hold. Thus, we can infer, by contradiction rule (Proposition 2), that it is not strongly provable that *phd*, i.e., that eventually in some state the intention *phd* does not hold. Thus, the result of the query should be that the agent will get its PhD defeasibly under the Δ configuration. On the contrary, the query *research?* will succeed as $\vdash \phi[research]$, and thus, we would say *research* is both strongly and weakly provable (Proposition 1).

As for soundness, we consider the framework is sound with respect to its semantics.

Definition 6. *(Satisfaction) A formula ϕ is true in K iff ϕ is true in all configurations σ in K. This is to say, $K \models \phi \Leftrightarrow K, \sigma \models \phi$ for all $\sigma \in S$.*

Definition 7. *(Run of an agent in a model) Given an initial configuration β, a transition system Γ and a valuation V, $K_\Gamma^\beta = \left\langle S_\Gamma^\beta, R_\Gamma^\beta, V \right\rangle$ denotes a run of an agent in a model.*

Definition 8. *(Validity) A formula $\phi \in BDI^{CTL}_{AS(L)}$ is true for any agent run in Γ iff $\forall K^{\beta}_{\Gamma} \models \phi$*

By denoting $(\exists K^{\beta}_{\Gamma} \models \phi \mathsf{\ U\ } \neg \mathsf{BEL}(ctx(\phi))) \vee \models \phi$ as $\mid\!\approx \phi$, and assuming $\models \phi \geq\ \mid\!\approx \phi$ and $\approx\!\mid \phi \geq\ =\!\mid \phi$, a series of *translations* can be found s.t.:

$$\begin{tikzcd}
\vdash \phi \arrow[r] \arrow[dr] & \forall K^{\beta}_{\Gamma} \models \phi \arrow[r] & \models \phi \arrow[d] \\
 & \mid\!\sim \phi \arrow[r] & \mid\!\approx \phi
\end{tikzcd}$$

And also for the rest of the fragments.

Proposition 5. *The following relations hold:*

$$a)\ \mathit{If} \vdash \phi \mathit{\ then} \models \phi \qquad b)\ \mathit{If}\ \mid\!\sim \phi \mathit{\ then}\ \mid\!\approx \phi$$

Proof. Base case. Taking Δ_i as a sequence with length $i = 1$.

Case a) If we assume $\vdash \phi$, we have two subcases. First subcase is given by Definition 5 item 1.1. Thus we have $\Box \mathsf{A}(\mathsf{INT}(\phi))$. This means, by Definition 4 items P4 and S5 and Definition 3, that for all paths and all states $\phi \in C_I \vee C_E$. We can represent this expression, by way of a translation, in terms of runs. Since paths and states are sequences of agent configurations we have that $\forall K^{\beta}_{\Gamma} \models \phi$, which implies $\models \phi$. Second subcase is given by Definition 5 item 1.2, which in terms of runs means that for all runs $\exists \phi[g] \in F_{ps} : \mathsf{BEL}(ctx(\phi)) \wedge \forall \psi[g'] \in body(\phi) \vdash \psi[g']$. Since Δ_1 is a single step, $body(\phi) = \top$ and for all runs $\mathsf{BEL}(ctx(\phi)))$, $ctx(\phi) \in F_{bs}$. Then $\forall K^{\beta}_{\Gamma} \models \phi$ which, same as above, implies $\models \phi$.

Case b) Let us suppose $\mid\!\sim \phi$. Then we have two subcases. The first one is given by Definition 5 item 2.1. So, we have that $\vdash \phi$ which, as we showed above, already implies $\models \phi$. On the other hand, by item 2.2, we have $\dashv \neg \phi$ and two alternatives. The first alternative, item 2.2.1, is $\Diamond \mathsf{E}(\mathsf{INT}(\phi) \mathsf{\ U\ } \neg \mathsf{BEL}(ctx(\phi)))$. Thus, we can reduce this expression by way of Definition 4 items P3 and S4, to a translation in terms of runs: $\exists K^{\beta}_{\Gamma} \models \phi \mathsf{\ U\ } \neg \mathsf{BEL}(ctx(\phi))$, which implies $\mid\!\approx \phi$. The second alternative comes from item 2.2.2, $\Diamond \mathsf{E}(\exists \phi[g] \in ps : \mathsf{BEL}(ctx(\phi)) \wedge \forall \psi[g'] \in body(\phi) \mid\!\sim \psi[g'])$ which in terms of runs means that for some run $\exists \phi[g] \in ps : \mathsf{BEL}(ctx(\phi)) \wedge \forall \psi[g'] \in body(\phi) \mid\!\sim \psi[g']$, but Δ_1 is a single step, and thus $body(\phi) = \top$. Thus, there is a run in which $\exists \phi[g] \in ps : \mathsf{BEL}(ctx(\phi))$, i.e., $(\exists K^{\beta}_{\Gamma} \models (\phi \mathsf{\ U\ } \neg \mathsf{BEL}(ctx(\phi)))$ by using the weak case of Definition 5 P5. Thus, by addition, $(\exists K^{\beta}_{\Gamma} \models (\phi \mathsf{\ U\ } \neg \mathsf{BEL}(ctx(\phi))) \vee \models \phi$, and therefore, $\mid\!\approx \phi$.

Inductive case. Case a) Let us assume that for $n \leq k$, if $\Delta_n \vdash \phi$ then $\Delta \models \phi$. And suppose Δ_{n+1}. Further, suppose $\Delta_n \vdash \phi$, then we have two alternatives. First one being, by Definition 5 item 1.1, that we have an intention ϕ s.t. $ctx(\phi) = body(\phi) = \top$. Since $body(\phi)$ is empty, it trivially holds at n, and by the induction hypothesis, $body(\phi) \subseteq \Delta_{n+1}$, and thus $\models \phi$. Secondly, by Definition 5 item 1.2, for all runs $\exists \phi[g] \in ps : \mathsf{BEL}(ctx(\phi)) \wedge \forall \psi[g'] \in body(\phi) \vdash \psi[g']$. Thus, for all runs $n, \forall \psi[g'] \in body(\phi) \vdash \psi[g']$, and so by the induction hypothesis, $body(\phi) \subseteq \Delta_{n+1}$, i.e., $\Delta \vdash \psi[g']$. Therefore, $\models \phi$.

Case b) Let us assume that for $n \leq k$, if $\Delta_n \mathrel{|\!\sim} \phi$ then $\Delta \mathrel{|\!\approx} \phi$. And suppose Δ_{n+1}. Assume $\Delta_n \mathrel{|\!\sim} \phi$. We have two alternatives. The first one is given by Definition 5 item 2.1, i.e., $\vdash \phi$, which already implies $\models \phi$. The second alternative is given by item 2.2, $\Delta \dashv \neg\phi$ and two subcases: $\Diamond \mathsf{E}(\mathsf{INT}(\phi)\ \mathsf{U}\ \neg\mathsf{BEL}(ctx(\phi)))$ or $\Diamond \mathsf{E}(\exists \phi[g] \in ps : \mathsf{BEL}(ctx(\phi)) \wedge \forall \psi[g'] \in body(\phi) \mathrel{|\!\sim} \psi[g'])$. If we consider the first subcase there are runs n which comply with the definition of $\mathrel{|\!\approx} \phi$. In the remaining subcase we have $\forall \psi[g'] \in body(\phi) \mathrel{|\!\sim} \psi[g']$, since $body(\phi) \subseteq \Delta_n$, by the induction hypothesis $\Delta \mathrel{|\!\sim} \psi[g']$, and thus, $\Delta_{n+1} \mathrel{|\!\sim} \phi$, i.e., $\mathrel{|\!\approx} \phi$. ■

Corollary 3. *The following relations hold:*

$$a)\ If\ \dashv \phi\ then\ \mathrel{=\!\!|}\ \phi \qquad b)\ If\ \mathrel{\sim\!|}\ \phi\ then\ \mathrel{\approx\!|}\ \phi$$

But there are other formal properties that may be used to explore and define the rationality of intentional reasoning, i.e., its good behavior. In first place, it results quite reasonable to impose Reflexivity on the consequence relation so that if $\phi \in \Delta$, then $\Delta \mathrel{|\!\sim} \phi$.

Further, another reasonable property should be one that dictates that strong intentions imply weak intentions. In more specific terms, that if an intention ϕ follows from Δ in a monotonic way, then it must also follow according to a non-monotonic approach. Thus, in second place, we need the reasonable requirement that intentions strongly mantained have to be also weakly mantained, but no the other way around:

Proposition 6. *(Supraclassicality) If $\Delta \vdash \phi$, then $\Delta \mathrel{|\!\sim} \phi$.*

For a proof of this proposition we can see Proposition 1.

Another property, a very strong one, is Consistency Preservation. This property tells us that if some intentional set is classically consistent, then so is the set of defeasible consequences of it:

Proposition 7. *(Consistency preservation) If $\Delta \mathrel{|\!\sim} \bot$, then $\Delta \vdash \bot$.*

Proof. Let us consider the form of the intention $\bot$. Such intention is the intention of the form $\phi \wedge \neg\phi$, which is, therefore, impossible to achieve, that is to say, for all agent runs, $\mathrel{|\!\sim} \bot$ is never achieved. Thus $\Delta \mathrel{|\!\sim} \bot$ is false, which makes the whole implication true. ■

If an intention ϕ is a consequence of Δ, then ψ is a consequence of Δ and ϕ only if it is already a consequence of Δ, because adding to Δ some intentions that are already a consequence of Δ does not lead to any *increase* of information. In terms of the size of a proof [1], such size does not affect the degree to which the initial information supports the conclusion:

Proposition 8. *(Cautious cut) If $\Delta \mathrel{|\!\sim} \phi$ and $\Delta, \phi \mathrel{|\!\sim} \psi$ then $\Delta \mathrel{|\!\sim} \psi$.*

Proof. Let us start by transforming the original proposition into the next one: if $\Delta \mathrel{\sim\!|} \psi$ then it is not the case that $\Delta \mathrel{|\!\sim} \phi$ and $\Delta, \phi \mathrel{|\!\sim} \psi$. Further, this proposition can be transformed again: if $\Delta \mathrel{\sim\!|} \psi$ then either $\Delta \mathrel{\sim\!|} \phi$ or $\Delta, \phi \mathrel{\sim\!|} \psi$

from which, using Corollary 1, we can infer: if $\Delta \dashv \psi$ then either $\Delta \dashv \phi$ or $\Delta, \phi \dashv \psi$. Now, let us assume that $\Delta \dashv \psi$ but it is not the case that either $\Delta \dashv \phi$ or $\Delta, \phi \dashv \psi$, i.e., that $\Delta \dashv \psi$ but $\Delta \vdash \phi$ and $\Delta, \phi \vdash \psi$. Considering the expression $\Delta, \phi \vdash \psi$ we have two alternatives: either $\psi \in \mathit{body}(\phi)$ or $\psi \notin \mathit{body}(\phi)$. In the first case, given that $\Delta \vdash \phi$ then, since $\psi \in \mathit{body}(\phi)$ it follows that $\vdash \psi$, but that contradicts the assumption that $\Delta \dashv \psi$. In the remaining case, if $\Delta, \phi \vdash \psi$ but $\psi \notin \mathit{body}(\phi)$, then $\Delta \vdash \psi$, which contradicts the assumption that $\Delta \dashv \psi$. ■

If we go a little bit further, we should look for some form of Cautious Monotony as the converse of Cut in such a way that if ϕ is taken back into Δ that does not lead to any *decrease* of information, that is to say, that adding implicit information is a monotonic task:

Proposition 9. *(Cautious monotony) If $\Delta \mid\sim \psi$ and $\Delta \mid\sim \gamma$ then $\Delta, \psi \mid\sim \gamma$.*

Proof. Let us transform the original proposition: if $\Delta, \psi \sim\mid \gamma$ then it is not the case that $\Delta \mid\sim \psi$ and $\Delta \mid\sim \gamma$. Thus, if $\Delta, \psi \sim\mid \gamma$ then either $\Delta \sim\mid \psi$ or $\Delta \sim\mid \gamma$, and by Corollary 1, if $\Delta, \psi \dashv \gamma$ then either $\Delta \dashv \psi$ or $\Delta \dashv \gamma$. Now, let us suppose that $\Delta, \psi \dashv \gamma$ but it is false that either $\Delta \dashv \psi$ or $\Delta \dashv \gamma$, this is to say, that $\Delta, \psi \dashv \gamma$ and $\Delta \vdash \psi$ and $\Delta \vdash \gamma$. Regarding the expression $\Delta, \psi \dashv \gamma$ we have two alternatives: either $\gamma \in \mathit{body}(\psi)$ or $\gamma \notin \mathit{body}(\psi)$. In the first case, since $\gamma \in \mathit{body}(\psi)$ and $\Delta \dashv \psi$, then $\dashv \gamma$, which contradicts the assumption that $\Delta \vdash \gamma$. On the other hand, if we consider the second alternative, $\Delta \dashv \gamma$, but that contradicts the assumption that $\Delta \vdash \gamma$. ■

4 Conclusion

It seems reasonable to conclude that this bratmanian model of intentional reasoning captures relevant features of the nature of intentional reasoning and can be modelled in a well-behaved defeasible logic that clarifies its status, since it satisfies conditions of Consistency, Soundness, Supraclassicality, Consistency Preservation, Cautious Cut and Cautious Monotony. In other words, it is plausible to conclude that intentional reasoning has the right to be called *logical reasoning* since it behaves, materially and formally, as a logic, strictly speaking, as a non-monotonic logic.

The relevance of this work becomes clear once we notice that, although intentions have received a lot of attention, their dynamic features have not been studied completely [16]. There are formal theories of intentional reasoning [7,17,22,24] but very few of them consider the revision of intentions [16] or the non-monotonicity of intentions [13] as legitimate research topics, which we find odd since the foundational theory guarantees that such research is legitimate and necessary [4]. Recent works confirm the status of this emerging area [13,16,19].

Acknowledgements. The author would like to thank each one of the anonymous reviewers for their helpful comments and precise corrections; and the School of Philosophy at UPAEP for all the assistance. This work has also been supported by the CONACyT scholarship 214783.

References

1. Antonelli, A.: Grounded Consequence for Defeasible Logic. Cambridge University Press, Cambridge (2005)
2. Bordini, R.H., Moreira, Á.F.: Proving BDI properties of agent-oriented programming languages. Annals of Mathematics and Artificial Intelligence 42, 197–226 (2004)
3. Bordini, R.H., Hübner, J.F., Wooldridge, M.: Programming Multi-Agent Systems in AgentSpeak using Jason. Wiley, England (2007)
4. Bratman, M.: Intention, Plans, and Practical Reason. Harvard University Press, Cambridge (1987)
5. Castro-Manzano, J.M., Barceló-Aspeitia, A.A., Guerra-Hernández, A.: Consistency and soundness for a defeasible logic of intention. Advances in soft computing algorithms, Research in Computing Science 54 (2011)
6. Clarke Jr., E.M., Grumberg, O., Peled, D.A.: Model Checking. MIT Press, Boston (1999)
7. Cohen, P., Levesque, H.: Intention is choice with commitment. Artificial Intelligence 42(3), 213–261 (1990)
8. Couturat, L.: La logique de Leibniz d'aprés de documents inédits. G. Olms, Hildesheim (1962)
9. Dastani, M., van Riemsdijk, M.B., Meyer, J.C.: A grounded specification language for agent programs. In: AAMAS 2007, pp. 1–8. ACM, New York (2007)
10. Emerson, A.: Temporal and modal logic. In: Handbook of Theoretical Computer Science. Elsevier Science Publishers B.V, Amsterdam (1990)
11. Feinberg, B., Kasrils, R. (eds.): Dear Bertrand Russell... A selection of his correspondence with the general public, 1950–1968. George Allen and Unwin, London (1969)
12. Gabbay, D.M.: Theoretical foundations for nonmonotonic reasoning in expert systems. In: Apt, K. (ed.) Logics and Models of Concurrent Systems, pp. 439–459. Springer, Berlin (1985)
13. Governatori, G., Padmanabhan, V., Sattar, A.: A Defeasible Logic of Policy-Based Intention (Extended Abstract). In: McKay, B., Slaney, J.K. (eds.) Canadian AI 2002. LNCS (LNAI), vol. 2557, pp. 723–723. Springer, Heidelberg (2002)
14. Governatori, G., Terenziani, P.: Temporal Extensions to Defeasible Logic. In: Proceedings of IJCAI 2007 Workshop on Spatial and Temporal Reasoning, India (2007)
15. Guerra-Hernández, A., Castro-Manzano, J.M., El-Fallah-Seghrouchni, A.: CTLAgentSpeak(L): a Specification Language for Agent Programs. Journal of Algorithms in Cognition, Informatics and Logic (2009)
16. Hoek, W., van der Jamroga, W., Wooldridge, M.: Towards a theory of intention revision. Synthese. Springer (2007)
17. Konolige, K., Pollack, M.E.: A representationalist theory of intentions. In: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI 1993), pp. 390–395. Morgan Kaufmann, San Mateo (1993)
18. Nute, D.: Defeasible Logic. In: Bartenstein, O., Geske, U., Hannebauer, M., Yoshie, O. (eds.) INAP 2001. LNCS (LNAI), vol. 2543, pp. 151–169. Springer, Heidelberg (2003)
19. Icard, T., Pacuit, E., Shoham, Y.: Joint revision of belief and intention. In: Proceedings of the Twelfth International Conference on the Principles of Knowledge Representation and Reasoning (2010)
20. Prakken, H., Vreeswijk, G.: Logics for defeasible argumentation. In: Gabbay, D., Guenthner, F. (eds.) Handbook of Philosophical Logic, 2nd edn., vol. 4, pp. 219–318. Kluwer Academic Publishers, Dordrecht (2002)

21. Rao, A.S., Georgeff, M.P.: Modelling Rational Agents within a BDI-Architecture. In: Huhns, M.N., Singh, M.P. (eds.) Readings in Agents, pp. 317–328. Morgan Kaufmann (1998)
22. Rao, A.S.: AgentSpeak(L): BDI Agents Speak Out in a Logical Computable Language. In: Perram, J., Van de Velde, W. (eds.) MAAMAW 1996. LNCS, vol. 1038, pp. 42–55. Springer, Heidelberg (1996)
23. Turner, R., Eden, A.: The Philosophy of Computer Science. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy (Summer 2009 Edition), `http://plato.stanford.edu/archives/sum2009/entries/computer-science`
24. Wooldridge, M.: Reasoning about Rational Agents. MIT Press, Cambridge (2000)

Appendix

AgentSpeak(L) Syntax. An agent ag is formed by a set of plans ps and beliefs bs (grounded literals). Each plan has the form $te : ctx \leftarrow h$. The context ctx of a plan is a literal or a conjunction of them. A non empty plan body h is a finite sequence of actions $A(t_1, \ldots, t_n)$, goals g (achieve ! or test ? an atomic formula $P(t_1, \ldots, t_n)$), or beliefs updates u (addition + or deletion −). $\top$ denotes empty elements, e.g., plan bodies, contexts, intentions. The trigger events te are updates (addition or deletion) of beliefs or goals. The syntax is shown in Table 1.

Table 1. Sintax of *AgentSpeak*(*L*)

ag	$::= bs \quad ps$	h	$::= h_1; \top \mid \top$
bs	$::= b_1 \ldots b_n \ (n \geq 0)$	h_1	$::= a \mid g \mid u \mid h_1; h_1$
ps	$::= p_1 \ldots p_n \ (n \geq 1)$	at	$::= P(t_1, \ldots, t_n) \ (n \geq 0)$
p	$::= te : ctx \leftarrow h$	a	$::= A(t_1, \ldots, t_n) \ (n \geq 0)$
te	$::= +at \mid -at \mid +g \mid -g$	g	$::= !at \mid ?at$
ctx	$::= ctx_1 \mid \top$	u	$::= +b \mid -b$
ctx_1	$::= at \mid \neg at \mid ctx_1 \wedge ctx_1$		

AgentSpeak(L) Semantics. The operational semantics of AgentSpeak(L) are defined by a transition system, as showed in Figure 1, between configurations $\langle ag, C, M, T, s \rangle$, where:

- ag is an agent program formed by beliefs bs and plans ps.
- An agent circumstance C is a tuple $\langle I, E, A \rangle$ where I is the set of intentions $\{i, i', \ldots, n\}$ s.t. $i \in I$ is a stack of partially instantiated plans $p \in ps$; E is a set of events $\{\langle te, i \rangle, \langle te', i' \rangle, \ldots, n\}$, s.t. te is a *triggerEvent* and each i is an intention (internal event) or an empty intention $\top$ (external event); and A is a set of actions to be performed by the agent in the environment.
- M is a tuple $\langle In, Out, SI \rangle$ that works as a *mailbox*, where In is the mailbox of the agent, Out is a list of messages to be delivered by the agent and SI is a register of suspended intentions (intentions that wait for an answer message).

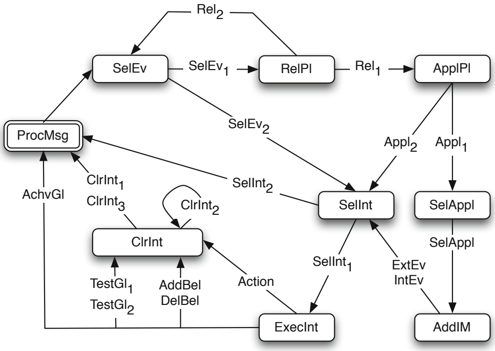


Fig. 1. The interpreter for AgentSpeak(L) as a transition system

- T is a tuple $\langle R, Ap, \iota, \epsilon, \rho \rangle$ that registers temporal information: R is the set of relevant plans given certain *triggerEvent*; *Ap* is the set of applicable plans (the subset of R s.t. $bs \models ctx$); ι, ϵ and ρ register, respectively, the intention, the event and the current plan during an agent execution.
- The label $s \in \{SelEv, RelPl, AppPl, SelAppl, SelInt, AddIM, ExecInt, ClrInt, ProcMsg\}$ indicates the current step in the reasoning cycle of the agent.

Towards Computational Political Philosophy

José Martín Castro-Manzano

Escuela de Filosofía
Universidad Popular Autónoma del Estado de Puebla
21 sur 1103 Barrio de Santiago, Puebla, México 72410
josemartin.castro@upaep.mx

Abstract. We present the initial results of a series of studies around the concept of computational political philosophy. We show a computational interpretation of three classical models of political philosophy that justify three different forms of government by implementing some experiments in order to provide evidence to answer the question of which political philosophy proposes a more plausible form of government. We focus in the relation commitment *vis-à-vis* earnings and we observe that although some political philosophies would justify highly united (communal) or highly competitive (liberal) communities, they would not necessarily imply societies with a reasonable level of welfare.

Keywords: Political philosophy, experimental philosophy, social commitment, welfare.

1 Introduction

The future of philosophy is next to the lab. The philosophy of the future is also an experimental one. This relatively novel way of *making* philosophy is highly entwined, by its methods and history, with exact philosophy [6]; thereby, finding an organic relation between philosophy and computer science is quite natural, specially by way of logic and formal methods. But the important thing is that from this unique relation we not only obtain interesting hypotheses, complex theories and even formal approximations to support our understanding of intelligent —and rational— behaviour, we also gain the advantages of the tools and techniques of artificial intelligence, a science currently used to construct new technologies but also a cultural field created to understand and represent cognitive phenomena, which includes socio-political behaviour: another fundamental aspect of the intelligence spectrum.

Under this horizon we can identify, very easily, different hypotheses from political philosophy, agent-based methods for social simulation and formal specifications for rational agency. More precisely, political philosophy offers the fundamentals of the several hypotheses and theories that fill our cultural schemas with ideologies, arguments and theses about our political behaviour [13]. In second place, simulation has been concerned, traditionally, with simulation of physical and numerical events; however, there is a growing trend in the field of social simulation in order to provide useful and comprehensible methods to explore —and

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 334–346, 2013.

even visualize— social phenomena that may be analitically derivable in principle but that is not directly observable [2]. And, last but not least, intelligent behaviour seems to have a sufficiently appropriate explanation in a theoretical framework that gives beliefs, desires and intentions a central place. This intentional stance [9] to model cognitive phenomena provides levels of explanation, prediction and coordination that other frameworks do not naturally provide [5].

Given this briefing, in this work we present the initial results of a series of studies about comparative political philosophy. We pretend to provide reasons to answer the question: which political philosophy proposes a more plausible form of government? In order to suggest an answer to this problem we follow three typical, famous and classical political philosophies: the first one justifies a form of communal government, closer to an anarchy; the second one supports the need for a sage ruler; and finally, the third one argues for a form of welfare state, closer to a democracy.

To achieve our goal we proceed in three general steps: we explain and instantiate the models suggested by Shook [22] and Bunge [7], then we describe some computational experiments after the previous representations and we finish with an interpretation and a discussion about the obtained results. We will focus on the relation commitment *vis-à-vis* earnings and we will quickly observe that although some political philosophies would justify highly united societies (communal) or highly competitive (liberal) communities, they would not necessarily imply societies with a reasonable level of welfare.

At this point we would like to mention, very quickly, that the importance of this study —and the relevance of the problem and its solution— comes not only from a genuine interest in the past of our political institutions, but rather from a vivid interest in the future of social and political behaviour, since we consider that, on one side, the experimental character of this study enriches the philosophical aspect of the general discussion —by providing quantitative evidence about questions that are usually treated on opinion grounds—, but its philosophical appeal expands the limits of the mere experiments —by offering arguments and interpretations about facts that we cannot directly observe but that we usually suspect.

The paper is distributed as follows. In Section 2 we discuss the role of political philosophy and how can we use social simulation to deal with these problems. In Section 3 we describe the abstract models and we propose an instantiation. In Section 4 we display the main results and a brief discussion. Finally in Section 5 we conclude pointing out some remarks and mentioning current and future work.

2 Preliminaries: Political Philosophy and Social Simulation

2.1 What's Political Philosophy?

Political philosophy is the branch of philosophy that studies the nature, causes and consequences of *good* and *bad* government [18]. This nice attempt of definition, rather vague, implies that there is at least something that allows us to distinguish the good governments from the bad ones. And this guarantees some sort of political knowledge. In this context the concept of government is, indeed, wider than the positive one, since it admits a more general interpretation of a political institution: it includes the notions of positive government, law, authority, and so on. Thus, the main goal of political philosophy is to provide knowledge about political institutions. In the quest to achieve this goal political philosophy has developed two large sets of problems that comprehend, *grosso modo*, its essence and scope: *de facto* and *ab initio* problems. So, for instance, the usual question "Why should we follow the law?" is a typical *de facto* problem; while "Which form of government is more sustainable?" is rather an *ab initio* question. In this work we are interested in the second kind of problems.

2.2 Fundamental Works

Different famous authors through several classical masterpieces have already discussed these affairs. In fact we can backtrack this tradition in the classical Greece with the sofists and Plato; in Medieval Europe it is possible to interpret a form of contract in the ways of feudalism, when at the same time it is not hard to denote Grotius and Pufendorf as some of the most representative authors of the problem of the contract and the natural law during the Renaissance and the Enlightenment [8]. However, few are the works that have the greatness and influence of, say, Plato's *Republic*, Hobbes' *Leviathan* [14], Rousseau's *Social contract* [21], Locke's *Second treatise* [16], Marx and Engels' *Manifesto* [17] or Rawls' *Theory of justice* [20]. In this work, nevertheless, due to reasons of space and for sake of the research, we focus on certain aspects of the *Leviathan*, the *Social Contract* and the *Theory of justice.*

Now, this choice is far from being arbitrary. We center our attention in these masterpieces not only because they are well known and famous, but because in they we find a very good (i.e., susceptible of being abstracted) description of the state of nature and the possible explanations of the origin of the government in formal and experimental terms; and because these three works provide classical arguments to justify —and in some cases even attempt to explain— which form of government is the *best*, to use the parlance of our times.

The state of nature is a hypothetical state that describes human life before any form of factual government. However, despite this state is generally well described there is no formal or experimental method to confirm or disconfirm such idea. On the other hand, possible explanations of the origin of the civil state, as we know, rely on the premises of some sort of contract that, in this context, is a form of self-organization among agents, an agreement in which the agents renounce their natural rights in order to become subjects of civil rights. This transference is a transition between states that is usually called *contract*: this is another hypothesis that needs some justification. The results in this work give some pointers to answer this couple of problems.

Meanwhile, let us start by describing, quickly, the general lines of Hobbes' description. In the *Leviathan* [14] Hobbes argues that in the absence of political institutions that guarantee superior forms of authority the state of nature is a state of constant war and competition because, given that all human beings share the same freedom and equality, it follows that everyone desires the same and, thus, everyone fights for it. But since no authority provides a basis for trust the social behaviour is characterized by an unrestricted freedom that, ultimately, leads to an egoist behaviour. So, once the legal and moral restrictions are gone, the state of nature is a state of *bellum omnium contra omnes.*

It is interesting to notice that this state seems to be a consequence of the conditions of equality and freedom, and not a proper consequence of a human nature that is essentially evil, as usually explained; but in any case, morality and responsibility are practically null. This is the sense behind the famous — and wrongly attributed to Hobbes— *dictum*: *homo homini lupus est.* However, according to the hobbesian anthropological analysis, humans, in their exercise of rationality prefer to guarantee their safety in spite of losing their natural freedom by alienating their power to a sovereing.

On the contrary, Rousseau's approach is rather a defence of the state of nature. In the *Social Contract* [21] Rousseau argues that the state of nature, given that has no influence from any political or social institution, is a state of natural freedom and happiness that is evident in the equality of the individuals under the premise that we are —again— naturally free. Since such is the natural state and, notices Rousseau, we live nowadays —*mutatis mutandis*— immersed in social chains there is some sort of social contradiction: if we are naturally free, how is it possible that we live with so many legal and moral constrains? This contradiction can only be explained by the existence of the current government that is the source of the social problems, inequality and loss of freedom.

Despite some obscurity Rousseau's goal is clear: it is not the innocent justification of the state of nature and the *good savage* —as opposed to Hobbes' *evil savage*, so to speak—, but rather the assertion that the government has to consider the state of nature as its future and not as its past, as its destiny and not as its origin.

In *Theory of justice* [20] Rawls uses the hypothetical resource of the state of nature, not as a historical hypothesis (as in the case of Hobbes and Rousseau), but as a special condition for a mental experiment. In an original position where the individual agents are in a veil of ignorance —a special condition in which none of the participants may know the result of their choice—, it follows that the agents choose, by *rational choice*, two principles: the principle of equality and the principle of difference.

Rawls' *gedanke* experiment goes, more or less, like this: let us suppose that we have two rational individuals prior to their arrival to the world. They are in a veil of ignorance and are presented with the following alternatives: equality or inequality of freedoms and opportunities. If both choose equality, then surely when they arrive to the world they would have justification to demand the reparation of inequalities and the convenient distribution of opportunities; otherwise, they would not have the right to demand any reparation nor distribution in case of being part of the unprotected (i.e., those who arrive to the world without the same liberties and opportunities). Given that both are in a veil of ignorance and they are both rational, the best strategy consists in picking equality.

Whether we agree or not with these general arguments of political philosophy —that are at the basis of current forms of government, as we will see— we can observe two important notes: *i)* these *models* can, properly speaking, be computationally *modeled* using agent-based social simulation, since these proposals offer a relatively nice characterization of the individual agents, the environment and their relations; and *ii)* these descriptions are done in terms of the beliefs, desires and intentions of the agents.

2.3 Agent-Based Social Simulation

This kind of simulation uses computational models to obtain comprehension about the behaviour of complex systems by way of visualizations and evaluations. The principal components of agent-based social simulation are:

1. A set of *agents* with their specifications
2. An *environment* that gives the agents a *locus* for perception and action
3. A mechanism of *interaction* that allows the communication with the environment and other agents

This type of simulation can be used as an experimental tool for the study of social problems [11,12] and, we think, this technique contributes to the treatment of problems of social and political philosophy because the general idea is that this kind of social simulation can show a behaviour that is similar to the behaviour of human social interactions. Plus, it has the following advantages [12]:

1. It allows the observation of properties of the model that may be analitically derivable in principle but that have not been stablished yet
2. Provides conditions to find alternative explanations to natural phenomena
3. Gives opportunity to study properties that are hard to observe in nature
4. It facilitates the study of phenomena in isolation that may be recorded and reproduced if necessary
5. Models sociality by way of representations that the agents have about the states of other agents

Thus, it is not hard to notice that we can complete our studies with an agent-based approach: the ideas of Hobbes, Rousseau and Rawls constitute classical models of political philosophy that may be modeled with multiagent technologies in order to confirm or falsify hypotheses, in order to provide answers to open problems and also to enrich the discussion in philosophy and politics.

3 The Model

Once we have exposed the problem and a possible critical course of action, we show the abstract specification of the agents and the environment. Let us recall that for this study we have proposed a representation of three classical political philosophies. To reach this first goal we begin by following the pattern suggested by Shook [22] in its novel project of comparative political philosophy —not comparative politics. Following his taxonomy the political philosophies we have sketched above can be described as in Table 1.

Table 1. Agent taxonomy. Adapted from [22]

Category	Hobbes	Rousseau	Rawls
Humanity	Obstinate	Simple	Neutral
Morality	Principled	Intuitive	Personal
Sociality	Plural	Sectarian	Plural
Responsibility	Obedience	Free	Autonomous

Of course, the terms used by this model are reserved and have a more or less precise meaning. So, under this model we must understand *Humanity* as the set of anthropological features that describe the state of individual agents. *Morality* has to be understood in terms of the kind of rules that determine the moral behaviour of individual agents. *Sociality* means the way in which agents organize themselves; and last, *Responsibility* has to be interpreted by the question: how does an agent responds to her morality? There is a fifth variable —*Form of government*— that, just for the moment, we will not consider.

An *Obstinate* humanity is defined as a humanity that, in everyday terms, can be classified as "evil", although it would be more precise to talk about an egoist behaviour. A *Simple* one has to be understood as a "good" one, but it is better to think about it as altruistic. Finally, a *Neutral* humanity is that which is neither obstinate nor simple. Indeed, these short descriptions do not quite clarify the definition of such variables, but in the meantime we will use these vague characterizations just to give some pointers, although we will define them in a more exact manner.

A *Principled* morality is founded in heteronomous rules, and so the corresponding responsibility is *Obedience.* An *Intuitive* morality needs no external justification or external rules because it is, let us say, naturally innate and so the corresponding responsibility is *Free.* A *Personal* morality is based upon autonomous rules and so its responsibility is *Autonomous. Sociality* variables are understood as they usually do, but as we will see, we will define sociality as the number of interactions between agents.

Now, despite this model allows us to think in terms of a research program of comparative political philosophy, there are some aspects of it that are not as precise as we would like them to be and they do not work for our purposes, they do not help us in achieving our goal. Thus, in order to preserve the advantages of this model and avoid its vagueness we adapt Bunge's model [7] where these types of agents can be represented. According to this second model there are three types of distinguished social-political behaviours (altruism, egoism and reciprocation) that can be captured by a measure of social commitment C:

$$C = r + e \times s + m \quad \text{where } e, m \in [0,1] \text{ and } s, r \in \mathbb{N} \tag{1}$$

The value r denotes the expected return of an action, e is an equity factor, m is a grade of morality and s is sociality value. Therefore, in the archetypical cases we have the following arrangements:

1. When $m = e = 0$, $C = r$: egoism occurs when social commitment is equal to the expected value
2. When $m = e = 1$, $C \geq r$: altruism occurs when social commitment is greater than the expected value
3. When $m > 0$ and $e \geq 0$, $C > r$: reciprocation occurs when social commitment is strictly greater than the expected value

With this information it is not to hard to observe that we can attach these models to each other. We use the first one to provide the taxonomy and the general foundations to interpret different political philosophies; the second, to give precision and formality to the first one. In this way what we suggest is that the variable m can be interpreted in terms of $Humanity$ and $Morality$; e in terms of *Responsibility* and s in terms of the *Sociality* understood as the number of agent interactions.

If we follow the classical arguments proposed by the political philosophies into discussion, since morality and responsibility in Hobbes have to be, respectively, principled and given by obedience, originally the equity and morality start in 0 (because they are null in a state of nature). In the case of Rousseau, since we have a simple and intuitive characterization, their initial value is rather 1; on the contrary, following Rawls we have a neutral and personal description, so we assume morality is different from 0 and equity is the average, 0.5 (see Table 2).

There are, of course, some strategies that the agents follow and are given by the already known strategies [1]. For sake of experiment, r will be defined by a payment defined after the classical Prisoner's dilemma payoff matrix (see Table 3).

Table 2. Instantiation of the model

Variable	**Hobbes**	**Rousseau**	**Rawls**
r	Payment	Payment	Payment
m	0	1	> 0
e	0	1	0.5
strategy	*Defect*	*Cooperate*	*Quid-pro-quo*

Therefore, with the adaptation of these two general models what we have is this:

1. When $m = e = 0$, $C = r$; i.e., Hobbes agents behave as egoist agents
2. When $m = e = 1$, $C \geq r$; i.e., Rousseau agents behave as altruist agents

3. When $m > 0$ and $e \geq 0$, $C > r$; i.e., Rawls agents behave as reciprocator agents

We suggest the environment defined by a payoff matrix based upon the traditional Prisoner's Dilemma (Table 3). The idea is this: given two agents, if both cooperate, both win 3 points, that is to say, $r = 3$ for each one of them. If both defect, $r = 2$ for both. If one cooperates and the other defects, the one that defects gets 5 points (a phenomenon currently known as *temptation to deffect*) and the one that cooperates gets 0 points (which is known as *sucker's payoff*).

Table 3. Environment. Payoff matrix based upon Prisoner's dilemma

Payment	**Cooperate**	**Defect**
Cooperate	3,3	5,0
Defect	0,5	2,2

So, with these components we have the general specification of the agents and the environment in which they will unfold to help us understand a problem of political philosophy.

4 Experiments and Discussion

4.1 Experiments and Results

With this model in mind we have conducted a series of experiments implemented in an agent-based simulation fashion. We have developed 4 experiments. Each one of them has a set of agents and an environment implemented in Jason [4]. We use this BDI approach not only because of its known advantages [10], but also because these proposals of political philosophy use explanations in terms of the beliefs, desires and intentions of the agents.

Briefly, the experiments are matches that confront:

1. 5 Hobbes agents vs 5 Rousseau agents
2. 5 Rousseau agents vs 5 Rawls agents
3. 5 Hobbes agents vs 5 Rawls agents
4. 5 Hobbes agents vs 5 Rawls agents vs 5 Rousseau agents

If everything goes in order what we would expect to see should be stated by the next propositions in terms of earnings (i.e., in terms of payments or the variable r):

1. Hobbes agents get more points than Rousseau agents.
2. Rousseau and Rawls agents win more or less the same number of points.
3. Hobbes and Rawls agents win more or less the same number of points.

4. In a Free-for-All scenario the earnings increase in the next order: Rousseau, Rawls, Hobbes.

But we should see the next behaviour in terms of social commitment (C):

1. Hobbes agents have less commitment than Rousseau agents.
2. Rousseau and Rawls agents have more or less the same commitment.
3. Hobbes and Rawls agents have more or less the same commitment.
4. In a Free-for-All scenario the social commitment decreases in the next order: Rousseau, Rawls, Hobbes.

Indeed, in a Free-for-All scenario the degree of social commitment decreases in that order. From 5 runs of 2000 iterations we have found that social commitment is, more or less, like the inverse of earnings: the bigger the earnings, the lesser the commitment, and vice versa (Figure 1).

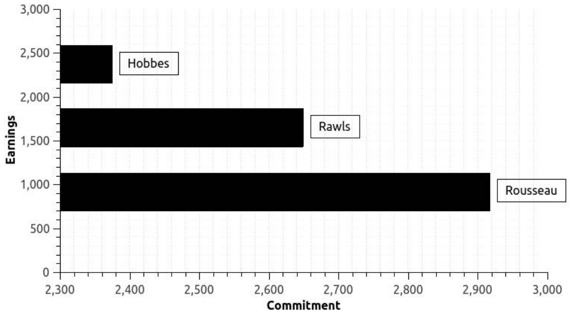


Fig. 1. Averages from 5 runs of 2000 iterations. Experiment 4

4.2 Discussion

This is clear: in terms of earnings we already know that the strategies ALLD (egoist), ALLC (altruist) and Tit-4-Tat (*quid pro quo*) behave as supposed [1,3]; but they do not explain the causes or consequences of the counterpart of earnings: social commitment between agents. Since we can observe that:

1. In Rousseau agents $C = r + s + 1$
2. In Rawls agents $C = r + \frac{s}{2} + 1$
3. And in Hobbes agents $C = r$

we argue that the important value is not that of payments (r) or morality (m), but rather the value of the product of equity (e) and social interactions s (whether they are total, null or different from zero), that is to say, the x factor that explains this trade-off is the relation between social interaction (*Sociality*) and their nature (*Responsibility*).

Thus, it would seem quite reasonable to conclude that Rousseau (altruism) provides a more plausible explanation of which form of government is preferable; however, there is no sustainable state without competition. If we consider competition as the input, Hobbes (egoism) is the undisputed winner, but high competition is condemned to failure since there is no sustainable state without social integration. If we look for a trade-off between competition and commitment we can find that Rawls (reciprocation) is more suitable. This is indeed what is expected, but the causes of this trade-off should not be put in the earnings or the morality of the agents, but rather in their strategy and in the quality of their social interactions.

4.3 Digresion about the Question of Welfare

The relevance of the previous result comes in handy when we consider the fifth variable of Shook's taxonomy. According to that taxonomy, a form of government based upon type Rousseau agents would justify highly united societies closer to communal forms of government (closer to forms of anarchy); on the other extreme, based upon type Hobbes agents, the justified form of government is that of highly competitive communities closer to liberal forms of government. However, using the previous argumentation, they would not only necessarily imply societies with a plausible basis. On the contrary, it seems that a political philosophy that prescribes reciprocation in terms of social commitment would tend to maintain higher levels of plausibility (based on type Rawls agents). What does this plausibility means?

We have used a series of undefined terms (mobility, integration, and so on) to defend the trade-off that favors reciprocation and its respective form of government: a form closer to a democracy. Actually we can attempt to define these important concepts using Bunge's model [7].

The first concept is that of social participation π. In this work we treat social participation as the quotient of agents' interactions i and social commitment C. The idea is that that mere social connections do not imply participation if there is no social commitment distributed, thus, participation is modeled as $\pi = \frac{i}{C}$. Social mobility μ is the ability of the agents of a community to move between different social groups, where N is the total population, M_{ij} is a quantitative matrix that describes the social groups that compose a community and T is the trace that denotes the rate of changes in mobility. So, social mobility is defined as $\mu = \frac{1}{N}\sum_{i\neq j} M_{ij} - T$. Social integration ι, on the other hand, is defined as the counterpart of participation by $\iota = 4\pi(1-\pi)$. Social cohesion κ is understood, then, as the average of social integration and mobility: $\kappa = \frac{\mu+\iota}{2}$. Finally, sustainability σ is the product of cohesion and efficiency $\sigma = \kappa\epsilon$, where the efficiency ϵ is to be determined by any particular index.

So let us see some extra assumptions. For the case of Hobbes we have set ϵ to 0 since it is rather a criminal society [7], by extension we have used $\epsilon = e$. Further, let us assume, for sake of experiment, that the communities are maximally mobile ($\mu = 1$). We define welfare $\omega = 4\sigma(1-\sigma)$. Then we can see the behaviour of Table 4.

Table 4. The question of welfare

Type	π	μ	ι	κ	ϵ	σ	ω
Hobbes	0.84	1	0.53	0.76	0	0	0
Rousseau	0.68	1	0.86	0.93	1	0.93	0.26
Rawls	0.75	1	0.73	0.61	.5	0.30	0.84

We have modeled welfare in this sense because when sustainability is total or null, it resembles of social participation, which would only favor welfare up to a certain degree after which it will impact negatively the former. The reason is that, when too many agents compete for the same resources and everyone meddles in the affairs of others, it ceases to be advantageous [7].

What these results indicate is that, although some political philosophies would justify highly united societies or highly competitive communities, they would not necessarily imply societies with a reasonable level of welfare (Figure 2). Actually quite the contrary, it seems that a political philosophy that prescribes reciprocation in terms of social commitment tends to maintain higher levels of welfare.

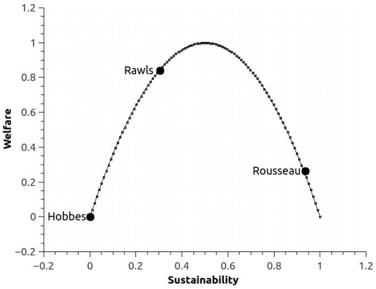


Fig. 2. The question of welfare

5 Conclusion

Computational political philosophy provides quantitative evidence about questions that are usually treated on opinion grounds, but also offers the possibility to interpret facts that we cannot directly observe but that we suspect, for instance, in this case, that reciprocation and forms of government closer to a democracy are better in terms of social optimization. That is the flavor of computational political philosophy.

Plus, the results obtained so far, if we go back to Shook's taxonomy, show us that the consequences of a political philosophy are highly entrenched with the concepts of humanity, morality and sociality of the individual agents: that is why it is extremely important to consider the precise nature of human beings within the uses and form of governments.

As part of current and future work we are conducting two kind of studies: *i)* research of comparative political philosophy (for instance, what results can we obtain from modelling Plato's communties (cast society) or Locke's societies (democratic and liberal)?) and *ii)* implementation of more complex behaviour in terms of learning and adaptation [15].

Acknowledgments. The author would like to thank the anonymous reviewers for their helpful comments and precise corrections; and the School of Philosophy at UPAEP for all the assistance.

References

1. Axelrod, R.: The Evolution of Cooperation. Basic Books, New York (1984)
2. Bandini, S., Manzoni, S., Vizzari, G.: Agent Based Modeling and Simulation: An Informatics Perspective. Journal of Artificial Societies and Social Simulation 12(4), 4 (2009), `http://jasss.soc.surrey.ac.uk/12/4/4.html`
3. Bazzan, A., Bordini, R.H., Campbell, J.A.: Agents with Moral Sentiments in an Iterated Prisoner's Dilemma Exercise. In: Proceedings of the AAAI Fall Symposium on Socially Intelligent Agents, November 8-10 (1997)
4. Bordini, R.H., Hübner, J.F., Wooldridge, M.: Programming Multi-Agent Systems in AgentSpeak using Jason. Wiley, England (2007)
5. Bratman, M.: Intention, Plans, and Practical Reason. Harvard University Press, Cambridge (1987)
6. Bunge, M.: Philosophical dictionary. Prometheus Books (2003)
7. Bunge, M.: Political philosophy. Transaction Publisher (2009)
8. Copleston, F.: A History of Philosophy (9 volumes reissued). Image Books, New York (1993-1994)
9. Dennett, D.: The Intentional Stance. MIT Press, Cambridge (1987)
10. Georgeff, M., Pell, B., Pollack, M., Tambe, M., Wooldridge, M.: The Belief-Desire-Intention Model of Agency. In: Müller, J.P., Rao, A.S., Singh, M.P. (eds.) ATAL 1998. LNCS, vol. 1365, pp. 1–10. Springer, Heidelberg (1998)
11. Gilbert, N., Doran, J. (eds.): Simulating Societies: the Computer Simulation of Social Phenomena. UCL Press, London (1994)
12. Gilbert, N., Conte, R. (eds.): Artificial Societies: the Computer Simulation of Social Life. UCL Press, London (1995)
13. Goodin, R.E., Pettit, P. (eds.): Contemporary political philosophy An anthology. Blackwell Publishers Ltd. (1997)
14. Hobbes, T.: Leviathan. Hackett, Indianapolis (1994 [1651/1668]) Edwin Curley (ed.)
15. Kinny, D., Georgeff, M.: Commitment and effectiveness of situated agents. In: Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI 1991), Sydney, Australia, pp. 82–88 (1991)
16. Locke, J.: The Two Treatises of Civil Government. Hollis ed. (1689)
17. Marx, K., Engels, F.: Selected Works, vol. 1. Progress Publishers, Moscow (1969)
18. Millner, D.: Political philosophy. A Very Short Introduction. Oxford University Press (2003)

19. Rao, A.S., Georgeff, M.P.: Modelling Rational Agents within a BDI-Architecture. In: Huhns, M.N., Singh, M.P. (eds.) Readings in Agents, pp. 317–328. Morgan Kaufmann (1998)
20. Rawls, J.: A Theory of Justice. Harvard University Press, Cambridge (1971/1999)
21. Rousseau, J.J.: The Basic Political Writings. Trans. Donald A. Cress. Hackett Publishing Company (1987)
22. Shook, J.R.: Comparative political philosophy categorizing political philosophies using twelve archetypes. Metaphilosophy 40(5), 0026–1068 (2009)

Exploring the Solution of Course Timetabling Problems through Heuristic Segmentation

Dulce J. Magaña-Lozano, Santiago E. Conant-Pablos, and Hugo Terashima-Marín

Tecnológico de Monterrey, Campus Monterrey
Av. Eugenio Garza Sada 2501, Monterrey, N.L. 64849 México
{dj.magana.phd.mty,sconant,terashima}@itesm.mx

Abstract. The aim of hyper-heuristics is to solve a wide range of problem instances with a set of heuristics, each chosen according to the specific characteristics of the problem. In this paper, our intention is to explore two different heuristics to segment the Course Timetabling Problem (CTT) into subproblems with the objective of solving them efficiently. Each subproblem is resolved as a Constraint Satisfaction Problem (CSP). Once the CTT is partitioned and each part solved separately, we also propose two different strategies to integrate the solutions and get a complete assignment. Both integration strategies use a Min-Conflicts algorithm to reduce the inconsistencies that might arise during this integration. Each problem instance was solved with and without segmentation. The results show that simple problems do not benefit with the use of segmentation heuristics, whilst harder problems have a better behavior when we use these heuristics.

Keywords: Constraint Satisfaction Problem (CSP), Course Timetabling Problem (CTT), heuristic, Min-Conflicts.

1 Introduction

In this paper, we focus on the Course Timetabling Problem (CTT), also known as University Timetabling Problem. The CTT occurs in higher educational institutions and consists in setting a sequence of meetings between teachers and students at predetermined periods of time (typically one week) satisfying a set of constraints [1].

CTT is considered interesting because in each school year the higher educational institutions spend days to build their schedules, which sometimes does not happen to be the best suited to the needs of their students. The building of a schedule can be extremely difficult and its manual solution may require considerable effort.

A wide variety of techniques have been proposed to solve the CTT, such as operations research, human-computer interaction and artificial intelligence. Some used techniques include tabu search, genetic algorithms, neural networks, simulated annealing, memetic algorithms, expert systems, constraint satisfaction, integer linear programming, and decision support systems.

In this paper, we are exploring two different heuristics to decompose CTTs into a set of subproblems that can be resolved in an efficient way. Each subproblem is

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 347–358, 2013.

solved as a CSP. We designed two different strategies to integrate the solutions of the independently solved CTT partitions into a complete assignment. The Min-Conflicts algorithm was used in both integration strategies to reduce the assignment inconsistencies that arise as result of these procedures.

We use two sets of benchmark problems to analyze the behavior of the segmentation heuristics and integration strategies. Besides, we solved these problems in the traditional way, i.e. without segmenting the problem into subproblems. Our experimental results show that some CTTs are better solved by using a segmentation approach, but not all them. Also, for the cases where partitioning the original problem was recommended, none of the segmentation heuristics nor integration strategies appeared to be dominant. This suggests a potential benefit of using a hyper-heuristic approach to decide the best segmentation heuristic and integration strategy to use during the process of solving each problem instance.

This paper is organized as follows: in section 2, the CTTs are formulated with their respective restrictions; in section 3, the CTT is defined as a CSP and is described the algorithm used to solve it; in section 4, we explain the heuristics implemented to segment the problem and strategies created to integrate the subproblems; in section 5, we present the experiments and the results obtained with the different segmentation heuristics and integration strategies; and finally, in section 6 the conclusions and future work are suggested.

2 Problem Definition

A CTT is the process of assigning courses to a limited number of time periods (timeslots) and rooms subject to a variety of restrictions and preferences. CTT is well-known as a NP-hard optimization problem [2].

Furthermore, CTT is a major and regular administrative activity in most academic institutions. It is important to mention that most universities employ the knowledge and experience of expert personnel for generating a good timetable that satisfy all given (and sometimes, conflicting) requirements [2]. In spite of this, this activity consumes a lot of time and even more when is done without an automated system.

A CTT involves a set of events (courses) $E = \{e_1, e_2, \dots, e_n\}$, a set of timeslots $T = \{t_1, t_2, \dots, t_s\}$, a set of places $P = \{p_1, p_2, \dots, p_n\}$ and a set of "agents" $A = \{a_1, a_2, \dots, a_o\}$ [3]. Each member of E is a unique event that requires the allocation of a timeslot, a place and a set of agents (students). Therefore, an assignment is a quadruple (e, t, p, a) such that $e \in E$, $t \in T$, $p \in P$ and $a \in A$ with the interpretation "the event e starts at the time t in the place p involving the agents a". Shortly, a class schedule is simply a collection of n assignments, one for each event. The problem is to find a class schedule that meets or minimizes the violations to the full collection of constraints (restrictions and preferences).

Two types of constraints are usually considered in a CTT: the hard constraints and the soft constraints. Hard constraints have a higher priority than soft, and cannot be violated under any circumstances. A timetable only can be considered "feasible" if all hard constraints have been satisfied. On other hand, soft constraints are desirable but not essential, and are usually used to evaluate how good the feasible timetables are. For real-world CTT instances, it is typically impossible to find solutions satisfying all the soft constraints. Indeed, sometimes it is very difficult to just find a feasible solution [4].

In this paper, only hard constraints are considered and they are the following:

- Courses with students in common cannot be programmed in the same period of time.
- Do not exceed the capacity of the classroom and it must be suited to the needs of the course.
- Only one course can be assigned to a classroom in a specific period of time.

3 Solution as a CSP

Many problems can be formulated as a Constraint Satisfaction Problem (CSP) [5]. To solve the instances of the CTT as a CSP, first it is necessary to define which are the variables, their domains and the constraints on these variables. Therefore, these problems can be represented as a tuple (V, D, R) where [6]:

- V: It is a finite set of courses that a university should schedule $V = \{v_1, v_2, \ldots, v_n\}$. This represents the variables of the CSP.
- D: It is a finite set of domains of variables $D = \{d_1, d_2, \ldots, d_n\}$. For this problem, the domain of a variable is the Cartesian product of two sets, where each set represents a different resource. The first set is $S = \{s_1, s_2, \ldots, s_m\}$ that represents the classrooms where a course can be scheduled and the second set is $P = \{p_1, p_2, \ldots, p_t\}$ that are the timeslots in which can be scheduled. All the timeslots have the same duration. Therefore, the domain of a variable i is given by $d_i \subseteq S \times P$.
- R: It is a finite set of constraints to be satisfied $R = \{r_1, r_2, \ldots, r_q\}$. The instances of the CTT that are being solved have three hard constraints which were mentioned in the previous section.

3.1 CSP Algorithm

The constraint satisfaction algorithm in this research performs a chronological backtracking search that implements a Depth First Search (DFS) in which values are chosen for a variable (one at a time) and when a variable does not have valid values to assign, returns to the previously instantiated variable and tries a different value for it (backtrack).

The CTT instances that are solved have a considerable amount of variables with different sizes of domain. Because of this, it is convenient to apply heuristics to dynamically select the next variable to instantiate and the next value to assign, as well as applying a look-ahead algorithm to effectively reduce the search space of the problem, and with this, also the time to find a solution. Below are the heuristics and look-ahead algorithm used in this work.

Heuristics for Variable and Value Selection. The Fail First (FF) heuristic was chosen to select the assignment order for the variables. This heuristic is also known as Minimum Remaining Values (MRV). It selects the variable more "restricted", i.e. one that has the smallest domain, to realize the next value assignment. Once you have selected the variable to instantiate is important to decide what value will be assigned.

The Least Constraining Value (LCV) heuristic was chosen to select the assignment order for the values. This heuristic selects the value that least reduces the domain of unassigned variables, i.e. the value that in a future generates less conflicts with the variables that have not yet been assigned. When two or more values reduce in the same size the domain of unassigned variables, one of these values is randomly chosen, for this reason, this heuristic may produce different solutions for each run.

Look-ahead Algorithm. The look-ahead algorithm does a forward checking of the still unassigned variables' domains in each step of the search. During this process, the values in the domains of unassigned variables that enter in conflict with the last variable assigned are removed. Thus, the situations that do not lead to a solution can be identified beforehand and therefore do not waste time in testing them in the future.

4 CTT Segmentation and Solution Integration

As mentioned in section 1, the objective of this paper is to explore the convenience of segmenting a CTT into subproblems with the aim of solving these in an efficient way. To accomplish this objective we designed and implemented two different segmentation heuristics and two different strategies to integrate the solution. Section 4.1 explains the heuristics implemented to partition the courses of the CTT in groups, while section 4.2 explains the strategies developed to integrate the solutions obtained independently for each group.

4.1 Heuristics for CTT Segmentation

In this research we designed two heuristics to segment a CTT by forming disjoint sets with its courses. The example used to explain in a better way the heuristics is shown in Fig. 1, where each node (or vertex) represents a course and each edge (or link) a constraint. The numbers on the edges represent the number of students that two courses have in common.

In the following sub-sections we explain the implemented heuristics.

Vertex Degree Block Segmentation Heuristic (VDH). First of all, the degree of a vertex v in a graph G is the number of vertices in G that are adjacent to v. Equivalently, the degree of v is the number of edges incident to v [7]. Then, this heuristic sorts in a decreasing manner the courses (vertices) by their degree. After the courses were sorted, the segmentation of variables is done according to the desired number of blocks, such that in the first block will be the courses more restricted (with the largest degree), and in the last block the courses less restricted (with the smallest degree). The size of each block is given by Equation 1.

$$\text{block size} = \lceil \#\text{ of courses} / \#\text{ of blocks} \rceil \tag{1}$$

When the courses are being ordered, it is possible that ties exist, i.e., that more than one course has the same degree. The way in which we decide what course is first and which later is based on a course identifier, which is sorted in an ascending order.

Suppose we want to segment the example of the Fig. 1 into 2 blocks of variables. The first step is to sort the variables by their degree; this arrangement is shown in Table 1. Next, using Equation 1 the size of each block is obtained, getting a size of 3. Finally, the block 1 would consist of the courses C2, C0 and C1, while in the block 2 would be the courses C3 and C4.

Table 1. Courses sorted by their degree

Vertex	Degree
C2	3
C0	2
C1	2
C3	2
C4	1

A disadvantage of this approach is that variables in a block are unrelated among them and related with variables in other blocks, so at the moment of integrating this block with others, many of its assignments will produce conflicts with the assignments of the other blocks, as consequence of the local solution of the subproblems.

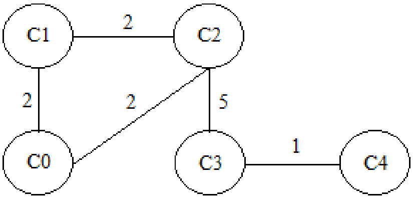


Fig. 1. Graph of a simple example of a CTT

Edge Weight Block Segmentation Heuristic (EWH). This heuristic accommodates each course in the block where it has more students in common with the block's representative course. The procedure to determine where each course may be included is as follows:

- The first step is to generate a relationship matrix that quantifies and shows the number of students in common for each pair of courses. Continuing with the example of Fig. 1, the relationship matrix that is obtained is shown in Table 2.

Table 2. Relationship matrix

	C0	C1	C2	C3	C4
C0	-	2	2	0	0
C1	2	-	2	0	0
C2	2	2	-	5	0
C3	0	0	5	-	1
C4	0	0	0	1	-

- Then, we select m as the number of desired blocks and select the m most constrained courses, i.e. courses with the highest degrees. Ties are resolved with the strategy previously explained. Each of the selected courses is assigned to a different block, and all other courses will be compared with these courses to decide in which block to assign them. Continuing with the example of Fig. 1, consider that it was decided to segment the courses of the CTT in two blocks ($m = 2$), then the two most restricted courses are C2 and C0, so each of them will be placed in a different block, as shown in Table 3, where 1 means that it belongs to the group and 0 does not belong to the group.

Table 3. Courses more restricted that represent the blocks of variables

	C0	C1	C2	C3	C4
G1 (C2)	0		1		
G2 (C0)	1		0		

- The next step is to place the other courses in the group where there exists a greater number of students associated with the course that represents that group. Based on Table 3 the following course to assign would be C1, to decide in which group to place it we can use the Table 2 which shows that the course C1 has 2 students in common with the course C0 and C2, but, as is the same number of students, the decision is to select the first course detected, in this case C0. On the other hand, the C3 course is placed in the group of the C2 course because there are 5 students in common, and in the group of the C0 course not have any student in common. Finally, the C4 course is not placed in any group because it has no relation to either the C2 course or the C0 course. In the Table 4 is shown the 2 groups of variables to be solved by the CSP.

Table 4. Groups of variables to be solved by the CSP

	C0	C1	C2	C3	C4
G1 (C2)	0	0	1	1	0
G2 (C0)	1	1	0	0	0

- The last step, if needed, is to create an additional block, where we can allocate all isolated courses, i.e., courses of degree 0. For example, if in the Fig. 1 there were a C5 course without any edge, then this course belongs to this block. In this same block are placed the courses that do not belong to any of the groups from the previous step. The C4 course in Fig. 1 would be in this block, this can be validated with Table 4.

A disadvantage of this heuristic is that the block size can vary considerably, because it depends on how the students are distributed in the courses that represent the blocks.

4.2 Strategies for Solution Integration

After the courses have been segmented into blocks and have been resolved by the CSP, it is necessary to integrate the blocks into one complete solution, but this could uncover conflicts between assignments to variables of different blocks.

To reduce the number of inconsistencies generated by the union of the blocks of variables, we are using a Min-Conflicts local search. This algorithm produces a list of variables in conflict and randomly chooses one of them to assign a different value. The value assigned to the variable that which generates the minimum number of conflicts with other variables, namely, the one that minimizes the number of unsatisfied hard constraints. This algorithm is repeated a number of times (attempts) in order to gradually reduce the number of conflicts that exist between the variables, and eventually find a feasible solution. We decided to use this algorithm because it has been proven effective for many CSPs and works well for hard problems [8].

To integrate the independently obtained blocks solutions into a single complete solution we used 2 different strategies; these strategies are explained in the following subsections.

Full Integration of Blocks (FIB). This strategy first, independently solves all the blocks with the CSP algorithm; then, it joins all the blocks solutions in one full, and usually, unfeasible assignment; and finally, this complete assignment is passed to the Min-Conflicts local search in order to eliminate, or at least to reduce, the inconsistencies generated by the union of all blocks.

Sequential Integration of Blocks (SIB). This strategy gets the complete CTT solution by sequentially forming a block, independently solving it with the CSP algorithm, and then integrating its solution to the last partially integrated solution of its preceding blocks by using the Min-Conflicts local search. The process stops when all the blocks solutions have been integrated. It is noteworthy that the blocks are formed and integrated in the order they are created by the segmentation heuristics.

5 Experiments and Results

In this section are the experimental results obtained with two sets of CTT instances. Such instances were resolved through the previously depicted CSP algorithm and with the different segmentation heuristics and integration strategies. The presented results were obtained by solving multiple times each CTT instance. A maximum runtime for each experiment was established in 1,000,000 milliseconds that is equivalent to 16.7 minutes. This limit was decided experimentally after trying with 10,000,000 milliseconds and observing that the results did not vary significantly. It is important to mention that when the runtime is over all the variables that have not been instantiated will be assigned a value chosen randomly with the objective of having a complete assignment and to determine the number of unsatisfied constraints.

The experiments have been run on a PC with an Intel Core i7 2.20 GHz processor and 6 GB of RAM, using Windows 7 and Java 1.7

5.1 Sets of CTT Instances

In this research, two types of CTTs were used to test and validate the proposed heuristics. All problems consist of a set of events E to be scheduled in 45 time periods (5 days 9 hours each), a set of S rooms in which events will be taught, a set of students A that attend to events and a set of characteristics C that have the rooms and are required by the events. In addition, each student attends a variable number of events and each classroom has a specified maximum capacity.

The first type of problems, which we refer to as Type I problems, was proposed by the International Timetabling Competition [9] and consists of a set of 20 CTT instances. The second type of problems, which we refer to as Type II problems, was proposed by Rhyd Lewis and is a set of 60 CTT instances [10]; these problems are particularly "harder" than Type I problems while looking for a feasible timetable. Both types of problems use an automated instance generator by Ben Paechter.

We decided to do experiments only with instances that had 400 courses. The reason of this is because both types of problems have instances with this number of courses, so this let us to compare the results in an equitable way. In the case of Type I problems, 11 instances were used which represents the 55% of the full benchmark set of these problems. Also in Type II problems, 11 instances were used which represents the 18.33% of the full benchmark set of these problems.

For each instance, we calculated three measures to assess the complexity of the instances. These measures are explained below:

- Density (p_1). It represents the proportion of edges in the constraint graph that describe the CTT [11]. The Equation 2 shows how to calculate this factor in an undirected graph, like is our case.

$$p_1 = (2 * |E|)/(|V| * (|V| - 1)) \tag{2}$$

- Average tightness ($\overline{p_2}$). The *tightness* of a constraint is defined as the proportion of all possible pairs of values from the domains of two variables that are not allowed by the constraint. The *average tightness* of a CSP is then the average tightness of all constraints in the CSP [12]. It is important to mention that all the instances used in our experiments have the same average tightness of **0.02**.
- Average domain size ($\overline{p_3}$). It represents the average of the proportion of the domain size of all variables of a CSP. The proportion of the domain size of a variable v_i is calculated as $\frac{|d_i|}{|D|}$.

Table 5 presents the characteristics and measures of complexity calculated for the Type I and Type II instances. As all instances have 400 courses, 10 rooms, and 0.02 of average tightness, we decided not to show this information in the Table 5.

Table 5. Characteristics and measures of complexity of the instances

	Type I Problems					Type II Problems			
Instance	C	A	p_1	$\overline{p_3}$	Instance	C	A	p_1	$\overline{p_3}$
comp1	10	200	0.18	0.20	med1	10	400	0.23	0.14
comp2	10	200	0.18	0.19	med8	10	400	0.37	0.19
comp3	10	200	0.19	0.34	med9	10	400	0.38	0.12
comp4	5	300	0.21	0.24	med10	8	500	0.24	0.51
comp8	5	250	0.16	0.29	med11	8	800	0.35	0.45
comp10	5	200	0.19	0.35	med12	8	800	0.26	0.34
comp11	6	220	0.18	0.21	med13	8	800	0.35	0.42
comp12	5	200	0.18	0.20	med14	8	1000	0.32	0.49
comp13	6	250	0.18	0.24	med16	8	1000	0.50	0.52
comp18	10	200	0.17	0.18	med17	8	800	0.43	0.40
comp19	5	300	0.20	0.39	med18	8	1000	0.63	0.46

5.2 How to Calculate the Number of Blocks in the Instances

The number of blocks in each instance is calculated depending on the segmentation heuristic used. We use Equation 3 for experiments that apply the Vertex Degree segmentation heuristic.

$$\#\text{ of blocks} = \left\lfloor \sqrt{\#\text{ of courses}} \right\rfloor \tag{3}$$

On the other hand, we use the Equation 4 for experiments where the courses are segmented by the Edge Weight heuristic. For this heuristic, we calculate the number of blocks based on the number of courses that are related with other courses, i.e. all courses that had more than 0 edges. It is important to remember that if there are courses that are not related to other courses or do not belong to any of the blocks, it is necessary to create an additional block where these courses are resolved, as mentioned in section 4.1.

$$\#\text{ of blocks} = \left\lfloor \sqrt{\#\text{ of related courses}} \right\rfloor \tag{4}$$

5.3 Results of Type I Problems

In Table 6, we show the obtained results of applying the different solution approaches to solve the Type I problems. In this table and subsequent tables, the results under the T1 heading are those obtained by solving the CTT instances without segmentation; under T2 are the results obtained by segmenting the CTT instances using the Vertex Degree heuristic and integrating all the solutions at the same time; under T3 are the results from Vertex Degree segmentation and sequential integration of blocks solutions; under T4 appear the results obtained by segmentation with the Edge Weight heuristic and integration of all blocks solutions at the same time; and under T5 are the results from Edge Weight heuristic segmentation and sequential integration of solutions.

The purpose of Table 6 is to observe how the studied solution alternatives behave in relation to the number of unsatisfied hard constraints (UHC) and execution time (ET) measured in minutes.

Table 6. Results of Type I Problems using the different solution techniques

	Type I Problems									
	T1		T2		T3		T4		T5	
Instance	UHC	ET	UHC	ET	UHC	ET	UHC	ET	UHC	ET
comp1	0	0.21	0	0.81	0	0.38	0	0.91	0	0.48
comp2	0	0.24	0	0.33	0	0.31	0	0.36	0	0.35
comp3	0	0.52	0	0.8	0	0.67	0	0.85	0	0.78
comp4	0	0.33	0	0.76	0	0.57	0	0.71	0	0.72
comp8	0	0.39	0	0.95	0	0.67	0	0.92	0	0.82
comp10	0	0.38	0	3.94	0	4.26	0	6.37	0	2.90
comp11	0	0.25	0	0.65	0	0.87	0	0.84	0	0.70
comp12	0	0.21	4	13.52	0	1.04	1	13.67	2	11.14
comp13	0	0.28	0	1.82	0	1.34	0	1.33	0	1.48
comp18	0	0.21	0	0.38	0	0.32	0	0.33	0	0.38
comp19	0	0.76	0	1.88	0	1.72	0	1.23	0	1.43

The results of this experimentation show us how T1 is capable to solve all Type I instances in a better runtime, this in comparison to the other techniques. Besides, it is important to note that T3 also was capable of solving all the instances, although the runtime increases considerably.

5.4 Results of Type II Problems

In Table 7, we show the obtained results of applying the different techniques to solve the Type II problems. As in the previous section, the aim of this table is to observe the behavior of the different solution approaches in relation to the UHC and ET.

The results in Table 7 show us how T1 only can solve the med1 instance, so we can infer that as the problem density increases, T1 has much more difficulty to solve the problem in a short execution time. Besides, these results indicate that any of the segmentation-integration solution alternatives are better for solving the Type II instances.

5.5 Summary of Results

Table 8 resumes the best solution strategy resulting for each of the CTT instances, in accordance with the number of UHC and ET. In this table, we can easily observe that there is no dominant strategy.

Table 7. Results of Type II Problems using the different solution techniques

	Type II Problems									
	T1		T2		T3		T4		T5	
Instance	UHC	ET	UHC	ET	UHC	ET	UHC	ET	UHC	ET
med1	0	0.2	0	1.6	0	1.5	0	1.3	0	1.6
med8	6935	16.7	28	16.7	12	13.3	35	15.6	15	16.7
med9	12701	16.7	185	16.7	178	16.7	193	16.7	194	16.7
med10	1974	11.7	0	2.9	0	7.7	0	2.8	0	3.0
med11	18747	16.7	1	16.0	0	13.5	16	16.7	23	16.7
med12	12345	16.7	3	15.7	0	14.1	7	15.3	1	14.5
med13	30432	16.7	67	16.7	84	16.7	59	16.7	73	16.7
med14	19839	16.7	16	16.7	18	16.7	13	16.7	26	16.7
med16	31115	16.7	163	16.7	145	16.7	153	16.7	242	16.7
med17	22287	16.7	23	16.7	7	16.7	33	16.7	38	16.7
med18	40688	16.7	222	16.7	284	16.7	242	16.7	327	16.7

Table 8. Best solution strategy for each CTT instance

Type I Problems		Type II Problems	
Instance	Best technique	Instance	Best technique
comp1	T1	med1	T1
comp2	T1	med8	T3
comp3	T1	med9	T3
comp4	T1	med10	T4
comp8	T1	med11	T3
comp10	T1	med12	T3
comp11	T1	med13	T4
comp12	T1	med14	T4
comp13	T1	med16	T3
comp18	T1	med17	T3
comp19	T1	med18	T2

6 Conclusions and Future Work

This research focused on exploring two different heuristics to segment CTT instances in several subproblems and then integrate their solutions through two different integration strategies. The tests were performed on two sets of instances of the CTT, one of the sets containing instances characterized by a low density measure (Type I instances), usually considered easier problems, while the other set contained instances with a high density measure, common in harder problems. To validate the advantages of segmenting a problem into subproblems, it was first necessary to observe the behavior of the problems when they are not segmented.

The results of the experimentation show us that some problems are solved better if they are not segmented as in the case of Type I instances, while other problems such as Type II instances are solved in a better way when they are segmented, resulting in smaller number of unsatisfied constraints and shorter time of execution. The experimental results also indicate that none of the solution strategies dominated the others, suggesting the convenience of implementing a hyper-heuristic approach that could decide when and which segmentation heuristic and integration strategy to use, depending on the characteristics of the CTT instance to solve, such as density, tightness, domain size, etc.

Some ideas to extend this research include: 1) try with different number of blocks, 2) handle a minimum and maximum default block size, when we segmented the problem by the weight of the edges and 3) generate a hyper-heuristic system that tells us the best segmentation heuristic and integration strategy to use for each CTT instance.

Acknowledgments. This research was supported in part by ITESM under the Research Chair CAT-144 and the CONACYT Project under grant 99695.

References

1. Lee, J., Yong-Yi, F., Lai, L.F.: A software engineering approach to university timetabling. In: International Symposium on Multimedia Software Engineering, pp. 124–131. IEEE Press (2000)
2. Adewumi, A.O., Sawyerr, B.A., Ali, M.M.: A heuristic solution to the university timetabling problem. Engineering Computations 26, 972–984 (2009)
3. Adamidis, P., Arapakis, P.: Evolutionary algorithms in lecture timetabling. In: Congress on Evolutionary Computation, pp. 1145–1151. IEEE Press (1999)
4. Qu, R.R., Burke, E.K.: Hybridizations within a graph-based hyper-heuristic framework for university timetabling problems. Journal of the Operational Research Society 60, 1273–1285 (2009)
5. Tsang, E.: Foundations of Constraint Satisfaction. Academic Press, London (1993)
6. Conant-Pablos, S.E., Magaña-Lozano, D.J., Terashima-Marín, H.: Pipelining Memetic Algorithms, Constraint Satisfaction, and Local Search for Course Timetabling. In: Aguirre, A.H., Borja, R.M., Garciá, C.A.R. (eds.) MICAI 2009. LNCS, vol. 5845, pp. 408–419. Springer, Heidelberg (2009)
7. Chartrand, G., Lesniak, L., Zhang, P.: Graphs & Digraphs. CRC Press, Boca Raton (2010)
8. Rusell, S., Norving, P.: Artificial Intelligence: A Modern Approach. Prentice-Hall, Englewood Cliffs (2010)
9. Metaheuristics Network, `http://www.idsia.ch/Files/ttcomp2002/`
10. New "harder" instances for the university course timetabling problem, `http://www.soc.napier.ac.uk/~benp/centre/timetabling/harderinstances`
11. Terashima-Marín, H., Ortiz-Bayliss, J.C., Ross, P., Valenzuela-Rendón, M.: Using Hyper-heuristics for the Dynamic Variable Ordering in Binary Constraint Satisfaction Problems. In: Gelbukh, A., Morales, E.F. (eds.) MICAI 2008. LNCS (LNAI), vol. 5317, pp. 407–417. Springer, Heidelberg (2008)
12. Craenen, B.G.W., Paechter, B.: A Conflict Tabu Search Evolutionary Algorithm for Solving Constraint Satisfaction Problems. In: van Hemert, J., Cotta, C. (eds.) EvoCOP 2008. LNCS, vol. 4972, pp. 13–24. Springer, Heidelberg (2008)

Verifying Nested Workflows with Extra Constraints

Roman Barták and Vladimír Rovenský

Charles University in Prague, Faculty of Mathematics and Physics
Malostranské náměstí 25, 118 00 Praha 1, Czech Republic
bartak@ktiml.mff.cuni.cz, v.rovensky@lit.cz

Abstract. Nested workflows are used to formally describe processes with a hierarchical structure similar to hierarchical task networks in planning. The nested structure guarantees that the workflow is sound in terms of possibility to select for each involved activity a process that contains the activity. However, if extra synchronization, precedence, or causal constraints are added to the nested structure, the problem of selecting a process containing a given activity becomes NP-complete. This paper presents techniques for verifying such workflows; in particular, they verify that a process exists for each activity.

Keywords: workflow, verification, constraint satisfaction, temporal consistency.

1 Introduction

Workflow management is a technology for automating work with processes. It is frequently applied to business processes and project management, but it is useful also for description of manufacturing processes. Briefly speaking, workflow is a formal description of a process typically as a set of interconnected activities. There exist many formal models to describe workflows [15] that include decision points for process splitting as well as loops to describe repetition of activities. In this paper we adapted the idea of *Nested Temporal Networks with Alternatives* (Nested TNA) [2], in particular the form used in the manufacturing scheduling system FlowOpt [3]. TNA is a directed acyclic graph with parallel and alternative splitting and joining of processes that is also known as AND-split and OR-split (and AND-join, OR-join) in traditional workflow management systems [1,14]. Hence TNA can describe alternative processes in the style similar to scheduling workflows with optional activities introduced in [5] and used also in Extended Resource Constrained Project Scheduling Problems [11]. Nested TNA requires a specific structure of the underlying task graph where the split operations have corresponding join operations. This structure is obtained when seeing the workflow as a hierarchical structure of tasks that are decomposed to sub-tasks until the primitive activities are obtained. Figure 1 gives an example of a nested workflow with three types of decompositions: parallel decomposition (*Legs*), serial decomposition (*Seat*), and alternative decomposition (*Back Support*). The figure also highlights the hierarchical (tree) structure of the nested workflow. Such a structure is quite typical for real-life workflows [1] as many workflows are obtained by decompositions of tasks. The structure is also close to the idea of hierarchical task networks leading to Temporal Planning Networks [10].

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 359–370, 2013.

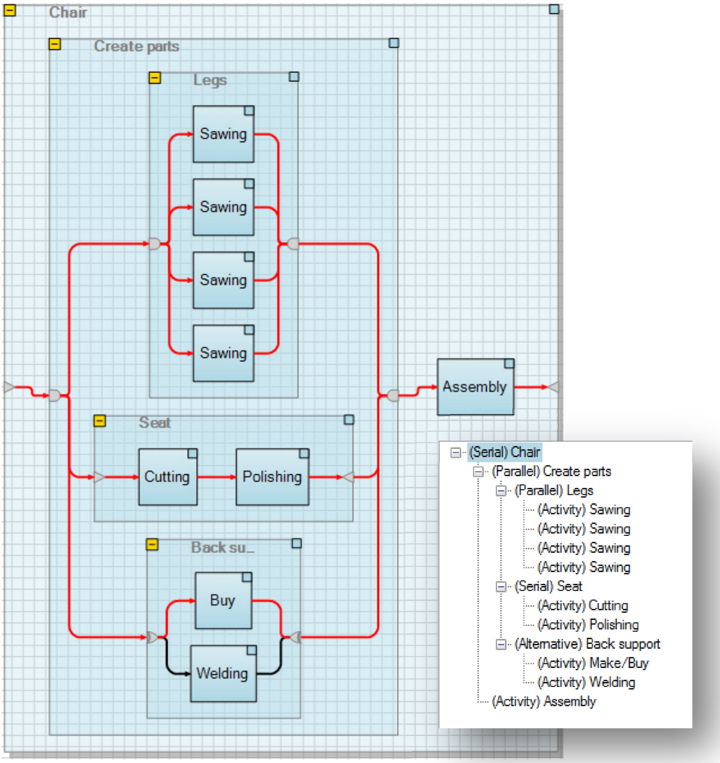


Fig. 1. Nested workflow as it is visualized in the FlowOpt Workflow Editor (from top to down there are parallel, serial, and alternative decompositions)

The hierarchical structure has a nice property that the nested workflow is always sound in the sense of having some process for each activity. Nevertheless, to model real-life problems it is beneficial to extend the core nested structure with additional constraints [1,11]. It can be easily shown that adding extra constraints can introduce flaws to the workflow. For example adding a mutex (mutual exclusion) constraint between activities *Buy* and *Cutting* to the workflow in Figure 1 causes the activity *Buy* to be inapplicable (*Cutting* must be part of any process selected from the workflow). This raises the question if such flaws in the design of a workflow can be detected automatically before the workflow is used for example to generate a manufacturing schedule. This process is called *workflow verification* and it should be an integral part of workflow management systems [9].

It has been shown that the problem whether there exists a feasible process for a nested workflow with extra synchronization, precedence, and causal constraints is NP-complete [4]. In this paper we propose a novel technique for verifying such workflows, that is, for ensuring that for each activity there exists a feasible process containing that activity. We will first formally introduce nested workflows with extra constraints and the verification problem. Then we will present the constraint-based verification technique and its improvement based on collapsing some tasks. Finally, we will present experimental comparison of both methods and their scalability.

2 Nested Workflows with Extra Constraints

The *nested workflow* is obtained from a root task by applying decomposition operations that split the task into subtasks until primitive tasks, corresponding to activities, are obtained. Three decomposition operations are supported, namely parallel, serial, and alternative decompositions. Figure 1 gives an example of a nested workflow that shows how the tasks are decomposed. The root task *Chair* is decomposed serially to two tasks, where the second task is a primitive task filled by activity *Assembly*. The first task *Create Parts* decomposes further to three parallel tasks *Legs*, *Seat*, and *Back Support*. *Back Support* is the only example here of alternative decomposition to two primitive tasks with *Buy* and *Welding* activities (*Welding* is treated as an alternative to *Buy*). Naturally, the nested workflow can be described as a tree of tasks (Figure 1). Formally, the nested workflow is a set *Tasks* of tasks that is a union of four disjoint sets: *Parallel*, *Alternative*, *Serial*, and *Primitive*. For each task T (with the exception of the root task), function *parent*(T) denotes the parent task in the hierarchical structure. Similarly for each task T we can define the set *subtasks*(T) of its child nodes ($subtasks(\mathrm{T}) = \{\mathrm{C} \in Tasks \mid parent(\mathrm{C}) = \mathrm{T}\}$). The tasks from sets *Parallel*, *Alternative*, and *Serial* are called *compound tasks* and they must decompose to some subtasks:

$$\mathrm{T} \in (Parallel \cup Alternative \cup Serial) \Rightarrow subtasks(\mathrm{T}) \neq \emptyset,$$

while the primitive tasks do not decompose:

$$\mathrm{T} \in Primitive \Rightarrow subtasks(\mathrm{T}) = \emptyset.$$

The workflow defines one or more processes in the following way. *Process* selected from the workflow is defined as a subset $\mathrm{P} \subseteq Tasks$ in the workflow satisfying the following properties:

- for each task T in the process, that is not a root task, its parent task is also in the process, $\mathrm{T} \in \mathrm{P} \wedge \mathrm{T} \neq root \Rightarrow parent(\mathrm{T}) \in \mathrm{P}$,
- for each compound task T in the process with a serial or parallel decomposition, all its subtasks are also in the process,
  $\mathrm{T} \in \mathrm{P} \cap (Serial \cup Parallel) \Rightarrow subtasks(\mathrm{T}) \subseteq \mathrm{P}$,
- for each compound task T in the process with the alternative decomposition, exactly one of its subtasks is in the process,
  $\mathrm{T} \in \mathrm{P} \cap Alternative \Rightarrow \mid subtasks(\mathrm{T}) \cap \mathrm{P} \mid = 1$.

So far we defined only the hierarchical structure of the nested workflow, but as Figure 1 shows the nested structure also defines certain temporal constraints. These temporal relations must hold for all tasks in a single process. Assume that S_T is the start time and E_T is the completion time of task *T*. The primitive tasks *T* are filled with activities and each activity has some duration D_T. Then for tasks in a certain process P the following relations hold:

$$T \in P \cap \mathit{Primitive} \Rightarrow S_T + D_T = E_T$$

$$T \in P \cap (\mathit{Parallel} \cup \mathit{Alternative} \cup \mathit{Serial}) \Rightarrow S_T = \min\{\ S_C \mid C \in P \cap \mathit{subtasks}(T)\ \}$$
$$E_T = \max\{E_C \mid C \in P \cap \mathit{subtasks}(T)\ \}.$$

Notice that the duration of compound task is defined by the time allocation of its subtasks while the duration of the primitive tasks is defined by the activity. Moreover, for the serial decomposition we expect the subtasks to be ordered, say $T_1, \ldots, T_n$, where n is the number of subtasks of a given task. The following constraints must hold if the serial task is included in the process:

$$T \in P \cap \mathit{Serial} \wedge \mathit{subtasks}(T) = \{\ T_1, \ldots, T_n\ \} \Rightarrow \forall i = 1,\ldots, n\text{-}1\text{: } E_i \leq S_{i+1}$$

A *feasible process* is a process where the time variables S_T and E_T can be instantiated in such a way that they satisfy the above temporal constraints. It is easy to realize that if there are no additional constraints then any process is feasible. The process defines a partial order of tasks so their start and end times can be easily set in the left-to-right order while satisfying all the above temporal constraints.

The nested structure may not be flexible enough to describe naturally some additional relations in real-life processes, for example when the alternative for one task influences the selection of alternatives in other tasks. The following constraints can be added to the nested structure to simplify description of these additional relations between any two tasks *i* and *j*:

- precedence constraint: $i, j \in P \Rightarrow E_i \leq S_j$
- start-start synchronization: $i, j \in P \Rightarrow S_i = S_j$
- start-end synchronization: $i, j \in P \Rightarrow S_i = E_j$
- end-start synchronization: $i, j \in P \Rightarrow E_i = S_j$
- end-end synchronization: $i, j \in P \Rightarrow E_i = E_j$
- mutex constraint: $i \notin P \vee j \notin P$
- equivalence constraint: $i \in P \Leftrightarrow j \in P$
- implication constraint: $i \in P \Rightarrow j \in P$

It is interesting that if these extra constraints are used then the existence of a feasible process is no longer obvious. In fact, the recent paper [4] shows that the problem, whether a feasible process exists or not, is NP-complete. Because the existence of a feasible process is not guaranteed, it is important to verify the workflow structure before the workflow is used further [9].

3 Workflow Verification

We formulate the *workflow verification problem* as the problem of checking that for each task there exists a feasible process containing this task. Workflow verification has been studied for some time. Various methods of verification have been proposed, e.g., using Petri Nets [14], graph reductions [13], or logic-based verification [6]. These methods deal with complex workflow structures that are used for example to model business processes. The focus of these methods is on structural properties of the workflows and they are not applicable to the extra constraints that are going beyond the core workflow structure. Recall, that the nested structure forces the workflows to be sound; it is the addition of extra constraints that may introduce flaws to the workflow. Hence we need a novel verification method that covers both the core workflow constraints (its structure) as well as the extra constrains. We will now present how to formulate the workflow verification problem as a constraint satisfaction problem. A *Constraint Satisfaction Problem* (CSP) is given by a finite set of variables, each variable taking its value from a finite set of possible values (called a domain), and a finite set of constraints restricting assignments to variables. A solution to a CSP is an instantiation of variables satisfying all the constraints [7].

To describe a feasible process we need to decide which tasks the process consists of and whether the tasks can be consistently allocated to time. Hence each task i is described by three variables: V_i, S_i, and E_i, where V_i is a 0-1 validity variable describing the presence of task i in the process, S_i is a temporal variable describing the start time of task i, and finally E_i is the end time of task i. For all activities (primitive tasks) the user specifies their fix duration D_i:

$$V_i = 1 \Rightarrow S_i + D_i = E_i$$

For the compound task, the duration is given by its subtasks (see below) so we only know that the task does not start later than it finishes:

$$V_i = 1 \Rightarrow S_i \leq E_i$$

The structure of the workflow can be specified by the following constraints that are defined based on the type of the task. Let task i decompose to subtasks $i_1, \ldots, i_k$. If task i is decomposed in a *parallel* way then we use the following constraints:

$$\forall j = i_1, \ldots, i_k\text{: } V_i = V_j$$
$$V_i = 1 \Rightarrow S_i = \min\{\, S_j \mid j = i_1, \ldots, i_k\}$$
$$V_i = 1 \Rightarrow E_i = \max\{\, E_j \mid j = i_1, \ldots, i_k\}$$

If task i is decomposed in a *serial* way then we use the following constraints:

$$\forall j = i_1, \ldots, i_k\text{: } V_i = V_j$$
$$V_i = 1 \Rightarrow S_i = S_{i_1}$$
$$V_i = 1 \Rightarrow E_i = E_{i_k}$$
$$\forall j = 1, \ldots, k\text{-}1\text{: } E_{i_j} \leq S_{i_{j+1}}$$

Finally, if task i is decomposed in an *alternative* way then we use the following constraints:

$$V_i = \Sigma_{j = i_1,\ldots,i_k} V_j$$
$$\forall j = i_1, \ldots, i_k\text{: } V_j = 1 \Rightarrow S_i = S_j$$
$$\forall j = i_1, \ldots, i_k\text{: } V_j = 1 \Rightarrow E_i = E_j$$

The above constraint model fully specifies the nested workflows. Now we need to add the extra constraints going beyond the nested structure. These constraints restrict further the tasks that are included in the process. Let us specify how these constraints look if they connect tasks i and j.

- precedence constraint: $V_i = V_j = 1 \Rightarrow E_i \leq S_j$
- start-start synchronization: $V_i = V_j = 1 \Rightarrow S_i = S_j$

- start-end synchronization: $V_i = V_j = 1 \Rightarrow S_i = E_j$
- end-start synchronization: $V_i = V_j = 1 \Rightarrow E_i = S_j$
- end-end synchronization: $V_i = V_j = 1 \Rightarrow E_i = E_j$
- mutex constraint: $V_i + V_j < 2$
- equivalence constraint: $V_i = V_j$
- implication constraint: $V_i \leq V_j$

Every instantiation of validity and temporal variables that is consistent with the above constraints defines a feasible process with tasks allocated in time. Obviously, different feasible processes differ in the instantiation of validity variables only. Notice also that an empty process, where $\forall i$: $V_i = 0$, trivially satisfies all the constraints and hence it is a feasible process though not a very interesting one from the user perspective.

With such a constraint model R, the verification problem for task *T* consists of determining whether the problem $R \cup \{V_T = 1\}$ has a solution, that is, we are looking for a feasible process containing task *T*. The workflow verification problem can be formulated as a problem of checking that problems $R \cup \{V_T = 1\}$ for all tasks *T* have a solution. The naïve approach is to solve these problems using standard constraint satisfaction techniques. However, there is a better method based on the following observations.

Observation 1: If there is a compound task in a feasible process then the same process must also contain some primitive task from the workflow rooted at that compound task. Hence, it is enough to verify just the primitive tasks; each compound task is an ancestor of some primitive task and ancestors are present in the process too. Consequently, the root task is present in all non-empty processes.

Observation 2: The valid process for some primitive task may contain other primitive tasks. Hence, the validity of such primitive tasks does not need to be verified separately as we already have a feasible process for them.

Observation 3: The process is defined by its tasks so to determine the process we need to instantiate the validity variables. It is interesting to see that temporal constraints between the tasks in a process are simple temporal constraints. It is known that consistency of simple temporal constraints can be validated in polynomial time even without instantiation of the temporal variables [8]. Hence, we do not need to instantiate the temporal variables provided that we have a procedure for checking consistency of simple temporal constraints.

Based of above observations, we decided to implement the verification procedure as an ad-hoc algorithm exploiting constraint satisfaction techniques rather than using an existing constraint solver. The reason is that current constraint solvers do not provide the complete consistency check for simple temporal constraints. In our implementation we used algorithm IFPC [12] for maintaining temporal consistency. The reason for choosing this algorithm is that we are frequently adding new constraints to the network during problem solving (the temporal constraint is added when the condition in the implication – see the constraint model above – becomes true, that is, when some tasks are added to the process). This algorithm is part of procedure *Solve* that basically solves the problem $R \cup \{V_T = 1\}$ by exploring instantiations of validity variables in the depth-first search manner while maintaining consistency of constraints [7]. The variables are taken in the order given in the input (fixed order) and value 1 is tried first (the task is included in the process).

Based on observations 1 and 2 the procedure *Solve* is wrapped in the following verification algorithm that either returns the first task that is found infeasible (no feasible process exists for this task) or confirms that the workflow is sound:

```
verify(Workflow)
  ToCheck ← PrimitiveTasks(Workflow)
  R ← CSP(Workflow) ∪ {$V_{root}$ = 1}
  while ToCheck is not empty do
      select i from ToCheck
      Sol ← Solve(R ∪ {$V_i$ = 1})
      if Sol=fail then return fail(i)
      else ToCheck ← ToCheck \ {j | ($V_j$ = 1) in Sol}
  end while
  return true
```

3.1 Task Collapsing

So far we discussed a general verification method for nested workflows that checks every task in the workflow. Recall that if there are no extra constraints then the nested workflow is always sound. So, if there is a task such that there is no custom constraint that uses any of the subtasks of this task then a workflow consisting of this task (and its subtasks) is always semantically correct if it is seen separately from the rest of the workflow. This raises the question whether such tasks can be removed from the general verification procedure and for example treated as specific primitive tasks. This can be done in case of using logical (causal) constraints that may influence only the appearance of the task T in the process but that are not affecting which subtasks of this task are included. The situation is more complicated when temporal constraints (precedence and synchronization) are used because they may influence the duration of

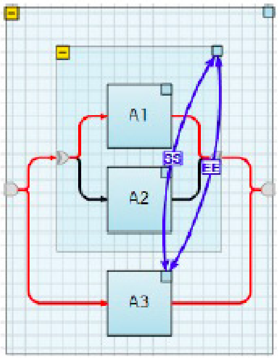


Fig. 2. Example of restricting the duration of task via extra synchronization start-start and end-end constraints

task T that further influences which of its sub-tasks may be selected. Figure 2 demonstrates this situation where the top task is forced to have the same duration as activity A3 that obviously restricts which (if any) activities A1 or A2 can be used (only the activity with the duration equal to the duration of A3 can be used).

Based on the above observations, we suggest modifying the verification method in such a way that some tasks are collapsed and treated as specific tasks during the verification process. In particular, the task T can be collapsed in the workflow if:

C1. none of the sub-tasks of T participates in any logical constraint, and

C2. the duration of task T is not influenced by any synchronization or precedence constraint.

The second condition above is not specified operationally. We use a conservative approach and we include the following tasks T in the verification process (all other tasks are collapsed):

V1. T is an ancestor task of tasks A or B (in the tree structure), where A and B are connected with a logical constraint,

V2. T is a parent task of some precedence or synchronization constraint C or T is an ancestor task of the parent of C or T is a descendant task of the parent of C (the parent of constraint C over tasks A and B is defined as the closest joint ancestor of tasks A and B).

Figure 3 demonstrates which tasks will be verified according to the conditions V1 and V2. Obviously, the conditions V1 and C1 are complementary, so any task that is selected for verification due to the condition V1 violates the condition C1 and no task that satisfies the condition C1 is selected for verification due to the condition V1. Conditions V2 and C2 are not necessarily complementary, but if a task is not selected for verification due to the condition V2 then this task satisfies the condition C2. According to V2, the task is not selected for verification if there is no extra temporal constraint among its subtasks (descendants) and there is no extra temporal constraint connected to any of its ancestors. Then such task behaves like being included in a workflow with no extra temporal constraints and hence it can be collapsed. Figure 3 gives an example of collapsed tasks omitted from the verification process.

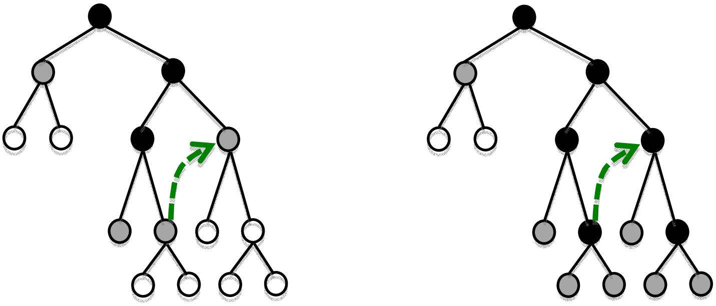


Fig. 3. Tasks to be verified (in black) due to the presence of an extra logical constraint (left) or an extra temporal constraint (right). Shadow nodes correspond to collapsed compound tasks or primitive tasks to be verified. The extra constraint is displayed as a dashed arrow.

4 Experimental Results

We are not aware about another verification method for workflows with extra constraints to compare with so to show scalability of the proposed techniques we experimentally compared the methods with and without task collapsing. The problem complexity surely depends on the structure of the workflow and on the location of extra constraints but it is hard to describe such features quantitatively. Moreover, there does not exist any benchmark set for verifying nested workflows. Hence we propose a method to generate random workflows with a given number of tasks and extra constraints. We start with a required number N of tasks and a root task. With the uniform distribution we select a random integer number K from the interval ⟨1, min(10,N-1)⟩ and we decompose the task to K subtasks. The type of decomposition (serial, parallel, alternative) is selected randomly with the uniform distribution. We generate K random positive integers such that their sum is N-K-1. These numbers define how many tasks will be generated for each subtask. The above process is repeated for each subtask until the number of required tasks equals one. In such a case the primitive task is generated with random integer duration selected from the interval ⟨1, 10⟩. To the obtained workflow we add a given number of extra constraints. Again, for each constraint a random pair of tasks is selected from the workflow and the type of constraint is randomly chosen with uniform distribution. We generated 30 problems for each size of the workflow and the number of extra constraints. The verification procedure finishes when the first flaw is found or the workflow is proved to be flawless. The experiments run on 3.3 GHz Intel Core i5 (using a single core) with 16 GB RAM.

Figures 4 and 5 show the aggregated results of experiments. In particular, the graphs show dependence of average runtime (in milliseconds) on the number of tasks in the workflow for different quantities of extra constraints. We measure the number of extra constraints as percent from the number of tasks in the workflow, so 10% for the workflow with 200 tasks means 20 extra constraints. The graphs show some interesting and somehow surprising trends. When we look at the verification of workflows without extra constraints (dashed lines in the figures) there is a significant difference between the algorithm with and without task collapsing. When task collapsing is included, the runtime is significantly smaller in comparison to workflows with extra constraints while this trend is completely opposite for the algorithm without task collapsing. This is because in the experiment we were looking for a single flaw so it may actually take more time when verifying the workflows without flaws (which is the case of workflows without extra constraints). Task collapsing makes this verification much easier as we actually collapse all the tasks into the root task. Surprisingly there does not seem to be a significant difference in runtime for different numbers of extra constraints. In general, it seems that the more extra constraints we add the smaller runtime is obtained. The reason could be as above – more constraints imply higher probability of flaw, which can be detected faster. The experiment did not confirm significant speed-up of the method when task collapsing is used, thought from the graphs it seems that this method is more stable regarding the runtime. This could be because even a single extra constraint connecting tasks from the opposite sides of the hierarchical structure may disallow task collapsing. A promising observation is that the runtime does not increase significantly with the size of the workflow and we can verify quickly even quite large workflows with hundreds of tasks (this is more than the size of typical real-life production workflow).

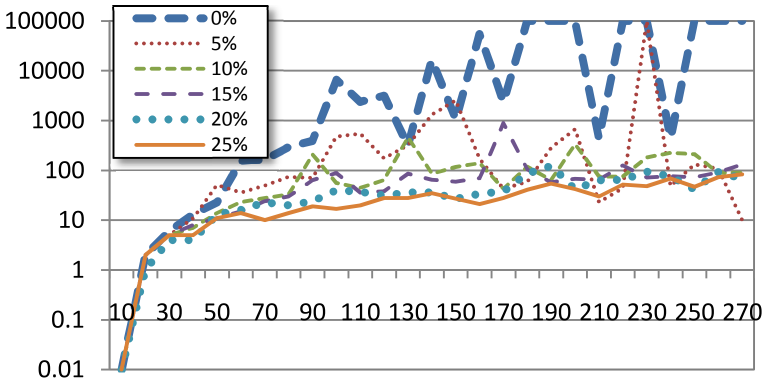


Fig. 4. Runtime (in milliseconds) for verifying workflows of given size using the algorithm **without task collapsing** for various quantities of extra constraints measured in percent of the tasks in the workflow (logarithmic scale)

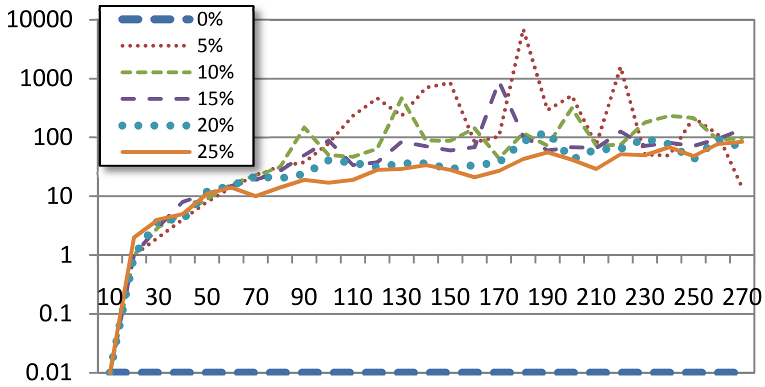


Fig. 5. Runtime (in milliseconds) for verifying workflows of given size using the algorithm **with task collapsing** for various quantities of extra constraints measured in percent of the tasks in the workflow (logarithmic scale)

In the previous text we mentioned several hypothesis about the behavior of the algorithm. The first hypothesis was that it is easier to detect a flaw than to prove that there are no flaws in the workflow. The second hypothesis was that the location of extra constraints influences efficiency of tasks collapsing. Hence we performed a second experiment where we separately verified workflows with and without flaws and we placed the extra constraints only locally. In particular, we placed the extra constraint in such a way that it influenced at most 20% of the whole workflow. This has been done in the following way. Let *size*(T) be the number of tasks in the workflow defined by task T. Formally, if T is a primitive task then *size*(T) = 1. If T is a compound task then *size*(T) = 1 + $\Sigma_{\mathrm{K}\in \mathrm{subtasks(T)}}$ *size*(K). When defining the extra constraints for the workflow with N tasks, we randomly select task T_1 such that *size*(T_1) ≤ 0.2N. Let P be the farthest ancestor task of task T_1 such that *size*(P) ≤ 0.2N. Then we select the second task T_2 for the extra constraint in the tree defined by task P.

Figure 6 shows the results. We used flawless workflows with up to 350 tasks and we varied the number of extra constrains from 0% to 25%. For the flawed workflows we used problems with up to 600 tasks and we varied the number of extra constrains from 5% to 25%. Each cross in the graph indicates the runtimes for both methods (with and without task collapsing) when verifying identical workflows. The scale of graphs clearly demonstrates that the runtimes for flawless workflows are larger than the runtimes for flawed workflows despite that fact that larger flawed workflows were used. The graphs also show that task collapsing pays off especially for flawless workflows and for larger flawed workflows. This confirms our hypotheses that it is easier to detect a flaw than to prove that the workflow is flawless and that task collapsing is more useful when the extra constraints connect tasks close to each other.

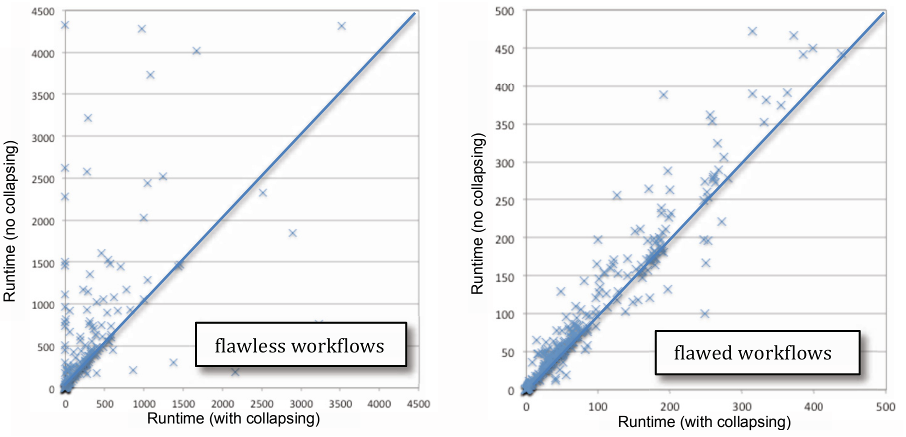


Fig. 6. Direct comparison of runtimes of verification algorithms with (X axis) and without (Y axis) task collapsing. The points above the diagonal indicate faster runs of the algorithm with task collapsing.

5 Conclusions

We presented novel constraint-based verification techniques that discover structural flaws in nested workflows with extra constraints and we experimentally showed that these techniques are fast enough to verify hundreds of tasks in a matter of seconds. This speed is enough for real-life workflows that typically consists of tens of tasks.

The proposed verification techniques were integrated to the manufacturing optimization software FlowOpt, where the user can verify the workflow any time by pressing a button in the user interface. The found flaws are reported and highlighted in the workflow editor so the user can immediately correct them. Hence the full power of workflow verification is put to the fingertips of regular users without the hassle of algorithmic complexity of the verification problem. Currently we can report the flawed task and the future research can focus on identifying the real reason of the flaw, for example the set of extra constraints causing the flaw.

Acknowledgements. The research is supported by the Czech Science Foundation under the contract P202/10/1188.

References

1. Bae, J., Bae, H., Kang, S.-H., Kim, Y.: Automatic Control of Workflow Processes Using ECA Rules. IEEE Transactions on Knowledge and Data Engineering 16(8), 1010–1023 (2004)
2. Barták, R., Čepek, O.: Nested Precedence Networks with Alternatives: Recognition, Tractability, and Models. In: Dochev, D., Pistore, M., Traverso, P. (eds.) AIMSA 2008. LNCS (LNAI), vol. 5253, pp. 235–246. Springer, Heidelberg (2008)
3. Barták, R., Cully, M., Jaška, M., Novák, L., Rovenský, V., Sheahan, C., Skalický, T., Thanh-Tung, D.: Workflow Optimization with FlowOpt, On Modelling, Optimizing, Visualizing, and Analysing Production Workflows. In: Proceedings of Conference on Technologies and Applications of Artificial Intelligence (TAAI 2011), pp. 167–172. IEEE Conference Publishing Services (2011)
4. Barták, R.: On Complexity of Verifying Nested Workflows with Extra Constraints. In: Proceedings of 4th International Conference on Agents and Artificial Intelligence (ICAART 2012), vol. 1, pp. 346–354. SciTePress (2012)
5. Beck, J.C., Fox, M.S.: Constraint-directed techniques for scheduling alternative activities. Artificial Intelligence 121, 211–250 (2000)
6. Bi, H.H., Zha, J.L.: Applying Propositional Logic to Workflow Verification. Information Technology and Management 5(3-4), 293–318 (2004)
7. Dechter, R.: Constraint Processing. Morgan Kaufmann (2003)
8. Dechter, R., Meiri, I., Pearl, J.: Temporal Constraint Networks. Artificial Intelligence 49, 61–95 (1991)
9. Giro, S.: Workflow Verification: A New Tower of Babel. In: AIS-CMS International Modeling and Simulation Multiconference, Buenos Aires, Argentina (2007)
10. Kim, P., Williams, B., Abramson, M.: Executing Reactive, Model-based Programs through Graph-based Temporal Planning. In: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), pp. 487–493 (2001)
11. Kuster, J., Jannach, D., Friedrich, G.: Handling Alternative Activities in Resource-Constrained Project Scheduling Problems. In: Proceedings of Twentieth International Joint Conference on Artificial Intelligence (IJCAI), pp. 1960–1965 (2007)
12. Planken, L.R.: New Algorithms for the Simple Temporal Problem. Master Thesis, Delft University of Technology (2008)
13. Sadiq, W., Orlowska, M.E.: Analyzing Process Models using Graph Reduction Techniques. Information Systems 25(2), 117–134 (2000)
14. van der Aalst, W., ter Hofstede, A.H.M.: Verification of Workflow Task Structures: A Petri-Net-Based Approach. Information Systems 25(1), 43–69 (2000)
15. van der Aalst, W., ter Hofstede, A.H.M.: Yawl: Yet another workflow language. Information Systems 30(4), 245–275 (2005)

Shortest Stochastic Path with Risk Sensitive Evaluation

Renato Minami and Valdinei Freire da Silva*

Escola de Artes, Ciências e Humanidades, Universidade de São Paulo (EACH-USP)
Av.Arlindo Bettio, 1000 - 03828-000, São Paulo, Brazil
renatomnm@gmail.com, valdinei.freire@usp.br

Abstract. In an environment of uncertainty where decisions must be taken, how to make a decision considering the risk? The shortest stochastic path (SSP) problem models the problem of reaching a goal with the least cost. However under uncertainty, a best decision may: minimize expected cost, minimize variance, minimize worst case, maximize best case, etc. Markov Decision Processes (MDPs) defines optimal decision in the shortest stochastic path problem as the decision that minimizes expected cost, therefore MDPs does not care about the risk. An extension of MDP which has few works in Artificial Intelligence literature is Risk Sensitive MDP. RSMDPs considers the risk and integrates expected cost, variance, worst case and best case in a simple way. We show theoretically the differences and similarities between MDPs and RSMDPs for modeling the SSP problem, in special the relationship between the discount factor γ and risk prone attitudes under the SSP with constant cost. We also exemplify each model in a simple artificial scenario.

Keywords: Markov Decision Process, Expected Utility Theory, Risk Sensitive.

1 Introduction

Achieving a goal in an environment with uncertainty is a common task in the area of planning [1,2]. In some cases uncertainty is modeled through probability distribution and the plan found by an agent must take into account the risk involved. In this work we consider the sequential decision process in stochastic environments where sequential decisions must be taken to reach a goal aiming at the lowest stochastic cost, the stochastic shortest path (SSP) problem.

Bertsekas and Tsitsiklis [3] analyze the SSP problem by using the theory of Markovian decision processes (MDPs) [4]. MDPs model the SSP problem by defining states, actions and rewards. A sequence of states defines a path, actions chosen at each state (policies) guide the probability distribution of paths and rewards specify the cost of a path. Path probability distribution are constrained

[*] Valdinei Silva would like to thank FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo – proc. 11/19280-8) for their financial supports.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 371–382, 2013.

by a transition function, and cost-path probability distributions are also constrained by a reward function. Path is simply evaluated by the sum of rewards.

Although under determinism, evaluating a path by the sum of rewards is clearly the best option, under risk we may choose different alternatives. A normative theory to evaluate decision under risk is the Expected Utility Theory (EUT) [5]. Under such theory a decision maker can state three distinct attitude towards risk: averse, neutral or prone. Although its validation as a descriptive theory has been under debate for several decades [6], as a normative theory, which is clearly the case for autonomous decision it has shown fruitful[1].

MDP literature considers the expected accumulated reward to evaluate policies, which presents a neutral attitude regarding risk. However, most decision makers show a risk-averse attitude, that is the reason why investments in stock market are spread among several assets, i.e., avoiding risk. An MDP version which presents risk attitude has been called Risk Sensitive MDPs (RSMDPs) and basic algorithms were obtained in the 70's [7,8]. RSMDPs incorporate risk attitude by considering the expected exponential of accumulated reward, which makes it possible to evaluate policies with risk-prone or risk-averse attitudes.

Although RSMDP presents algorithms and convergence condition similar to MDP, few applications consider RSMDPs as decision model. Recent advances have been made regarding models that are aware of the risk attitude but not considering RSMDPs. Liu and Koenig [9] consider probabilistic planning with nonlinear utility functions, from which RSMDPs is a special model. Delage and Mannor [10] consider a relaxed version of minimax by defining probabilistic constraints on minimal performance. Finally, Mannor and Tsitsiklis [11] consider a trade-off between expected and variance of accumulated rewards. In all of these approaches, optimal policies are non-stationary in general, whereas in RSMDPs optimal policies are stationary.

By using two simple scenarios, we present the differences and similarities between the two models: MDP and RSMDP. Such presentation is done under the light of EUT, basic algorithms and empirical examples. In section 2 we present the Markov Reward Process, a general framework for both MDP and RSMDP. In section 3 we present MDP and RSMDP formulations and basic algorithms. Finally section 4 and section 5 presents our experiments and conclusion respectively.

2 Markov Reward Process

In this section we present a general framework for sequential decision problems: Markov Reward Process. An MRP is defined by a tuple $\langle \mathcal{S}, \{\mathcal{A}_s\}, T(\cdot), r(\cdot) \rangle$ as follows:

- $\mathcal{S}$ is the set of all states of the process,
- $\mathcal{A}_s$ is the set of all possible actions to be executed at each state $s \in \mathcal{S}$,

[1] The aim of a descriptive theory is to predict how an agent (for instance a human) takes decision, whereas the aim of a normative theory is to prescribe how an agent should take decisions.

- $\mathcal{A} = \bigcup_{s \in \mathcal{S}} \mathcal{A}_s$ is the set of all possible actions,
- $T : \Gamma \times \mathcal{S} \rightarrow [0, 1]$ is the transition function, where $\Gamma = \{(s, a) | s \in \mathcal{S} \wedge a \in \mathcal{A}_s\}$, and
- $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ is the reward function.

The dynamic of an MRP is as follows at any time t: (i) the process is at state $s_t \in \mathcal{S}$, (ii) the action $a_t \in \mathcal{A}_{s_t}$ is executed, (iii) the process generates reward $r_t = r(s_t, a_t)$, and (iv) the process transits to some state $s' \in \mathcal{S}$ with probability $P(s_{t+1} = s' | s_t = s, a_t = a) = T(s, a, s')$. The process is Markovian because transitions and rewards depend only on the current state and the current action, i.e., $P(s_{t+1} = s' | s_t = s, a_t = a) = P(s_{t+1} = s' | s_t = s, a_t = a, s_{t-1}, a_{t-1}, \ldots, s_0, a_0)$ [12]. The process is *reward* because any history can be evaluated only taking into account rewards, i.e, the history $h = s_0, a_0, r_0, s_1, a_1, r_1, s_2, a_2, r_2, \ldots$ with regard to evaluation can be reduced to $x = r_0, r_1, r_2, \ldots$.

In an MRP, decisions consist in choosing a strategy to map states at any time to actions. A common set of restricted decision are stationary deterministic policies, from now on called just policy. A policy π maps each state $s \in \mathcal{S}$ into an action $a \in \mathcal{A}_s$, i.e., $\pi : \mathcal{S} \rightarrow \mathcal{A}$.

A special case of MRPs is the shortest stochastic path (SSP) problem where absorbing states are considered. A state s_G is an absorbing state if $r(s_G, a) = 0\ \forall a \in \mathcal{A}$ and $T(s_G, a, s_G) = 1\ \forall a \in \mathcal{A}_{s_G}$.

2.1 A Simple Example

Consider that a robot must reach a given destination in the shortest time and it must choose between two paths. The first path is 40 meters long through a sandy soil in which the robot get stuck from time to time. The second path is a very steep climb of 30 meters long, in which robots skids until the beginning when the robot gets stuck.

We model such problem with a simple MRP. The MRP consists of eight states: an initial state, four states of the first path and three state of the second path (see figure 1). States $s1$ has two actions available (go up or go down), where any other states has only one action available (go ahead). We consider that an action succeeds with probability p and fail with probability $1 - p$. If an action fails in the first path (states $s2$, $s3$, $s4$ and $s5$) the robot stays in the same state, whereas if an action fails in the second path (states $s6$, $s7$ and $s8$) the robot returns to the initial state $s1$. The goal is to reach state $s9$, when the process ends, i.e., $s9$ is an absorbing state. The robot receives a negative reward of -1 (punishment or cost) at any state but the goal.

In this example scenario, there are only two possible policies: go up or go down. We choose $p = 0.8689$ so that both policies present equal expected number of steps to reach the goal. Despite presenting the same expected accumulated reward, both policies present completely different distributions over accumulated reward. Figure 2 displays the histogram for both policies within a simulation over 100,000 executions.

We can read both histograms over various aspects. First, both policies present the same expected accumulated reward. Second, policy π_1 presents a smaller

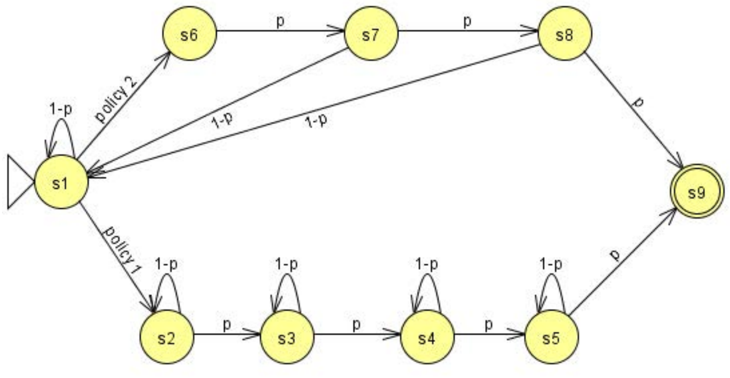


Fig. 1. The MRP transitions for the simple robot scenario

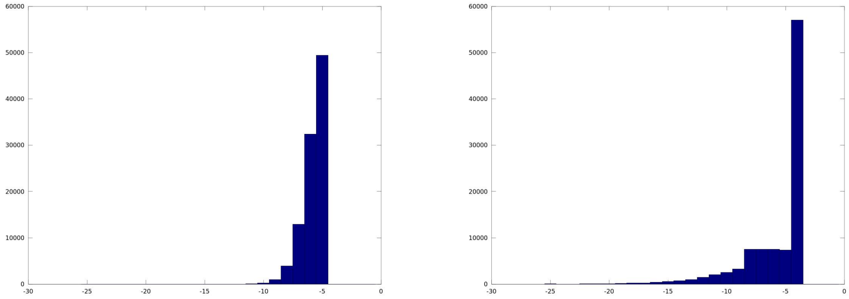


Fig. 2. Comparison between policies π_1 (left histogram) and π_2. (right histogram). Abscissa stands for accumulated reward and ordinate stands for number of ocurrences in 100,000 executions.

variance than policy π_2. Third, policy π_1 presents a faster decay regarding the probability of occurrences of larger number of steps over the 100,000 executions, whereas policy π_2 presents the smallest number of steps over all of the 100,000 executions. Then, which criteria should be used to choose between policies?

2.2 Expected Utility Theory

The Expected Utility Theory (EUT) is a normative theory to describe rational decision [5]. In EUT decisions are taken over lotteries, which are defined over probability distributions and a set of outcomes $\mathcal{X}$. Then, optimal decision results from a utility function definition. A utility function assigns a scalar value to a given outcome $x \in \mathcal{X}$, i.e., $u : \mathcal{X} \to \mathbb{R}$. A lottery defines the probability of each outcome occurring, i.e., $\ell : \mathcal{X} \to [0, 1]$. Then, given the utility function u, the value of lottery ℓ is given by the expected utility, i.e.,

$$V(\ell) = \sum_{x \in \mathcal{X}} \ell(x) u(x).$$

In an MRP we choose the outcome to be the accumulate rewards, i.e., $\mathcal{X} \subset \mathbb{R}$ and $x = \lim_{M \to \infty} \sum_{t=0}^{M} r_t$ and each policy generates a lottery. In this case the utility function is also increasing monotonically, i.e., the absolute utility value preserves the order among outcome values. An interesting qualitative characteristic is the attitude towards risk. A utility function is risk-averse if it is concave down, i.e., $\frac{d^2u}{dx^2} < 0$; and a utility function is risk-prone if it is concave up, i.e., $\frac{d^2u}{dx^2} > 0$.

Getting back to our simple example in figure 1, we have three cases: if a decision maker is indifferent to both policy, she is risk neutral; if a decision maker prefers policy π_1, she is risk averse; and if a decision maker prefers policy π_2, she is risk prone. These three attitudes can be modeled respectively in the following utility functions:

$$u(x) = x,\, u(x) = -e^{-\lambda^{+} x}, \text{ and } u(x) = e^{-\lambda^{-} x} \tag{1}$$

where $\lambda^{+} > 0$ and $\lambda^{-} < 0$. In the next section we show how these utility functions can be integrated within MRPs to prescribe optimal decisions.

3 Evaluating Markov Reward Process

3.1 Markov Decision Process

As viewed in previous session, an MRP models the sequential decision problem, but does not define how to evaluate policies. The Markov Decision Process (MDP) framework is an MRP plus a policy evaluation. The utility of the outcome is simply the accumulated rewards. A policy π is evaluated according to its expected utility, i.e,

$$V^{\pi}(s_0) = \lim_{M \to \infty} \mathrm{E}\left[\sum_{t=0}^{M} r(s_t, \pi(s_t))\right].$$

The agent of the process tries to find out the policy which maximizes the expected reward [4].

Given a policy, if its value does exist (we discuss existence below), the value of the policy for each state (s_0 inclusive) is defined by:

$$V^{\pi}(s) = r(s, \pi(s)) + \sum_{s' \in \mathcal{S}} T(s, \pi(s), s') V^{\pi}(s') \forall s \in \mathcal{S}.$$

A policy π^* is optimal if $V^{\pi^*}(s) \leq V^{\pi}(s)$ for any $s \in \mathcal{S}, \pi \in \mathit{\Pi}$, where $\mathit{\Pi}$ is the set of stationary deterministic policies.

To maximize expected utility, the optimal policy π^* must be found. An algorithm known in the literature is the Policy Iteration algorithm [4]. This algorithm iterates over policies, improving the current policy at each iteration. It does as follows:

1. Choose initial policy π_0 arbitrarily
2. At any iteration $i \geq 0$, while $(i = 0 \vee \pi_i \neq \pi_{i-1})$ do
   (a) Policy evaluation: obtain the value within the current policy π_i for every $s \in \mathcal{S}$ solving the system of equations below

$$V^{\pi_i}(s) = r(s, \pi_i(s)) + \sum_{s' \in \mathcal{S}} T(s, \pi_i(s), s') V^{\pi_i}(s') \forall s \in \mathcal{S}$$

   (b) Policy Improvement: improve the current policy by doing the update below for every $s \in \mathcal{S}$

$$\pi_{i+1}(s) = \arg\max_{a \in \mathcal{A}_s} \left[r(s, a) + \sum_{s' \in \mathcal{S}} T(s, a, s') V^{\pi_i}(s') \right]$$

Let $\mathbf{T}(\pi)$ be a $|\mathcal{S}| \times |\mathcal{S}|$ matrix where elements $[\mathbf{T}]_{i,j}(\pi) = T(i, \pi(i), j)$ and G be the index of the unique absorbing state. The policy π is proper if $\lim_{t\to\infty}[\mathbf{T}(\pi)^t]_{i,G} = 1 \forall i \in \{1, 2, \ldots, |\mathcal{S}|\}$, i.e., the absorbing state s_G is reached with probability 1 [3].

Under the SSP problem, two sufficient conditions to policy iteration convergence are: (i) $r(s, a) \leq 0$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$, and (ii) the initial policy π_0 is a proper policy. If such conditions are not observed, a common approach to guarantee convergence is to consider a discount factor $\gamma < 1$ [4]. Then, the following value function is considered:

$$V^{\pi}(s_0) = \lim_{M\to\infty} \mathrm{E}\left[\sum_{t=0}^{M} \gamma^t r(s_t, \pi(s_t))\right],$$

which is guaranteed to be finite independently of the policy π.

3.2 Shortest Stochastic Path under Constant Cost

A common semantic attributed to the discount factor γ is that the decision maker considers rewards received in the future as less appealing than rewards received in the present. Another common semantic is considering that γ dictates the chance of being alive for one more time step. If we consider the SSP problem under constant cost, a new semantic can be attributed to the discount factor γ.

Let the reward function be $r(s, a) = -1$ for all $s \in \mathcal{S}$, $a \in \mathcal{A}$ just like in our previous simple example, i.e., the decision maker wants to minimize the number of steps. Then, the decision maker minimizes the utility function

$$u(x) = -\frac{1 - \gamma^{-x}}{1 - \gamma} = K_1\gamma^{-x} + K_2 = K_1 e^{-\ln(\gamma)x} + K_2.$$

Since $\ln(\gamma) < 0$ and $K_1 > 0$ the decision maker presents a risk prone attitude (remember equation 1). The previous analysis can be generalized to $\gamma > 1$. In this case $\ln(\gamma) > 0$ and $K_1 < 0$, therefore the decision maker presents a risk averse attitude.

Although the common semantics attributed to the discount factor γ make sense in a system acting for a long time, when the decision maker can take advantage of the result of such acts before the process ends, the risk attitude semantic makes sense when the decision maker only takes advantage of the result of such acts when the process ends. In the last case, using $\gamma < 1$ may be undesirable since human tends to be risk averse. However, restricting policy iteration to proper policy is not anymore enough if $\gamma > 1$. In the next section we discuss the more general case.

3.3 Risk Sensitive Markov Decision Process

If the risk attitude is important for the decision maker, MDPs cannot represent such risk attitude arbitrarily. If rewards are $r(s, a) = -1$ and discount is used it can represent risk prone attitude. The extension Risk Sensitive MDP (RSMDP) allows to choose the risk attitude arbitrarily under finite rewards, i.e., $-\infty < r(s, a) < \infty$ for all $s \in \mathcal{S}, a \in \mathcal{A}$.

Like an MDP, an RSMDP is an MRP plus a policy evaluation. The RSMDP has an additional factor λ that models the risk attitude. If $\lambda < 0$, we have risk prone attitude, if $\lambda > 0$, we have risk averse attitude, and in the limit $\gamma \to 0$, we have risk neutral attitude [13].

A policy π is evaluated according to the expected exponential of accumulated rewards by

$$\begin{aligned} V^\pi(s_0) &= \lim_{M\to\infty} \mathrm{E}\left[-\mathrm{sign}(\lambda) \textstyle\prod_{t=0}^{M} \exp(-\lambda r_t)\right] \\ &= \lim_{M\to\infty} \mathrm{E}\left[-\mathrm{sign}(\lambda) \exp\left(-\lambda \textstyle\sum_{t=0}^{M} r_t\right)\right], \end{aligned}$$

where $\exp(x)$ stands for e^x. Given a policy π and the unique absorbing state s_G, the value of π can be calculated as follows: (i) $V^\pi(s_G) = -\mathrm{sign}(\lambda)\exp(0) = -\mathrm{sign}(\lambda)$; and (ii) for all $s \neq s_G \in \mathcal{S}$, the value of a policy π can be calculated by:

$$V^\pi(s) = \exp(-\lambda r(s, \pi(s))) \sum_{s' \in \mathcal{S}} T(s, \pi(s), s') V^\pi(s').$$

Just like an MDP, there is also a Policy Iteration algorithm for RSMDPs [14]. It does as follows:

1. Choose initial policy π_0 arbitrarily
2. At any iteration $i \geq 0$, while $(i = 0 \vee \pi_i \neq \pi_{i-1})$ do
   (a) Policy evaluation: obtain the value within the current policy π_i for every $s \in \mathcal{S}$ solving the system of equations below

$$\begin{gathered} V^{\pi_i}(s_G) = -\mathrm{sign}(\lambda) \\ V^{\pi_i}(s) = \exp(-\lambda r(s, \pi_i(s))) \sum_{s' \in \mathcal{S}} T(s, \pi_i(s), s') V^{\pi_i}(s') \ \forall s \neq s_G \in \mathcal{S} \end{gathered}$$

(b) Policy Improvement: improve the current policy by doing the update below for every $s \in \mathcal{S}$

$$\pi_{i+1}(s) = \arg\max_{a \in \mathcal{A}_s} \left[\exp(-\lambda r(s,a)) \sum_{s' \in \mathcal{S}} T(s,a,s') V^{\pi_i}(s') \right]$$

Two sufficient conditions to policy iteration convergence are: (i) $r(s,a) \leq 0$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$, and (ii) the initial policy π_0 is a λ-feasible policy. Let $\mathbf{r}(\pi)$ be a $|\mathcal{S}|$ vector where elements $[\mathbf{r}]_i(\pi) = r(i, \pi(i))$ and $\mathbf{D}(\pi) = \text{diag}[\exp(-\lambda \mathbf{r}(\pi))]\mathbf{T}(\pi)$, where diag$[\mathbf{r}]$ stands for the diagonal matrix defined by the vector $\mathbf{r}$. The policy π is λ-feasible if $\lim_{t \to \infty}[\mathbf{D}(\pi)^t] = \mathbf{0}$, i.e., the probability of not being in an absorbing state vanishes faster than the exponential accumulated reward [14]. Additionally, if $\lambda < 0$ (risk prone) and policy π is proper then policy π is λ-feasible. The same is not always true for $\lambda > 0$ (risk averse).

An alternative to λ-feasibility is adding a discount factor $\gamma < 1$. Chung and Sobel [15] discount rewards, which guarantees that values of policies does exist. However, by discounting rewards stationary policies are not any more optimal ones and non-stationary policies must be considered. Another option is discounting recursively future values, i.e., exponential of future accumulated reward, instead of the reward itself [8], in this case the optimal policy is stationary.

4 Results

In order to better expose the differences between MDPs and RSMDPs we first present another simple example and shows the results obtained with it.

4.1 Example Scenario

Consider a candidate wants to take his/her drivers license and must take classes to improve his/her driving experience. The most experience the candidate has the greater is the chance of being approved in the practical exam. The candidate must pay for classes, but he/she wants to spend the less money as possible. Additionally, to take the practical exam a tax must also be paid. How much classes the candidate must take before taking the practical exam?

We model such problem as follows:

1. the candidate can take the practical exam as often as necessary by paying \$2;
2. the candidate must choose how many classes to do before taking the practical exam and he/she pays \$1 for each class;
3. the candidate can take at the most 4 classes before each practical exam;
4. the candidate accumulates at a maximum of 10 hours of experience; and
5. the chance of candidate being approved in the practical exam depends on the previous accumulated experience x and current number of classes taken y and is given by $p(x, y) = 0.08x + 0.04y$.

To model this problem we define an MRP with 12 states and 5 actions. The states $\{0, 1, 2, \ldots, 10, s_G\}$ keep information of the number of hours accumulated before the current classes section. State s_G is an absorbing state. Actions $\{0, 1, \ldots, 4\}$ indicate how many classes to take before each practical exam. The reward function for any state $s \neq s_G$ is given by $r(s, a) = -2 - a$, i.e., \$2 for taking the practical exam and \$$a$ for taking the classes. The transition function for any state $s \neq s_G$ is given by:

$$T(s, a, s') = \begin{cases} 0.08s + .4a & \text{, if } s' = s_G \\ 1 - (0.08s + .4a) & \text{, if } s' = \max\{s + a, 10\} \\ 0 & \text{, otherwise} \end{cases} .$$

4.2 Experiments

This empirical experiment shows a practical application of RSMDPs. We analyzed the behavior of the RSMDP as we varied the parameter λ from negative (risk prone) to positive (risk averse). We make such analysis through the following characteristic: optimal policies, expected cost, deviation cost, and outcome distribution. The experiments was conducted as follows: (i) we varied λ in order to define all possible optimal policies; and (ii) for each policy we simulated the MRP for 100,000 times.

A summary of the results is shown in table 1. In the higher values of λ, the decision maker tends to take more classes before each practical exam in order to minimize the risk of not being approved. In contrast, in the lower values of λ the decision maker tends to take less classes and takes the risks involved in doing the exam with such a few driving experience. When $\lambda = 0$ (MDP), the decision maker takes a middle term of both: taking long hour of classes when has no experience accumulated and trusting in such experience there after.

Table 1. Optimal policies, means and standard deviations obtained for different values of λ

States	-0,7	-0,5	-0,4	-0,3	-0,2	-0,1	0	0,1	0,2	0,3	0,4	0,5	0,6
0	1	2	2	3	3	4	4	4	4	4	4	4	4
1	0	0	1	1	2	3	4	4	4	4	4	4	4
2	0	0	0	0	1	2	3	4	4	4	4	4	4
3	0	0	0	0	0	1	2	3	4	4	4	4	4
4	0	0	0	0	0	0	1	2	3	4	4	4	4
5	0	0	0	0	0	0	0	1	2	3	4	4	4
6	0	0	0	0	0	0	0	0	1	2	3	4	4
7	0	0	0	0	0	0	0	0	0	1	2	3	3
8	0	0	0	0	0	0	0	0	0	0	1	1	2
9	0	0	0	0	0	0	0	0	0	0	0	1	1
10	0	0	0	0	0	0	0	0	0	0	0	0	0
mean	-27.0	-15.5	-15.4	-12.4	-12.3	-11.3	-11.2	-11.4	-11.9	-12.4	-12.7	-12.8	-13.1
deviation	24.0	11.5	11.4	7.3	7.4	5.2	4.2	3.7	3.5	3.4	3.6	3.7	3.8

Table 1 also summarizes mean and standard deviation accumulated reward. As it was expected the best mean reward is obtained within an MDP, since MDP maximizes only this variable. In the RSMDP, the mean gets smaller as λ

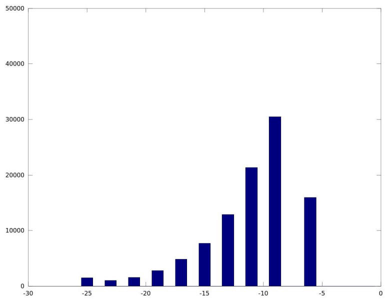


Fig. 3. Histogram for optimal policy with risk neutral attitude ($\lambda = 0$). Abscissa stands for accumulated reward and ordinate stands for number of ocurrences in 100,000 executions. The abscissa value -25 accumulates rewards occurrence smaller than -25.

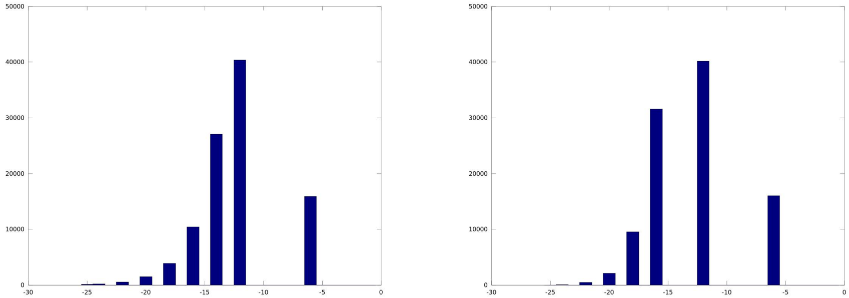


Fig. 4. Histogram for optimal policy with risk averse attitude when $\lambda = 0.3$ (left) and $\lambda = 0.6$ (right). Abscissa stands for accumulated reward and ordinate stands for number of ocurrences in 100,000 executions. The abscissa value -25 accumulates rewards occurrence smaller than -25.

get away from 0, in both case: risk prone and risk averse. Another interesting theoretical result of RSMPDs is that when $\lambda \approx 0$, risk averse makes a trade-off between maximizing the expected reward and minimizing (maximizing) the standard deviation under risk averse (prone) attitude [13]. This behavior can be seen when $\lambda \in [-0.2, 0.2]$. Finally, we note that $\lambda > 0.6$ does not produce λ-feasibility, so we does not have an optimal policy that always takes 4 classes before taking the practical exam.

Figures 3, 4 and 5 show histograms for some executions. Analyzing the histograms, we find that the histogram of the RSMDP with risk averse presents less percentage of values < -25. In the histogram of the RSMDP with risk prone there is a high percentage of values < -25, but it also present some percentage of high values (-3 or -5).

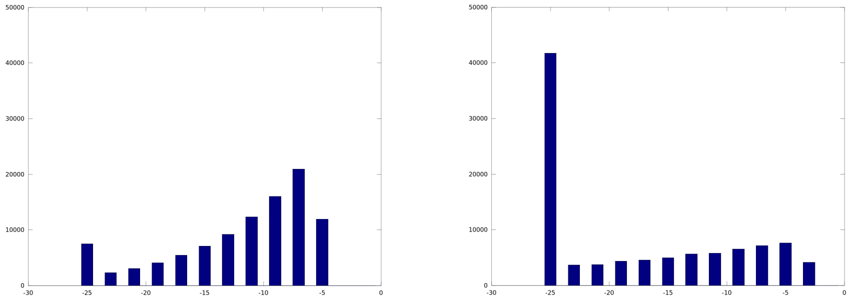


Fig. 5. Histogram for optimal policy with risk prone attitude when $\lambda = -0.3$ (left) and $\lambda = -0.7$ (right). Abscissa stands for accumulated reward and ordinate stands for number of ocurrences in 100,000 executions. The abscissa value -25 accumulates rewards occurrence smaller than -25.

5 Conclusion

The stochastic shortest path problem aims to find the lowest cost for a decision involving a stochastic environment. The use of Markov Decision Processes is common in the literature for modeling such problem [3]. However, MDPs does not contemplate various aspect for taking decision such as: variance, expectation, worst cases, best cases, etc. An alternative is taking MDPs as a second criteria, whereas another criteria is first guaranteed by another process [1].

Risk Sensitive MDPs present a more versatile way of modeling the SSP problem, allowing the arbitrarily choice of risk attitude. Although it has been defined in the 70's [7,8], RSMDPs have not been used within Artificial Intelligence community. Recently, the idea of risk sensitive approach has been presented under different formulation in [16]. RSMDPs have been more appealing to the Control Optimization community [17].

The experiments aimed to show how different decision under MDPs and RSMDPs can be for the SSP problem. Despite the versatile obtained within RSMDPs, basic algorithms are similar to MDPs algorithm which allows it practical implementation in most application where MDPs is used. This is very helpful when compared to other arbitrary alternatives to explicit model utility function by not considering Markov property [18,19].

References

1. do Lago Pereira, S., de Barros, L.N., Cozman, F.G.: Strong Probabilistic Planning. In: Gelbukh, A., Morales, E.F. (eds.) MICAI 2008. LNCS (LNAI), vol. 5317, pp. 636–652. Springer, Heidelberg (2008)
2. Trevizan, F.W., Cozman, F.G., de Barros, L.N.: Planning under risk and knightian uncertainty. In: Veloso, M.M. (ed.) IJCAI, pp. 2023–2028 (2007)

3. Bertsekas, D.P., Tsitsiklis, J.N.: An Analysis of Stochastic Shortest Path Problems. Mathematics of Operations Research 16(3) (1991)
4. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming, 1st edn. John Wiley & Sons, Inc., New York (1994)
5. Keeney, R.L., Raiffa, H.: Decisions with Multiple Objectives: Preferences and Value Tradeoffs. Wiley, New York (1976)
6. Braga, J., Starmer, C.: Preference anomalies, preference elicitation and the discovered preference hypothesis. Environmental & Resource Economics 32, 55–89 (2005)
7. Howard, R.A., Matheson, J.E.: Risk-sensitive markov decision processes. Management Science 18(7), 356–369 (1972)
8. Porteus, E.L.: On the optimality of structured policies in countable stage decision processes. Management Science 22(2), 148–157 (1975)
9. Liu, Y., Koenig, S.: Probabilistic planning with nonlinear utility functions. In: Long, D., Smith, S.F., Borrajo, D., McCluskey, L. (eds.) ICAPS, pp. 410–413. AAAI (2006)
10. Delage, E., Mannor, S.: Percentile optimization for markov decision processes with parameter uncertainty. Oper. Res. 58(1), 203–213 (2010)
11. Mannor, S., Tsitsiklis, J.N.: Mean-variance optimization in markov decision processes. In: Getoor, L., Scheffer, T. (eds.) ICML, pp. 177–184. Omnipress (2011)
12. Kaelbling, L.P., Littman, M.L., Cassandra, A.R.: Planning and acting in partially observable stochastic domains. Artificial Intelligence 101, 99–134 (1998)
13. Sladký, K.: Growth rates and average optimality in risk-sensitive markov decision chains. Kybernetika 44(2), 205–226 (2008)
14. Patek, S.D.: On terminating markov decision processes with a risk-averse objective function. Automatica 37(9), 1379–1386 (2001)
15. Chung, K.-J., Sobel, M.J.: Discounted mdp's: distribution functions and exponential utility maximization. SIAM J. Control Optim. 25, 49–62 (1987)
16. Ermon, S., Conrad, J., Gomes, C.P., Selman, B.: Risk-sensitive policies for sustainable renewable resource allocation. In: Walsh, T. (ed.) IJCAI, pp. 1942–1948. IJCAI/AAAI (2011)
17. Cavazos-Cadena, R., Salem-Silva, F.: The discounted method and equivalence of average criteria for risk-sensitive markov decision processes on borel spaces. Applied Mathematics & Optimization 61, 167–190 (2010)
18. Koenig, S., Liu, Y.: The interaction of representations and planning objectives for decision-theoretic planning tasks. J. Exp. Theor. Artif. Intell. 14(4), 303–326 (2002)
19. Thiebaux, S., Gretton, C., Slaney, J., Price, D., Kabanza, F.: Decision-theoretic planning with non-markovian rewards. Journal of Artificial Intelligence Research 25, 17–74 (2006)

Forward and Backward Feature Selection in Gradient-Based MDP Algorithms

Karina Olga Maizman Bogdan and Valdinei Freire da Silva*

Escola de Artes, Ciências e Humanidades, Universidade de São Paulo (EACH-USP)
Av.Arlindo Bettio, 1000 - 03828-000, São Paulo, Brazil
karina.bogdan@gmail.com, valdinei.freire@usp.br

Abstract. In problems modeled as Markov Decision Processes (MDP), knowledge transfer is related to the notion of generalization and state abstraction. Abstraction can be obtained through factored representation by describing states with a set of features. Thus, the definition of the best action to be taken in a state can be easily transferred to similar states, i.e., states with similar features. In this paper we compare forward and backward greedy feature selection to find an appropriate compact set of features for such abstraction, thus facilitating the transfer of knowledge to new problems. We also present heuristic versions of both approaches and compare all of the approaches within a discrete simulated navigation problem.

Keywords: Feature Selection, Knowledge Transfer, State Abstraction.

1 Introduction

Problems that involve sequential decisions can be modeled as a Markov Decision Process (MDP) [1]. An MDP agent must evaluate the resulting value of decisions in order to define a policy that maximizes the expected value of the sequence of decisions. However, the resolution of MDPs with a relatively large set of states is computationally expensive. The course of dimensionality is even worse, when the Reinforcement Learning problem is considered [2]. In reinforcement learning, a sequential decision problem must be solved by trial-and-error iterations within an unknown environment, i.e., an agent interacts with the environment to collect statistic significant experience.

An alternative to learning from scratch is to transfer the knowledge produced from source problems to new, similar ones. Knowledge transfer is related to the notion of generalization and state abstraction [3]. State abstraction is an approach in order to generalize knowledge among states that compose an MDP problem, by ignoring irrelevant information about them [4]. In this case, experience or information obtained about a state can be transferred to similar ones. This approach also allows the transfer of knowledge between different MDPs.

* Valdinei Silva would like to thank FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo – proc. 11/19280-8) for their financial supports.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 383–394, 2013.

The state abstraction can be performed using a relational representation of the problem, where by using features we can describe relations between objects of the environment [5, 6]. In order to validate the state abstraction, we need to generate knowledge from source problems and evaluate that knowledge obtained for transferring. For this purpose, the AbsProb-PI algorithm is used [7]. The AbsProb-PI algorithm makes use of a relational representation to abstract states in an MDP and transfer the knowledge generated to new MDPs. Relational representation has the advantage of representing state compactly such that abstract state can be matched among different MDPs even if the number of concrete states and the number of objects in the environment are different.

The quality of the policy found by the AbsProb-PI algorithm depends on the quality of relational representation. The relational description consists in defining a set of features, which describes abstractly the states in order to find an appropriate policy for transferring. However, such set has often many features, which may present the problem of overfitting when transferring, as also increases the computational complexity for learning. Thus, in order to find an appropriate abstraction for an MDP environment, it is necessary to define a compact set of features to describe the states by ignoring irrelevant information.

Several works in this area focus on choosing a compact set of features that approximates well the value function, which induces an approximated policy. In [8, 9] regularization is used in order to find a sparse representation of states, whereas in [9, 10] forward and backward greedy selection is used to compose incrementally a compact set of features. Some of this works have also applied such ideas within Reinforcement Learning problems [9–11]. Another approach consists in analyzing directly the MDP model [11]. In this paper we explore forward and backward greedy selection. While previous works choose features based on the value function, our approach chooses among features by evaluating them directly on policy space by considering gradient-based MDP algorithms.

The paper is organized as follows. Section 2 reviews basic concepts of MDP, state abstraction and the AbsProb-PI algorithm, which is used to validate the state abstraction. Section 3 presents our approaches to find an appropriate compact set of features to describe the states. Section 4 reports experiments with the approaches presented, and Section 5 summarizes our conclusions.

2 Background

2.1 Markov Decision Process

A Markov Decision Process models a sequential decision problem for an agent by defining a process involving the following concepts: states, actions and rewards. The process runs as follows: (*i*) the agent perceives state s_t at time t, (*ii*) the agent executes action a_t, (*iii*) the agent receives reward r_t and (*iv*) the process transits to next state s_{t+1}. Formalizing, an MDP is defined by the tuple $\langle \mathcal{S}, \mathcal{A}, T, r, b^0 \rangle$ [7], where $\mathcal{S}$ is a finite set of states where the agent can be at any time t; $\mathcal{A}$ is a finite set of actions that can be executed at any time

t; $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ is a transition function that defines the probability to go from state $s \in \mathcal{S}$ to state $s' \in \mathcal{S}$ after executing the action $a \in \mathcal{A}$, i.e., $P(s_{t+1} = s'|s_t = s, a_t = a) = T(s, a, s')$; $r : \mathcal{S} \rightarrow \mathbb{R}$ is the reward function associated to any state and indicates a partial evaluation of the process; and $b^0 : \mathcal{S} \rightarrow [0, 1]$ is the distribution for the initial state, i.e., $P(s_0 = s) = b^0(s)$.

The solution for an MDP is a stationary policy π that defines for each state a strategy to choose an action to be executed [1]. We consider stationary probabilistic policies $\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$. Then, after perceiving state s, the agent chooses action a with probability $\pi(s, a)$, i.e., $P(a_t = a|s_t = s) = \pi(s, a)$. A policy is evaluated by the sum of expected (discounted) reward, i.e., $V^\pi = \mathrm{E}_{s_0 \sim b^0} \left[\sum_{t=0}^{\infty} \gamma^t r_t | \pi\right]$, where V^π is the value of policy π and $0 \leq \gamma < 1$ is a discount factor, which guarantees value V^π is finite for any policy. Let Π be the set of all stationary probabilistic policy, policy π^* is optimal if and only if $V^{\pi^*} \geq V^\pi \forall \pi \in \Pi$. It is also common to define the value of a state s when following policy π by $V^\pi(s) = \mathrm{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t | \pi, s_0 = s\right]$.

2.2 Features and Abstract States

The transfer of knowledge between MDPs is related to the idea of state abstraction, and in this context relational representations for states and actions is usual [5]. A relational representation allows the use of features to evidence the relationship among objects of an environment [6, 7]. We consider these features to be propositions, i.e., we describe states of an MDP with propositions. Then, state abstraction is performed by defining similarity among states if they presents similar features, i.e., states with the same features represent the same abstract state.

Consider the set of states $\mathcal{S}$ and an ordered set of boolean functions $\mathcal{F} = f_1, f_2, \ldots, f_N$ where $f_i : \mathcal{S} \rightarrow \{0, 1\}$ indicates if a state presents features i. $\mathcal{S}$ and $\mathcal{F}$ induce a space of abstract states $\sigma \in \mathcal{S}_{ab}^{\mathcal{F}}$ and each abstract state induces a set of states $\mathcal{S}_\sigma$ such that: (i) for all $s \in \mathcal{S}$ there exists $\sigma \in \mathcal{S}_{ab}^{\mathcal{F}}$ such that $s \in \mathcal{S}_\sigma$ and if $s \in \mathcal{S}_{\sigma'}$ and $s \in \mathcal{S}_{\sigma''}$ then $\mathcal{S}_{\sigma'} = \mathcal{S}_{\sigma''}$, i.e., every state in $\mathcal{S}_\sigma$ is represented by one and only one abstract state; and (ii) if $s', s'' \in \mathcal{S}_\sigma$ then $f_i(s') = f_i(s'')$ for every feature $f_i \in \mathcal{F}$, i.e., states that are represented by the same abstract state present equal features. In the same way that an abstract state σ induces a set of states $\mathcal{S}_\sigma$, an (abstract) state also induces a set of features $\mathcal{F}_\sigma \subseteq \mathcal{F}$ where $f_i \in \mathcal{F}_\sigma$ if $f_i(s) = 1 \forall s \in \mathcal{S}_\sigma$. Also, each subset of $\mathcal{F}$ potentially induces an abstract state, which may not exist if there is not states that fulfill such set of features. With regard to the size of the space of abstract states $\mathcal{S}_{ab}^{\mathcal{F}}$ we have the following property $|\mathcal{S}_{ab}^{\mathcal{F}}| \leq \min\{|\mathcal{S}|, 2^{|\mathcal{F}|}\}$. Note that, if $\mathcal{F}$ is complete [12], the spaces of abstract states and original states are the same.

As an example, consider a set of features $\mathcal{F} = \{f_1, f_2, f_3\}$ and a set of states $\mathcal{S} = \{s_1, s_2, s_3\}$ where $\mathcal{F}_{s_1} = \{f_1, f_2\}$, $\mathcal{F}_{s_2} = \{f_3\}$, and $\mathcal{F}_{s_3} = \{f_1, f_2\}$. Then, there exist two abstract states: σ_1 and σ_2, such that $\mathcal{S}_{\sigma_1} = \{s_1, s_3\}$ and $\mathcal{S}_{\sigma_2} = \{s_2\}$, and $\mathcal{S}_{ab} = \{\sigma_1, \sigma_2\}$ which is a partition of the finite set of states $\mathcal{S}$.

2.3 Abstract Policies

To evaluate a given abstraction, it is necessary to solve the MDP problem by finding the best strategy for executing action in each abstract state. Similarly to the definition of an abstract state space, we can also define a set of abstract actions $\alpha \in \mathcal{A}_{ab}$, i.e., an abstract action α induces a set of actions $\mathcal{A}_\alpha$ [7]. Then, an abstract policy π_{ab} is defined within abstract state space and abstract action space, i.e., $\pi_{ab} : \mathcal{S}_{ab} \times \mathcal{A}_{ab} \to [0, 1]$. A local-optimal abstract policy can be found by using the algorithm AbsProb-PI which implements Policy Iteration algorithm based on gradient [7]. First, define the transition matrix $T^{\pi_{ab}}(s, s')$ and the value function $V^{\pi_{ab}}(s)$ by:

$$T^{\pi_{ab}}(s,s') = \sum_{\alpha \in A_{ab}} \pi_{ab}(\alpha|\sigma(s)) \sum_{a \in A_\alpha} \frac{1}{|A_\alpha|} T(s,a,s') \,,$$
$$\mathbf{V}^{\pi_{ab}} = (\mathbf{I} - \gamma \mathbf{T}^{\pi_{ab}})^{-1}\mathbf{r} \,,$$

where $\mathbf{r}$, $\mathbf{T}^{\pi_{ab}}$ and $\mathbf{V}^{\pi_{ab}}$ are vector representations of the reward function $r(s)$, the transition matrix $T^{\pi_{ab}}(s, s')$ and the value function $V^{\pi_{ab}}(s)$ respectively. Second, define the transition matrix $\mathbf{T}^{\alpha,\epsilon}$ that chooses an abstract action α with probability $1 - \epsilon$ and chooses uniformly among all other abstract actions with probability ϵ. The parameter $\epsilon > 0$ guarantees that the policy converges at most to a local ϵ-greedy policy. Third, define a step size function $\delta(i) = (1 + \frac{i}{k})^{-1}$, at iteration i where k is a constant. Let $\mathbf{b}^0$ be the vector representation of the initial distribution $b^0(s)$. The AbsProb-PI algorithm is defined by [7]:

1. Initiate the abstract probabilistic policy π_{ab} arbitrarily such that

$$\pi_{ab}(\sigma,\alpha) \geq \frac{\epsilon}{|A_{ab}|} \forall \sigma \in S_{ab}, \alpha \in A_{ab} \text{ and } \sum_{a \in A_{ab}} \pi_{ab}(\sigma,\alpha) = 1 \forall \sigma \in S_{ab}$$

2. At each iteration i:
   (a) Calculate the value function $\mathbf{V}^{\pi_{ab}}$
   (b) Calculate the product $\mathbf{C} = \gamma \mathbf{b}^{0\top}(\mathbf{I} - \gamma \mathbf{T}^{\pi_{ab}})^{-1}$
   (c) For each $\alpha \in \mathcal{A}_{ab}$ calculate $\mathbf{\Delta}^{\alpha,\pi_{ab}} = (\mathbf{T}^{\alpha,\epsilon} - \mathbf{T}^{\pi_{ab}})\mathbf{V}^{\pi_{ab}}$
   (d) For each $\sigma \in S_{ab}$ and $\alpha \in A_{ab}$ calculate $W(\sigma,\alpha) = \sum_{s \in S_\sigma} C(s) \Delta^{\alpha,\pi_{ab}}(s)$
   (e) For each $\sigma \in S_{ab}$ find the best direction $\alpha^*_\sigma = \arg\max_{\alpha \in A_{ab}} W(\sigma,\alpha)$
   (f) Choose a step size $\delta(i)$ and updates policy π_{ab} by:

$$\pi_{ab}(\sigma,\alpha) \leftarrow \begin{cases} (1-\delta(i))\pi_{ab}(\sigma,\alpha) + \delta(i)\left(\frac{\epsilon}{|\mathcal{A}_{ab}|} + (1-\epsilon)\right) & \text{, if } \alpha = \alpha^*_\sigma \\ (1-\delta(i))\pi_{ab}(\sigma,\alpha) + \delta(i)\frac{\epsilon}{|\mathcal{A}_{ab}|} & \text{, if } \alpha \neq \alpha^*_\sigma \end{cases} .$$

3 State Abstraction by Feature Selection

Remember that the larger the set of features, the lesser abstraction is obtained. Although abstraction can: accelerate learning, accelerate planning and make possible knowledge transfer; abstraction can also degrade the value of optimal abstract policies. Then, finding an appropriate set of features that makes a tradeoff between learning, planning, knowledge transfer and quality of policy is quite interesting. In this section we define some algorithms to looking for an appropriated set of features regarding quality of policy.

To find an appropriate compact set of features to describe the states of an MDP problem, we explore two approaches: forward and backward. We also present heuristics derived from these approaches as an alternative to find such set of features. For both approaches the algorithm AbsProb-PI has been used to solve the MDP problem so we can validate the state abstraction obtained from each iteration of the algorithms proposed. We consider the following interface for the AbsProb-PI algorithm $\hat{\pi}^*_{ab} \leftarrow \texttt{AbsProb-PI}(\mathcal{S}_{ab}, N_{\text{steps}})$, where N_{steps} is the number of iteration to be done before returning. We also make use of some parts of AbsProb-PI algorithm, for instance to calculate the vector $\mathbf{C}$ or the weight function $W(\sigma, \alpha)$. In the following algorithms we also consider a maximum set $\mathcal{F}_{\max}$, i.e., $\mathcal{F}_{\max}$ contains all possible features that algorithms consider for state description.

3.1 Backward Approach

In the backward (BA) approach we start with the largest set of features and eliminates one feature per iteration to find a compact set of features that still produces good abstract policies. Summarizing, when we have many features, we have small abstract state and generalization is more difficult to be made. So, for each feature eliminated, abstract states get larger by grouping together other abstract states that were before separated by these features. Given the previous definition, we define the backward approach:

```
1. Initiate the set of features $\mathcal{F} \leftarrow \mathcal{F}_{\max}$
2. While $|\mathcal{F}| > 1$, i.e., not all features were eliminated:
(a) For each feature $f \in \mathcal{F}$
      i. Simulate the elimination of $f$ by constructing $\mathcal{F}_{f^-} = \mathcal{F} - \{f\}$
      ii. Evaluate the set of features $\mathcal{F}_{f^-}$ by defining a value $V(\mathcal{F}_{f^-})$
(b) Find the most irrelevant feature for state abstraction $f^* = \arg\max_{f \in \mathcal{F}} V(\mathcal{F}_{f^-})$
(c) Eliminate $f^*$ of the set $\mathcal{F}$, i.e., $\mathcal{F} \leftarrow \mathcal{F} - \{f^*\}$
3. Return the list of eliminated features in order of relevance
```

At the end of the algorithm, we will get the features ordered by relevance in the state abstraction of an MDP problem as it was found by the approach, i.e., the first feature eliminated in this algorithm is the most irrelevant to describe the states compared with the rest of features, and so on. Thus, the last feature eliminated in this algorithm is the most relevant for the state abstraction. This algorithm depends on the function $V(\mathcal{F}_{f^-})$ that defines the relevance of feature f in set $\mathcal{F}$.

Most of the computational complexity of the backward approach is spent in the step 2, whereas inside step 2 most of the computational complexity is spent in step 2(a)ii. The algorithm goes through step 2 for $|\mathcal{F}|-1$ times, and the number of execution of step 2(a)ii decays linearly at each iteration: $|\mathcal{F}|, |\mathcal{F}|-1, \ldots, 3, 2$. Then we have that

$$T_{BA}(|\mathcal{F}|) \approx (|\mathcal{F}|-1)\left(\frac{|\mathcal{F}|+2}{2}\right) T(V(\mathcal{F}_{f^-})).$$

To define the relevance of a feature we consider a greedy strategy. The greedy strategy considers any iteration as if it was the last one. In this case it is enough to evaluate a set of features by evaluating the optimal abstract policy within the abstract state space $\mathcal{S}_{ab}^{\mathcal{F}_{f^-}}$. Then, we define the value of a set of features by:

$$\begin{aligned} \hat{\pi}^*_{ab} &\leftarrow \texttt{AbsProb-PI}(\mathcal{S}_{ab}^{\mathcal{F}_{f^-}}, N_{\text{steps}}) \,, \\ V(\mathcal{F}_{f^-}) &= \mathbf{b}^{0\top}\left(\mathbf{I} - \gamma \mathbf{T}^{\hat{\pi}^*_{ab}}\right)^{-1} \mathbf{r} \,. \end{aligned} \tag{1}$$

In this case, the time spent in the algorithm is

$$T_{BA}(|\mathcal{F}|) \approx (|\mathcal{F}|-1)\left(\frac{|\mathcal{F}|+2}{2}\right) N_{\text{steps}} K_{BA},$$

where K_{BA} is the time spent in average at each iteration of the AbsProb-PI algorithm within the backward approach.

From the backward approach we can derive a weighted estimation heuristic. This heuristic produces an alternative so that the execution time is decreased in relation to the backward approach with AbsProb-PI. Remember that to solve the MDP problem, the AbsProb-PI algorithm calculates the vector $\mathbf{C}$. This vector represents the expected accumulate discounted occupation of each state $s \in \mathcal{S}$ for the MDP when following policy π_{ab}. The heuristic consists in taking such values to weight an average of the previous policy, i.e., the policy $\hat{\pi}_{ab}^{\mathcal{F}_{f^-}}$ is estimated at any iteration by:

$$\hat{\pi}_{ab}^{\mathcal{F}_{f^-}}(\sigma, \alpha) = \frac{\sum_{s \in \mathcal{S}_\sigma} C(s) \hat{\pi}_{ab}^{\mathcal{F}}(\sigma^{\mathcal{F}}(s), \alpha)}{\sum_{s \in \mathcal{S}_\sigma} C(s)} \,,$$

where the policy $\hat{\pi}_{ab}^{\mathcal{F}}$ is obtained with AbsProb-PI within the set $\mathcal{F}$ and the function $\sigma^{\mathcal{F}} : \mathcal{S} \to \mathcal{S}_{ab}^{\mathcal{F}}$ maps each state into the current set of abstract states. The values $V(\mathcal{F}_{f^-})$ is calculated by $V(\mathcal{F}_{f^-}) = \mathbf{b}^{0\top}\left(\mathbf{I} - \gamma \mathbf{T}^{\hat{\pi}_{ab}^{\mathcal{F}_{f^-}}}\right)^{-1} \mathbf{r}$.

In the heuristic version of backward approach the time spent in the execution can be approximated by

$$T^h_{BA}(|\mathcal{F}|) \approx (|\mathcal{F}|-1) N_{\text{steps}} K_{BA} + (|\mathcal{F}|-1)\left(\frac{|\mathcal{F}|+2}{2}\right) K_{BA} \,,$$

where the first term is spent to calculate $\hat{\pi}_{ab}^{\mathcal{F}}$ at each iteration and the second term is spent to calculate the values $V(\mathcal{F}_{f^-})$. We assume that one iteration of AbsProb-PI (K_{BA}) is enough time to estimate $\hat{\pi}_{ab}^{\mathcal{F}_{f^-}}$ and to calculate its value. If N_{steps} dominates $\frac{|\mathcal{F}|+2}{2}$ we have that our heuristic is at least $\frac{|\mathcal{F}|+2}{2}$ times faster.

3.2 Forward Approach

The forward (FO) approach is exactly the opposite of backward approach and starts with a unique abstract state that represents all the states in $\mathcal{S}$, i.e., no feature in the state description. A feature is added to generate another set of abstract states in each iteration until all the features be added. Given the previous definition, we define the forward approach:

```
1. Initiate the set of features $\mathcal{F} \leftarrow \emptyset$, and the complement set $\mathcal{F}^C \leftarrow \mathcal{F}_{\max}$
2. While $|\mathcal{F}^C| > 1$, i.e., all the features are not added:
  (a) For each feature $f \in \mathcal{F}^C$
      i. Simulate the addition of $f$ by constructing $\mathcal{F}_{f+} = \mathcal{F} \cup \{f\}$
      ii. Evaluate the set of features $\mathcal{F}_{f+}$ by defining a value $V(\mathcal{F}_{f+})$
  (b) Find the most relevant feature for state abstraction $f^* = \arg\max_{f \in \mathcal{F}^C} V(\mathcal{F}_{f+})$
  (c) Add $f^*$ into the set $\mathcal{F}$, i.e., $\mathcal{F} \leftarrow \mathcal{F} \cup \{f^*\}$ and $\mathcal{F}^C \leftarrow \mathcal{F}^C - \{f^*\}$
3. Return the list of added features in order of relevance
```

Similar to the backward approach, at the end of the algorithm, we will get the features ordered by relevance in the state abstraction of an MDP problem, i.e., the first feature added in this algorithm is the most relevant to describe the states compared with the rest of features, and so on. Thus, the last feature added in this algorithm is the most irrelevant for the state abstraction.

Here we propose two definitions to function $V(\mathcal{F}_{f+})$: based on AbsProb-PI and a gradient-based estimation. The first definition is the same used in backward approach and it is given in equation 1 and just like in backward approach the time to run the algorithm is given by:

$$T_{FO}(|\mathcal{F}|) \approx (|\mathcal{F}| - 1) \left(\frac{|\mathcal{F}| + 2}{2} \right) N_{\text{steps}} K_{FO} ,$$

where K_{FO} is the time spent in average at each iteration of the AbsProb-PI algorithm within the forward approach. Note that the average K_{FO} is smaller than K_{BA}. First, in one iteration of AbsProb-PI algorithm the larger abstract state space the longer time spent (see steps 2(e) and 2(f) in AbsProb-PI algorithm). Second, in backward approach most of the executions of AbsProb-PI algorithm is done with large abstract state spaces, whereas in forward approach most of the executions of AbsProb-PI algorithm is done with small abstract state spaces.

We also consider a heuristic to the forward approach. In the backward approach we define a heuristic by estimating a policy $\hat{\pi}_{ab}^{\mathcal{F}_{f-}}$. Such estimation cannot be done since abstract states in space $\mathcal{S}_{ab}^{\mathcal{F}}$ are expanded in the new space $\mathcal{S}_{ab}^{\mathcal{F}_{f+}}$. But, remember that when we solve the MDP problem with the AbsProb-PI algorithm, at each iteration we calculate the gradient $W(\sigma, \alpha)$ for each abstract state $\sigma \in \mathcal{S}_{ab}$ and action $\alpha \in \mathcal{A}_{ab}$ and this gradient indicates how and much the current policy can be improved. Given the policy $\hat{\pi}_{ab}^{\mathcal{F}}$ obtained with AbsProb-PI, the gradient estimation defines the value of set of features $\mathcal{F}_{f+}$ by:

$$V(\mathcal{F}_{f+}) = \sum_{\sigma \in \mathcal{S}_{ab}^{\mathcal{F}_{f+}}} \max_{\alpha \in \mathcal{A}_{ab}} W(\sigma, \alpha) ,$$

where $W(\sigma, \alpha)$ is obtained through one iteration of the AbsProb-PI algorithm by considering the abstract state space $\mathcal{S}_{ab}^{\mathcal{F}_{f+}}$ and initial policy $\hat{\pi}_{ab}^{\mathcal{F}}$. The time spent in the execution can be approximated by:

$$T_{FO}^{h}(|\mathcal{F}|) \approx (|\mathcal{F}|-1) N_{\text{steps}} K_{FO} + (|\mathcal{F}|-1) \left(\frac{|\mathcal{F}|+2}{2} \right) K_{FO} \,.$$

Again, if N_{steps} dominates $\frac{|\mathcal{F}|+2}{2}$ we have that our gradient-based heuristic is at least $\frac{|\mathcal{F}|+2}{2}$ times faster.

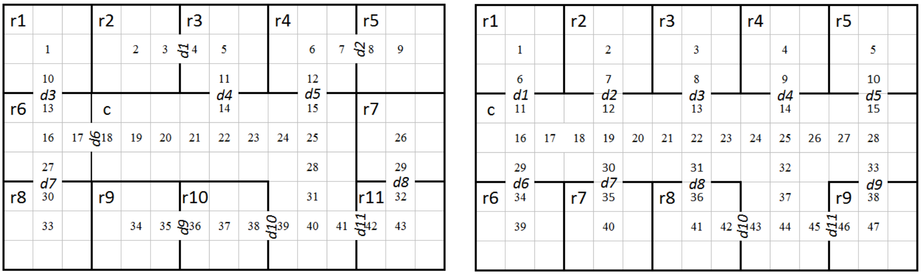
4 Experiments

To evaluate our algorithms we conducted experiments in three simulated robotic navigation environments (Figure 1). We use the environment present in [7] and define another two environments for evaluating.

Environments. Silva, Pereira and Costa [7] define the environment as composed of three types of objects: rooms (ri), doors (di) and a corridor (c). The robot can access some rooms immediately from the corridor and, depending on the environment, the robot can also access other rooms through these rooms. The robot can choose between four ground actions (actions N, S, E and W) which move the robot to the chosen direction with probability 0.9 if the direction correspond to a ground state and do not move otherwise. The first environment (Figure 1a) is the environment used in the experiments in [7] and is composed by: 43 discrete ground states (s_1, s_2,..., s_{43}); 11 rooms ($r1$, $r2$,..., $r11$); and 11 goals positions which represent different tasks, so for each task a room center is chosen as goal (s_1, s_2, s_5, s_6, s_9, s_{16}, s_{26}, s_{33}, s_{34}, s_{37} and s_{43}). The second environment (Figure 1b) is composed by: 47 discrete ground states (s_1, s_2,..., s_{47}); 9 rooms ($r1$, $r2$,..., $r9$); and 9 goals positions (s_1, s_2, s_3, s_4, s_5, s_{39}, s_{40}, s_{41} and s_{47}). In this environment the robot can access every room from the corridor, and between the rooms there is no doors for access. The third environment (Figure 1c) is composed by: 55 discrete ground states (s_1, s_2,..., s_{55}); 9 rooms ($r1$, $r2$,..., $r9$); and 9 goals positions (s_1, s_2, s_5, s_8, s_{11}, s_{45}, s_{46}, s_{49} and s_{55}). In this environment the robot can access every room from the corridor, and there is access between some rooms.

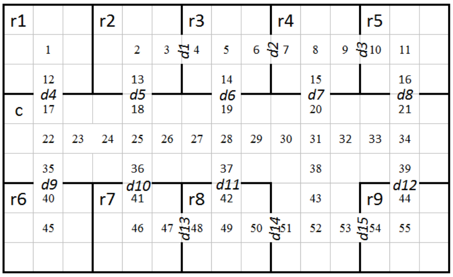
The state description consider the set of features related to local observations $\mathcal{F}_L$ = {see_empty_space, see_door_far, see_room, see_corridor, see_ambience, see_anything} and the set of features related to the goal position $\mathcal{F}_G$ = {near_goal, almost_near_goal}. The set $\mathcal{F}_L$ is composed of some features (see_ambience and see_anything) that are correlated to others. These features were added in order to compare more general features and more specific ones. A state s presents the feature see_ambience, if s presents any one of the features in {see_door_far, see_room, see_corridor}, and s presents the feature see_anything, if s presents any one of the features in $\mathcal{F}_L$ − {see_anything}.

With $\mathcal{F}_L$ we define sets of features $\mathcal{F}_L^h$ and $\mathcal{F}_L^{op}$. These sets represent features in the current state that are or not in the direction of goal. In the first, the suffix



(a) Environment 1

(b) Environment 2



Feature	Notation
see_empty_space	Es
see_door_far	Df
see_room	Sr
see_corridor	Sc
see_ambience	Sa
see_anything	An
in_room	In
near_to_goal	Ng
almost_near_to_goal	Ang

(c) Environment 3

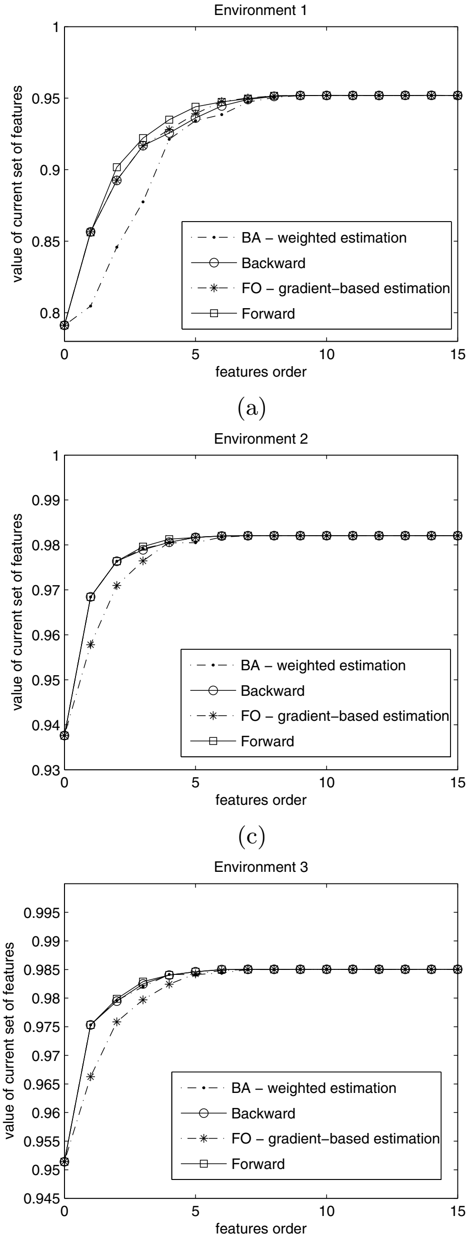
(d) Feature notation

Fig. 1. The simulated robotic navigation environments (a, b and c) and the notation for the features (d)

'head_to_goal' is added to the original feature notation in $\mathcal{F}_L$ and, in the latter, the suffix 'opposite_to_goal' is added to original feature notation. The features in $\mathcal{F}_G$ describe the distance from a state to the goal. The state is near_goal if it is at a Manhattan distance of 5 to the goal, and is almost_near_goal if it is at a Manhattan distance of 8 to the goal and is not near_goal. The final set which can describe a state is $\mathcal{F} = (\mathcal{F}_L^h \cup \mathcal{F}_L^{op} \cup \mathcal{F}_G \cup \{\text{in\_room}\})$, where in_room is a local observation feature without goal relation.

As an example, consider the state s_{16} in the environment 2 (Figure 1b) and s_1 as the current goal state. The state s_{16} is described by the features: see_door_far_head_to_goal at north ($d1$) which induces see_ambience_head_to_goal and see_anything_head_to_goal; see_door_far_opposite_to_goal at south ($d6$) and see_empty_space_opposite_to_goal at east (s_{17}) which induces see_ambience_opposite_to_goal and see_anything_opposite_to_goal; also the state is near_goal. For a simpler reference to features we present the notation in Figure 1d, also for the features head to goal is added the suffix '_H' in the feature notation, and similarly for the features opposite to goal is added the suffix '_O'.

For the variable N_{steps} of AbsProb-PI algorithm we used 280 iterations. For each environment we create a number of goals, which we describe earlier, and for each goal the state abstraction was produced. Then the state abstraction for all goals was evaluated in the same environment.



(a)

(c)

(e)

BA	BA_W	FO	FO_G
In	Es_H	In	In
Ng	Sa_H	Ang	Ng
Sa_H	Es_O	Df_H	Sa_H
Ang	Ang	Df_O	Sr_H
Sa_O	Sr_H	Ng	Df_O
Sr_H	Sc_O	Sr_H	Ang
Es_H	Ng	Es_O	Sr_O
Sr_O	In	Sc_O	Es_O
Es_O	Sa_O	Es_H	Es_H
Sc_H	Df_H	An_H	Sc_O
Sc_O	Df_O	Sa_H	Sa_O
An_O	An_H	Sc_H	Df_H
Df_O	Sr_O	An_O	Sc_H
Df_H	An_O	Sa_O	An_O
An_H	Sc_H	Sr_O	An_H

(b)

BA	BA_W	FO	FO_G
Es_H	Es_H	Es_H	In
In	In	In	Sr_H
Sr_H	Df_H	Df_O	Ng
Ng	An_H	Ng	Es_H
Sa_H	Sc_O	Df_H	Sa_H
Ang	Ng	Ang	Ang
Sa_O	Ang	Sc_O	Sr_O
Sc_O	Sr_H	An_H	Df_O
An_O	An_O	Es_O	Es_O
Es_O	Sa_H	Sr_H	Sc_H
An_H	Sa_O	An_O	Sa_O
Df_H	Sc_H	Sa_O	An_O
Df_O	Df_O	Sr_O	Df_H
Sc_H	Sr_O	Sa_H	Sc_O
Sr_O	Es_O	Sc_H	An_H

(d)

BA	BA_W	FO	FO_G
Es_H	Es_H	Es_H	In
In	Sa_H	Df_H	Sr_H
An_H	Sr_H	Es_O	Ng
Sr_H	Sc_O	In	Es_H
Sr_O	Sa_O	Sc_O	Sa_H
Ang	Ang	Ang	Sc_O
Ng	In	Ng	Ang
Df_O	Sc_H	Sa_H	Df_O
Sc_H	Ng	Sr_H	Sr_O
Sa_O	Sr_O	Sa_O	Df_H
Sc_O	Df_H	An_O	An_H
Sa_H	Df_O	Df_O	Es_O
Es_O	Es_O	Sr_O	Sa_O
An_O	An_H	An_H	Sc_H
Df_H	An_O	Sc_H	An_O

(f)

Fig. 2. The results of the algorithms at left and the feature order obtained for each algorithm at right

Results. The results for each algorithm: backward (BA), weight estimation (BA_W), forward (FO) and gradient-based estimation (FO_G) is presented in Figure 2. The results of BA and BA_W algorithms are presented in order of relevance. Then the first value in the results (Figure 2a, 2c and 2e) correspond to the value without any feature in the state abstraction and the last one correspond to the value of all possible features $\mathcal{F}_{\max}$ in the state abstraction. Similarly, the feature order obtained with BA and BA_W are presented. This was done to compare easily the results of all algorithms with each other.

For the environment 1 (Figure 1a) in Figure 2a we can see that the FO approach has the best result. With the exception of BA_W algorithm, the BA and FO_G algorithms has close values, where the heuristic has the advantage of being faster. In the Figure 2b the feature order obtained with the algorithms is presented, so the set of features $\{\mathrm{In}, \mathrm{Ang}, \mathrm{Df\_H}, \mathrm{Df\_O}, \mathrm{Ng}, \mathrm{Sr\_H}, \mathrm{Es\_O}, \mathrm{Sc\_O}\}$ obtained with FO approach is the best choice for state description in environment 1. The first eight features in feature order compose this set because after this point the value of the current set of features (value of optimal abstract policy) is constant, without an improvement. For the environment 2 (Figure 1b) in Figure 2c we can see that the FO_G algorithm has the worse result. The others (BA, BA_W and FO) present values that are very close to each other for each size of feature set, but the feature order for all of them are different as is presented in Figure 2d. Even then, the heuristic BA_W has the advantage of being faster. Thus, the set of features $\{\mathrm{Es\_H}, \mathrm{In}, \mathrm{Df\_H}, \mathrm{An\_H}, \mathrm{Sc\_O}, \mathrm{Ng}\}$ obtained with BA_W algorithm is the best choice for state description in environment 2. For the environment 3 (Figure 1c) we can see in Figure 2e that the behavior of result values are similar with the results of environment 2. Then again the FO_G algorithm has the worse result, and among the other algorithms the BA_W is the best choice to this environment for being faster, with $\{\mathrm{Es\_H}, \mathrm{Sa\_H}, \mathrm{Sr\_H}, \mathrm{Sc\_O}, \mathrm{Sa\_O}, \mathrm{Ang}\}$ as the appropriate set of features for state description that produce the best state abstraction.

5 Conclusions

The state abstraction can be made through relational representations of an MDP environment, which describes relations between objects of the environment. These relations can be represented by features which may aggregate states with similar representation. This aggregation allows the transfer of knowledge, since information obtained for a state can be transferred to similar ones, i.e., states with similar features, therefore making the planning faster.

In our approach, we use propositional features for state description, and the state abstraction obtained with such features is then evaluated to decide if the current feature selection is appropriate or not. We presented two different approaches (BA and FO), and two heuristics (BA_W and FO_G) derived from them that represent an alternative to find the appropriate set of features for an MDP problem. As we see in the Section 4, at least one of original approaches (BA and FO) have the best result for each environment, in the same way one of the heuristics (BA_W and FO_G) present satisfactory results, i.e., close to the best values having the advantage of being faster than the others. Thus the heuristics can be a good choice for feature selection.

By analysing the feature order for each environment we cannot define a pattern in relation to the order of features obtained with each algorithm. Therefore the set of best features to state description for an environment is not the same to every environment, since they present different structure navigation. Future research should evaluate the quality of the features in an different environment from where it was produced, since we created and evaluated the features in the same environment. Besides that, it is interesting to adapt and to apply the feature selection algorithms in a reinforcement learning environment [13].

References

1. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming, 1st edn. John Wiley and Sons, New York (1994)
2. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)
3. Walsh, T.J., Li, L., Littman, M.L.: Transferring state abstractions between mdps. In: ICML Workshop on Structural Knowledge Transfer for Machine Learning (2006)
4. Frommberger, L., Wolter, D.: Structural knowledge transfer by spatial abstraction for reinforcement learning agents. Adaptive Behavior 18, 507–525 (2010)
5. Otterlo, M.V.: Reinforcement learning for relational mdps. In: Machine Learning Conf. of Belgium and the Netherlands, pp. 138–145 (2004)
6. Matos, T., Bergamo, Y.P., Silva, V.F., Costa, A.H.R.: Stochastic Abstract Policies for Knowledge Transfer in Robotic Navigation Tasks. In: Batyrshin, I., Sidorov, G. (eds.) MICAI 2011, Part I. LNCS, vol. 7094, pp. 454–465. Springer, Heidelberg (2011)
7. da Silva, V.F., Pereira, F.A., Costa, A.H.R.: Finding Memoryless Probabilistic Relational Policies for Inter-task Reuse. In: Greco, S., Bouchon-Meunier, B., Coletti, G., Fedrizzi, M., Matarazzo, B., Yager, R.R. (eds.) IPMU 2012, Part II. Communications in Computer and Information Science, vol. 298, pp. 107–116. Springer, Heidelberg (2012)
8. Farahmand, A.M., Ghavamzadeh, M., Szepesvári, C., Mannor, S.: Regularized policy iteration. In: Neural Information Processing Systems, pp. 441–448 (2008)
9. Painter-Wakefield, C., Parr, R.: Greedy algorithms for sparse reinforcement learning. In: International Conference on Machine Learning (2012)
10. Geramifard, A., Doshi, F., Redding, J., Roy, N., How, J.: Online discovery of feature dependencies. In: International Conf. on Machine Learning (ICML), pp. 881–888. ACM, New York (2011)
11. Kroon, M., Whiteson, S.: Automatic feature selection for model-based reinforcement learning in factored mdps. In: ICMLA - International Conf. on Machine Learning and Applications, pp. 324–330 (2009)
12. Chankong, V., Haimes, Y.Y.: Multiobjective Decision Making: Theory and Methodology. North-Holland, New York (1983)
13. Ribeiro, C.H.C.: A tutorial on reinforcement learning. In: International Joint Conf. on Neural Networks. INNS Press, Washington DC (1999)

AI Based Fall Management Services – The Role of the *i-Walker* in I-DONTFALL

Ulises Cortés[2,⋆], Antonio Martínez-Velasco[2],
Cristian Barrué[2,⋆⋆], and Roberta Annicchiarico[1]

[1] Fondazione Santa Lucia, Rome, Italy
r.annicchiarico@hsantalucia.it
[2] Universitat Politècnica de Catalunya, Barcelona, Spain
{ia,cbarrue}@lsi.upc.edu

Abstract. Falls are a major international public health challenge because of the myriad effects on older individuals: direct physical and psychological injury; fear of falling and activity limitation; reduced social participation and quality of life; increased dependence; and because of the economic impact on health and social care providers [12].

In nowadays ageing society, many people require assistance for *safe* pedestrian mobility. In some cases, assistive devices require a certain degree of autonomy when the persons' disabilities difficult manual control. Our aim is to develop an open and innovative reference architecture, based upon ontologies and agent services, that will allow plug and play and cost-effective interconnection of existing and new services in all domains required for the independent and autonomous living of the elderly and their enhanced Quality of Life (QoL).

1 Introduction

Demographic change and aging of the population is one of the socioeconomic challenges societies have to face in the 21st century. Worldwide, the number of persons over 60 years is growing faster than any other age group. The number of this age group was estimated to grow to almost two billions by 2050. By that time, the population of older people will be much larger than that of children under the age of 14 years for the first time in human history. Moreover, the oldest segment of population, aged 80 and over, particularly prone to falls and its consequences is the fastest growing within older population expected to represent 20% of the older population by 2050 [20].

Of all world regions, Europe has the highest proportion of population aged 65 or over, a statistic that becomes more pessimistic according to the baseline projection of Eurostat, which shows that this percentage will almost double to more than 28% in the year 2060. Besides, while the share of the youngest, and especially the working

[⋆] All authors are partially supported by the project "Integrated prevention and Detection sOlutioNs Tailored to the population and Risk Factors associated with FALLs (I-DONTFALL). CIP-ICT-PSP-2011-5 #297225. The views expressed in this paper are not necessarily those of the consortium I-DONTFALL.

[⋆⋆] Author is partially supported by the project *Sistema Inteligente* i-Walker*: rehabilitacion colaborativa (SiRC)*. TEC2011-29106-C02-02.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 395–406, 2013.

population is shrinking, the baby boom generation will retire and the number of people between 65 and 80 years old will increase almost by 40% between 2010 and 2030.

There are currently approximately 45M people in Europe who report a long standing health problem or disability [1]. Further, the aging population means that more people will have to live with some sort of disability. Recent figures from the 2008-based national population projections EUROPOP20082 show that the share of people aged 65 years or over in the total population is projected to increase from 17.1% to 30.0% and the number is projected to rise from 84.6M in 2008 to 151.5M in 2060. This means there will be more than 80M people with several needs for care by 2050. It is estimated that the cost of this care will range from 5% to 20% of GDP by 2025.

In Mexico, the proportion of elderly population presents the fastest growing in the past 15 years. In accordance with the projections of the National Population Council, the current total is 9.4 million people, which represents 8.6% of the total population, and it will increase at an annual rate of 3.9% [5].

1.1 The Cost of Falling

For human beings getting old implies a reduction of functional reserve, which is the reserve margin of safety that organs have to take all the work they need to do (usually they are able to do more work than necessary, so that the body is confident that they will not fail under stress). With ageing this reserve decreases, especially in the brain and the musculoskeletal system, these organs therefore become the weakest point in case of aggression to the body from disease, and therefore often fail even though the disease is located elsewhere in the body. Illness or weakness of the body of aged people can result in falls. Falls are therefore a sign of *weakness* of the musculoskeletal system and the organs of the rest of the body, and therefore are very good indicators of lack of functionality reserve. Falls are prevalent, dangerous, and costly. Even falls that do not result in injury can have serious consequences. Psychological trauma and fear-of-falling produce a downward spiral of self-imposed activity reduction which leads to loss of strength, flexibility and mobility, thereby increasing the risk of future fall. However, falls are not a normal part of aging [18].

In order to measure and quantify a persons fall risk, two international scales are considered: the Morse Fall Scale (MFS) and the Falls Efficacy Scale (FES).

The causes leading to falls can be classified in 4 categories:

1. Defective gait: due to by various conditions such as structural disorders (arthritis, deformities) or central nervous system deficits (stroke, Parkinson, *etc*) or peripheral nervous system (neuropathy, *etc*), or even diabetes (diabetes acts through diabetic neuropathy).
2. Poor balance: its origin can be set to a vision problem, inner ear defect (which is responsible for balance (vestibular system or labyrinth)), lack of proprioceptive information or Parkinsons disease.
3. Accident: due to several factors including improper footwear, inadequate home environment items (carpets, wrongly located objects, *etc*.).
4. Blackout: caused by sudden low pressure, momentary cardiac syncope, paroxysmal asystole, bradykardia, *etc*.

The three main consequences of falls in ageing people are:

1. Fractures, especially hip fracture: causing death in more than 30% of cases, long term effects in a high percentage of survivors, and high costs on public health systems.
2. Long-lie syndrome: a product of long stays on the ground after falling, and because of the lack of functionality to rise, this situation can cause death from acute renal failure.
3. Fear of Falling: described extensively by Tinetti, the patient experiencing it, voluntarily reduces his activity, causing in turn a decrease of function in his system, which feeds the possibility of further falls [17], [18].

Studies from various countries have shown that about one third of people aged above 65 years fall each year. In UK primary care populations, the rate rises with age to over 60% of nonagenarians, and is generally higher in women and in socio-economically deprived populations. Fragility fractures are the commonest significant injury, 300,000 annually in the UK including upwards of 70,000 hip fractures [12].

1.2 State of the Art

The related investigations of ageing people and their real needs are not new, but they have not obtained the expected impact. Some reasons may be: study fragmentation, lack of critical mass, lack of vision of the relevant industry, and a missing political signal of favourable conditions for implementation and coordination on a large scale. In addition, the situation is not characterized by a lack of technologies, but by a lack of *complete* solutions, which should be easy to handle, easy to install, reliable, and affordable for the targeted people as well as the social security systems [7].

In recent years, ICT technologies and, specially, Artificial Intelligence have manifested their potential to enhance the autonomy and quality of life of elderly people, through boosting detection and/or prevention of elderly falls [15],[16] . Likewise, AI based solutions for fall detection and prevention (especially when deployed in the home environment) could significantly reduce the cost associated with elderly care (such as costs associated with the caretakers, personal nursing services, or the transfer to nursing homes). From a technological perspective a variety of ICT solutions for fall prevention and detection have been introduced spanning the areas of assistive training devices, biofeedback solutions, fall detectors, fall risk assessment systems [13] and more. Such solutions have been developed as enterprise products (*e.g.*, [19]), but also as part of R&D projects and research initiatives (*e.g.*, in the scope of the FALLWATCH (wearable miniaturized fall detector), CONFIDENCE (Independent living System), BIOTELEKINESY (home care service) projects).

The great interest in the treatment of disability has been reflected in the use of technological tools more or less sophisticated, the so-called *Assistive Technologies* (AT) [9], [7],[6],[4]. The AT can be defined as *technical devices that can eliminate or improve or compensate for functional limitations*. These tools are able to support people with disabilities, while making more effective and efficient interaction of the subjects themselves with their surroundings, physical and social.

Among the various technological tools, the walker is certainly a valid factor to increase the autonomy and to encourage integration of disabled people in social and family contexts, reducing the care burden of caregivers and improving motor performance. Many users are not prescribed a standard rollator because they lack the necessary cognitive skills or their impairment prevents them from steering it safely. So far this tool has not been used in practice in rehabilitation of post-stroke patients, as they are difficult to manage by the patient.

In the *SHARE-it* project we devised a walker modified to be accessible to post-stroke patients: the *i-Walker* [3], [8] (see §3). The use of this technology in this population is a clear case of success of Artificial Intelligence techniques in this field (see §3.2).

1.3 Plan of the Work

In §2 we explain our idea of an AI fall management system for elders which are likely to fall. In particular §3 addresses the assistive tool that we are proposing as the basis for a generic architecture to support mobility autonomy in the target population. We briefly describe a successful experiment with real patients. In §4 we propose our final remarks and conclusions.

2 New Approaches to Fall Management

The generic users of an intelligent fall management system can be defined as individuals which are likely to fall as a result of biological factors such as the decline of physical, cognitive and affective capacities, as well as other chronic illnesses/diseases. We need to distinguish among the various biological and disease related health factors, through considering fallers associated with balance / gait problems, cardiovascular diseases, vision problems, depression sufferers and more.

From the scenario depicted in §1.1 one can well understand the need for continued scientific research and the continuing interest in physician-directed to avoid individuals to fall or effectively rehabilitate those that have already fallen once or *secondary fallers*, to avoid them to fall again. These are two among the possible target groups. Physical mobility, the capability of autonomous movement, is necessary for the health and well-being of all persons, but is especially important in older adults because a variety of factors impinge upon mobility with aging.

As falls are not a normal part of aging, they may be preventable to a large degree. As obvious as it may sound, a lack of knowledge about risk factors and how to prevent them contributes to many falls. Lack of knowledge leads to lack of preventive action, resulting in falls. Nowadays, risk factors for falls have been identified and there are many screening tools available to determine the presence of balance and mobility disorders. Still, those are not integrated or they are not well tested to be widely delivered to the target population. Also, the high degree of variability is an issue.

There is a growing attention in rehabilitation research in new technologies that are able to favor the outcome motor and to reduce the functional disability in post-falling patients, so as to indirectly limit the socio-assistential costs.

Computer-aided architectures for elderly care are typically structured in three layers: sensors to capture the data on the user's situation, interpretation methods to *understand* the situation including the context, and services, which provide help or intervention based on the support system's understanding of the situation.

Our general aim is to develop an open and innovative reference architecture, based upon ontologies and agent services to do the reasoning, understanding and communication, that will allow plug and play and cost-effective interconnection of existing and new services in all domains required for the independent and autonomous living of the elderly and their enhanced Quality of Life (QoL).

The above user groups are in need of solutions that could minimize or remedy the risk of falling (*prevention*), while at the same time supporting them efficiently in the case of falls (*detection*). Such prevention and detection solutions (such as the one proposed by the I-DONT-FALL consortium [9]) should in particular address the users needs to (*i*) Gain instant and easy access to health and social services (*e.g.*, in the case of fall detection), including the ability to leverage tele-care services and, (*ii*) Undertake in-home training (*e.g.*, balance, mental training), which could help them reduce (*i.e.* prevent) fall incidents. At the same time, some activities could facilitate changes to the users behavioural factors, such as everyday habits that lead to a healthier lifestyle (as a prevention discipline). Trustful intelligent fall detection management services for the automatic and real time detection of elderly/patients falls are a real need for a growing segment of the population. The main characteristics of the I-DONT-FALL platform will be the following:

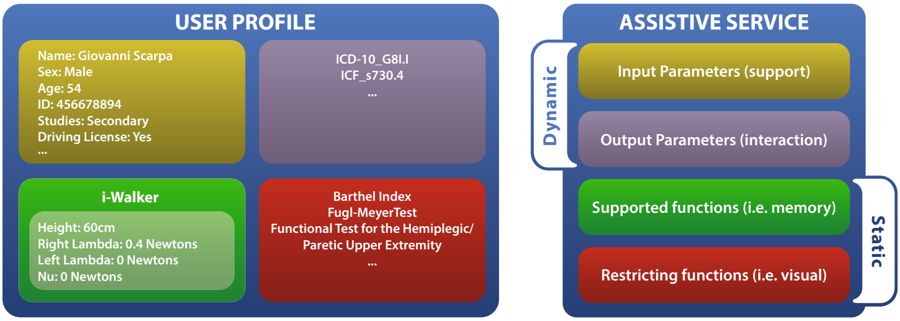


Fig. 1. Personalized profiles and Services

Integrated. The platform will integrate a host of detection and prevention functionalities and services, which will be empowered by a large pool of leading edge technologies including wearable fall detection devices, balance/motor training applications and devices, an Inertial Measurement Unit, a Healthcare Enterprise Resource Planning system along with Electronic Health Records (EHR) management technologies, telecare/teleassistance services for fallers, as well as technologies for mining, processing and managing users data and fall-related knowledge.

Configurable. The platform will enable flexible configuration of its functionalities on the basis of easy-to-use deployment, administration and management tools. These tools will be offered to professionals in order to enable them to customize the platform to the needs of the potential fallers, taking into account the fallers profile, medical record, preferences and fall-related root causes.

Programmable. The platform will provide open APIs (and associated development tools) enabling (medical application) developers and integrators to implement new innovative ideas that could leverage the platforms capabilities in order to serve/fulfill new requirements. To this end, the APIs will allow developers/integrators to leverage data streams stemming from sensors, access EHR and fall knowledge data, control sensors and actuators, specify rule-based detection functionalities and more.

Personalized. The solutions to be deployed with the I-DONT-FALL platform will be personalized to the needs of the specific users of the fall-related functionalities, notably in a home environment. To this end, the platform will provide access to profile data, demographics information, medical records, as well as information about the surrounding in-home environment of the fall sufferer, see figure 1.

In-Home. (yet remotely managed) The platform will be deployable within the home environment of potential fallers. This in-home nature of the platform is perfectly aligned with the personalization objectives (*i.e.* since the in-home environment is largely due to personalization), while also serving important financial objectives (*e.g.*, the need to reduce hospitalization). Nevertheless, the in-home platform will be manageable from remote, through appropriate tools and techniques enabling communication and intervention of health professionals and care centers.

We foresee a larger target population including elderly, stroke/demented patients, patients with chronic conditions, individuals in risk of falling and, second-time fallers will benefit from more effective fall management services, as well as from the availability of more and more versatile devices and configurations.

2.1 Services

Autonomy for the elderly or people with disabilities does not only refers to mobility terms, but to a set of domains influenced by functioning, activity limitations, participation restrictions and environmental factors. Life areas related to activities and participation are such as learning and applying knowledge, general tasks and demands, communication, mobility, self-care, interpersonal interactions and relationships as well as community and social life. Our approach is based on the notion of a shared autonomy between the user and its own agent-based mediator with any information system at hand. Existing telematic healthcare systems that provide integrated services to users are not, to our taste, flexible enough to allow a real personalization and maybe now it is too expensive to change them. This fact motivates us to foster services personalization as the cornerstone of AT design. Personalization, in this case, involves using technology to accommodate the differences between individuals.

Personalization and adaptation must be part of an iterative process through time in the user ATs life in order to allow a seamless integration between them. The user needs from the beginning to be provided with personalized assistive tools able to adapt to his autonomy needs. The adoption of a personalized tool facilitates the integration and the acceptance of the technology by the user. It is also an essential requirement, in order to achieve the basic goal of these technologies, to complement as much in as possible part of the lost users autonomy caused by a disability.

The first block, the right part of figure 1 depicts all the arguments or variables that can configure a service or an assistive tool. The more open and flexible the service is, the more variables it will have available to be configured to adapt to the end user. The second block defines the outputs of the service or tool, the information that can be gathered from it. This information can be very useful in order to observe the behaviour of the service/tool and the user interaction with it. Interaction data can be used to take decisions about the service fine tuning or adaptation. Specially, if you have information from many users. Thus, our service characterization proposal contains four information modules: two static covering the more descriptive information about the functions supported and the functions that can be restrictive in the use of the service, and two dynamic modules containing configuration variables and data generated that will be changing along the life of the service instance.

3 *i-Walker*

The *i-Walker* is a robotic rollator that integrates sensors and actuators. It is a walking aid manipulated by both arms. It uses a standard walker frame modified for this purpose. Actuators are two hub motors integrated in the rear wheels and are used for braking or helping the user. Sensors are arranged in the frame to detect forces, tilt and movement. An integrated battery supplies power. Finally, a network of distributed micro controllers drives the system and records and provides information to the therapists.

The *i-Walker* does help passively detecting the force imposed by the user on the handles through its sensors, so it is possible to determine and adjust the amount of help that each motor should be giving to the side with a deficit. The support given by the *i-Walker* is passive because it is never pulling the user, only when pushing forces are detected on the handles the *i-Walker* applies helping strategies through its motors. This configurable amount of support given by each rear wheel allows to daily adapt the *i-Walker*, according to the increased capacity of the upper paretic and support the patient in applying the necessary forces during walking, see figure 2.

Four main services are provided by the *i-Walker* platform. Three are related to elder /impaired assistance. The fourth is used for data logging. A physiotherapist should plan all the assistance. Services provided are:

- Active motor assistance to compensate lack of muscle force on climbs.
- Active brake assistance to compensate lack muscle force on descents.
- Active differential assistance to compensate unbalanced muscle force.
- Recording of sensor measurements and actuators activities for later evaluation (left and right hand forces, normal forces, tilt and odometry)

The amount of helping percentage and braking force in each hand can both be determined by a doctor. Described strategies are not exclusive: we can have the user pushing the *i-Walker* going downhill and at the same time the walker relieving him from part of the necessary pulling/pushing force to move around. For safety reasons the *i-Walker* automatically stops when the user releases the handles, that is, when no forces are detected on them.

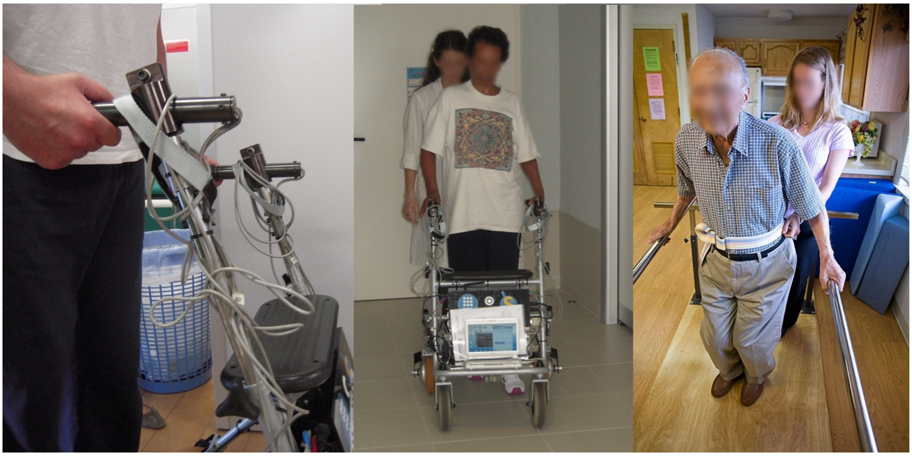


Fig. 2. The *i-Walker* compared with the use of parallel bars for rehabilitation in post-stroke individuals

3.1 AI on Board

The *i-Walker* platform has an integrated multi-agent system [2] that allows the configuration and management of the different services described previously. The therapist can configure the different helping parameters using the agents' interface and watch the logged data represented coming from the different sensor readings. The main objective of this agent system is to autonomously establish the helping parameters of the *i-Walker* depending on the user profile and his gait performance . In order to do that, the agents will use a specific disability ontology that allows to describe the user profile in terms of diseases, symptoms, assessments and assistive services prescribed (see figure 3). The continuous monitoring of the user's performance with the *i-Walker* will let the agent system to modify the user profile and dynamically adapt the support services that the *i-Walker* provides. In the right part of figure 3 we can see an overview of a multi-agent system that would support our personalization cycle [2], with different potential AI techniques associated with different agents to support their work. In figure 1 you can see a typical profile and its relation with the available services.

The *Profile Manager* would perform the task to structure a set of predefined user profiles using as distinctive parameters general disability treats like symptoms or diseases. The information gathered from experts to train the reasoning behind this agent could feed a Case-Based Reasoning. The *Profile Adapter* could use stored profile information to suggest adaptation schemas for the basic profile selected by the previous agent. This agent would interact with a medical expert that would introduce all the relevant user data. The *Service Selector* would use the information described in §2.1 to classify the available services and configure a service pack offered to each user profile depending on the functions to be supported. The *Service Adapter* could use interaction information and past experiences to adapt the input parameters of the services in order to increase personalization and usability on the users behalf. The *Monitor* would keep record of all the interaction information, analyze the data and raise possible behaviour patterns or interaction problems like described in the previous sections.

In the shared control paradigm tasks are performed by both the user and the system in perfect harmony so the two control inputs, the operator and the control system act in a *collaborative* manner. This means that the user is the one who makes the decisions, except possibly in hazardous situations.

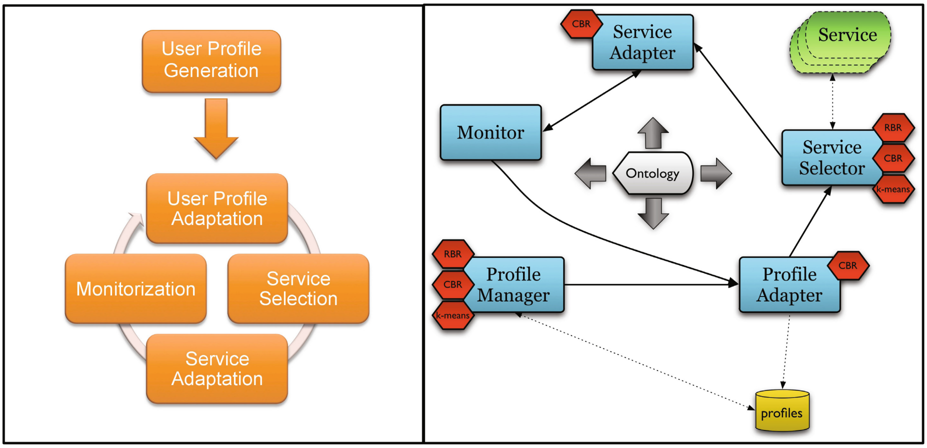


Fig. 3. Agent Based personalization cycle framework

Over the past decade, robotics has provided new material for rehabilitation of elder and/or neurological patients. Theoretically and in practice, the robots are useful tools for studying the evolution of recovering, for the quantitative assessment of therapeutic effect [10] and, finally, for the execution of the same exercise rehabilitation. Compared with conventional rehabilitation techniques, where it is difficult to quantify the dose, intensity and execution of the proposed exercise, the robot therapy is proposed as a valuable tool to study the processes of motor recovery.

Evidence suggests that this is the principle to follow for best results and to set up a personalized rehabilitation program that gradually increases the patient's active participation. The treatment should be challenging for the patient's residual abilities. Increasing somatosensory inputs to the hemiparetic hemisoma, through intensive treatment based on the repetition of gestures, the robotic therapy uses machine learning techniques to adapt itself to the user and learn from interaction. It is in fact based on explicit instructions that lead the patient to a self awareness of the motor task being carried out. The treatment should involve different modes of exercise, inducing passive movement, active or active-assisted, to be applied at different stages of rehabilitation according to the the patient's motor skills.

Without wishing to replace the physiotherapist, the robots can increase the opportunities available to the hemiplegic patients and the therapist, which can handle a means of training can provide a treatment with high intensity and specific, adaptable to different settings, including the home [11] .

3.2 A Small Step Ahead

The pilot study that we conducted, see [8], wanted to demonstrate how to effectively use Artificial Intelligence to build Assistive Technologies for therapeutics. For this study, the agent system only offered configuration and data logging services to the professional therapists, having disabled the autonomous support calibration. In this case, using the *i-Walker* to complete the rehabilitative intervention providing aid to a conventional system adding features that enable to achieve results not otherwise achievable. And above all it can be used in a population that normally is excluded: people with hemiparesis. The traditional walker, in fact, cannot be provided to a patient with hemiparesis due to the excessive gap between the forces of the two sides, which would imply a trajectory unbalanced and dangerous. The help that the system delivers is customized with the *i-Walker*, see the left part of figure 2. It also allows the hemiparetic patient to guide the *i-Walker* through the user's active participation supported by the shared control system, that provides only the amount of help needed in different situations, thus ensuring the user's autonomous movement in complete safety.

In particular, our study aimed to assess the possibility of using the *i-Walker* during the rehabilitation of the stroke patient, integrating it within the traditional therapy, then going to assess any differences in the recovery and management of resources and motor skills of patients who did use it. From the study, it has emerged a significant increase of the scales that take into account the walking (10MWT and Tinetti) and balance (Tinetti), in the experimental group who received therapy with the *i-Walker* for four weeks, compared with the control group that has not benefited with it. Each subject was clinically evaluated and undergoes specific assessments of several clinical, cognitive and psychological aspects

- Assessment of Mood : GDS
- Assessment of cognitive aspects : MMSE
- Assessment of spasticity Upper and lower libs : Ashwort scale
- Assessment of global functions: Canadian scale
- Assessment of ADL: Barthel Index

The data obtained show how the use of the *i-Walker* allows to improve the gait and in general the user's motor performance. From this it follows that the *i-Walker* can be a valuable aid in gait training for hemiparetic patients, resulting in a tool that they can effectively complement to traditional therapy. Moreover, this aid, having regard to its structural characteristics, is presented as a device able to increase the intensity and duration of rehabilitative treatment even in complete autonomy and safety of the patients.

4 Discussion and Conclusions

Humanity is in dire need of personalized and high-performance reasonably priced ambient assisted living for the elderly. Research prototypes are beginning to achieve the performance needed to make a difference in the lives of the elderly, while the market still offers only limited solutions to substantially prolonging the time that elderly people can live independently at home and be supported with relevant health and social care services in an integrated manner. In particular, there is a need to improve more the cost-to-benefit ratio of robot-assisted therapy strategies and their effectiveness for rehabilitation therapy.

As explained initial experimental results in post-stroke rehabilitation, see §3.2, show that an intelligent system coupled with the *i-Walker* can achieve the level of performance needed in real life [8]. The next step, is to extend this experimentation to individuals in risk of falling and, second-time fallers. In this phase we will heavily relay on personalization as explained on §3.1.

When talking about falls elders tend to describe them as a loss of balance whereas health care professionals generally refer to the consequence of falling, including injury and reduced QoL [22]. One has to remember that falls are the most prominent among the external causes of unintentional injury, since they account for approximately 40% of all injury deaths. We do believe that an intelligent fall management platform, as the one described in §2, based on the *i-Walker* can be used to help to prevent elders to fall, and therefore substantially prolonging the time that elderly people can live independently at home, if they decide so, by providing ICT-based safety and fall prevention/detection services. By using the proposed platform we expect to innovate at the medical/scientific level producing/deriving genuine medical knowledge associated with the applicability of certain solutions to particular elderly/patients needs. This knowledge will be accordingly used to drive the implementation of an intelligent decision support system.

The ideal target is combination of rehabilitative and integrated interventions to reduce reliance and negative impacts on hospitalization. Moreover, we expect that a conceptual model for safety solution services for elderly at home will be designed as a result of the intensive use of the platform by many users. Understanding *how* a safety service concept can give people a peace of mind and contribute in increasing the QoL while simultaneous preventing accidents, is a crucial part of the research. A good safety concept including ways of support could boost elderly and disable individuals' confidence, makes them feel more secure and would enable them to handle day-to-day activities with more ease [14].

Also, ethical issues including privacy, informed consent and others arising from the deployment and use of assistive technologies and services need to be further investigated. Wright, [21], argues that delivering personalized services opens up the possibility for the corresponding personal information to be stored and shared. Many users or they caregivers may think of this as intrusive and disturbing. Still, what is considered obtrusive and privacy invading differs by age group and culture. Once more tailored solutions are needed.

References

1. Applica CESEP Alphametrics. Men and women with disabilities in the EU (March 2007)
2. Barrué, C.: Personalization and Shared Autonomy in Assistive Technologies. PhD thesis, Universitat Politècnica de Catalunya (2012)
3. Barrué, C., Annicchiarico, R., Cortés, U., Martínez-Velasco, A., Martín, E.X., Campana, F., Caltagirone, C.: The i-Walker: An Intelligent Pedestrian Mobility Aid. In: Bichindaritz, I., Vaidya, S., Jain, A., Jain, L.C. (eds.) Computational Intelligence in Healthcare 4. SCI, vol. 309, pp. 103–123. Springer, Heidelberg (2010)

4. Boulos, M.K., Lou, R., Anastasiou, A., Nugent, C., Alexandersson, J., Zimmermann, G., Cortés, U., Casas, R.: Connectivity for Healthcare and well-being management: Examples from six European Projects. Int. J. Environ. Res. Public Health 6, 1947–1971 (2009)
5. Consejo Nacional de Población. Índices demográficos para adultos mayores (2010), `http://www.conapo.gob.mx/micros/anciano/adultmay.xls`
6. Cortés, U., Barrué, C., Campana, F., Martínez-Velasco, A., Urdiales, C., Annicchiarico, R., Caltagirone, C.: Assistive technologies for the new generation of senior citizens. the SHARE-it approach. International Journal of Computers in Healthcare 1(1), 35–65 (2010)
7. FAll detector for The Elder (2012), `http://www.project-fate.eu/`
8. Giulani, B., Cortés, U., Martínez-Velasco, A., Barrué, C., Annicchiarico, R.: The role of i-Walker in post-stroke training. In: Proceedings of the 15th International Conference of the Catalan Artificial Intelligence Association, CCIA (2012)
9. Integrated prevention and Detection sOlutions Tailored to the population and risk factors associated with FALLs, I-DONTFALL (2012), `http://www.risc-project.eu/`
10. Johnson, M.: Recent trends in robot-assisted therapy environments to improve real-life functional performance after stroke. J. Neuroeng. Rehabil. 3(29) (2006)
11. Johnson, M., Feng, X., Johnson, L., Winters, L.: Potential of a suite of robot/computer-assisted motivating systems for personalized, home-based, stroke rehabilitation. J. Neuroeng. Rehabil. 4(6) (2007)
12. Martin, F.C.: Next steps for falls and fracture reduction. Age and Ageing 38(6), 640–643 (2009)
13. Moylan, K., Binder, E.: Falls in older adults: risk assessment, management and prevention. Am. J. Med. 120(6), 493 (2007)
14. Ojasalo, J., Moonen, R., Suomalainen, N., Seppälä, H.: Designing services and systems for safety of elderly people at home: An ongoing empirical study. In: Rahal, I., Zalila-Wenkstern, R. (eds.) ISCA 19th International Conference on Software Engineering and Data Engineering (SEDE 2010), pp. 152–158. ISCA (2010)
15. Pineau, J., Montemerlo, M., Pollack, M.E., Roy, N., Thrun, S.: Towards robotic assistants in nursing homes: Challenges and results. Robotics and Autonomous Systems 42(3-4), 271–281 (2003)
16. Pollack, M.E.: Intelligent technology for an aging population: the use of AI to assist elders with cognitive impairment. AI Magazine, 9–24 (Summer 2005)
17. Tinetti, M.: Performance-oriented assessment of mobility problems in elderly patients. J. Am. Geriatr. Soc. 34, 119–126 (1986)
18. Tinetti, M., Speechley, M.: Prevention of falls among the elderly. The New England Journal of Medicine 320(16), 1055–1059 (1989)
19. Tunstall Group. Falls Management Solutions Sheet (March 2007)
20. United Nations. World Population Prospects: The 2004 Revision (March 2004)
21. Wright, D.: The dark side of ambient intelligence. The Journal of Policy, Regulation and Strategy for Telecommunications 7(6), 33–51 (2005)
22. Zecevic, A.A., Salmoni, A.W., Speechley, M., Vandervoort, A.: Defining a fall and reasons for falling: comparisons among the views of seniors, health care providers, and the research literature. Gerontologist 46(3), 367–376 (2006)

Performance Evaluation of Ranking Methods for Relevant Gene Selection in Cancer Microarray Datasets

Manju Sardana[1], Baljeet Kaur[2], and R.K. Agrawal[1]

[1] School of Computer and Systems Sciences, Jawaharlal Nehru University
New Delhi, India 110067
[2] Hansraj College, Delhi University, Delhi, India 110007
manjusardana12@yahoo.co.in, baljeetkaur26@hotmail.com,
rkajnu@gmail.com

Abstract. Microarray data is often characterized by high dimension and small sample size. Gene ranking is one of the most widely explored techniques to reduce the dimension because of its simplicity and computational efficiency. Many ranking methods have been suggested which depict their efficiency dependent upon the problem at hand. We have investigated the performance of six ranking methods on eleven cancer microarray datasets. The performance is evaluated in terms of classification accuracy and number of genes. Experimental results on all dataset show that there is significant variation in classification accuracy which depends on the choice of ranking method and classifier. Empirical results show that Brown Forsythe test statistics and Mutual Information method exhibit high accuracy with few genes whereas Gini Index and Pearson Coefficient perform poorly in most cases.

Keywords: Microarrays, Ranking method, Brown Forsythe test, Gini Index, Mutual Information, Pearson Coefficient, Cochran test, Adjusted Welch test.

1 Introduction

Cancer is one of the most invincible disease in the world leading to a large number of untimely deaths. All types of cancers are genetic i.e. they are triggered by altered genes. Advent of microarray technology has made simultaneous measurement of thousands of genes possible which may help in diagnosis of cancer and identification of genes responsible for it. Cancer diagnosis can be improved by accurate classification of gene microarray data. Data mining and machine learning techniques [12,15,17,24] have been extensively applied in this direction in the last few years. In general, the microarray data is characterized by a large number of genes (in thousands) and a small number of available samples (in tens) and hence suffers from the limitation called the curse of dimensionality [2]. Due to less number of available samples, classification of such data may suffer from the problem of over-fitting. This problem can be overcome by identifying a smaller number of genes accountable for a given disease. Since such genes are only a handful, there is a need to identify them. All the remaining genes are either redundant or irrelevant. Also, Guyon [13] observed

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 407–418, 2013.

that presence of redundant/irrelevant genes may deteriorate the performance of a classifier significantly. They also increase data acquisition cost and learning time. Hence dimensionality reduction is a crucial step for accurate classification of cancer microarray data.

Dimensionality reduction can be done in two ways [13]: feature extraction and feature selection. Feature extraction methods like Principal Component Analysis utilize all the information contained in the measurement space to obtain a new transformed space and then important features are selected from the new transformed space. These transformed features may provide a better discriminative ability but are unable to provide any physical meaning to the data. Feature selection refers to reducing the dimensionality of feature space by discarding redundant, noisy and irrelevant features. It leads to saving in measurement cost and the selected features retain their original physical interpretation. In addition, the retained features may be important for understanding the physical process related to these features. In microarray datasets one is not only interested in classifying the sample based on gene expression but also in identifying discriminatory features/genes responsible for a particular disease. Hence dimensionality reduction is normally carried out with feature/gene selection rather than feature extraction. There are two major approaches to feature selection [13,16]: filter approach and wrapper approach.

Filter methods employ statistical characteristics of data. They independently measure the importance of features without involving any classifier and hence are less computational intensive. The filter approach does not take into account the learning bias introduced by the final learning algorithm, so it may not be able to select the most relevant set of features for the learning algorithm. Many filter based feature/gene selection methods have been proposed in literature [11,16,17,22]. Wrapper methods on the other hand are computationally more expensive since a classifier must be trained for each candidate subset to find features suited to the predetermined learning algorithm. Wrapper approach is computationally feasible only for middle or low dimensional data.

Most of the studies on gene selection adopt gene ranking methods because of their computational efficiency. Gene ranking method aims to retain a certain number of genes, especially by ranking threshold, with scores determined according to a measure of relevance, discriminatory capability, information content or quality index. Some of the commonly used ranking methods for gene selection are Pearson correlation [20], Mutual Information [25], Gini Index [4], Cochran test statistics [6,7], Adjusted Welch test statistics [7,14,29] and Brown-Forsythe test statistics [5]. In literature each method is evaluated only on a handful of microarray datasets. Also to the best our knowledge, an extensive comparative analysis of these popular ranking techniques has not been done to determine a computationally simple and effective gene ranking method.

In this study we have investigated six popular gene ranking methods to determine a simple and efficient gene ranking method which can determine a smaller set of discriminatory set of genes to provide high classification accuracy. For this, we have considered eleven publicly available cancer microarray datasets which are considered challenging and are used by research community for the evaluation of their learning model. The paper is organized as follows: Section 2 contains a brief introduction to feature ranking techniques. Experimental setup and results are described in section 3 and finally, conclusion and future work is presented in section 4.

2 Feature Ranking Techniques

Feature ranking is one of the most widely explored feature selection techniques because of its simplicity and computational efficiency. Many feature ranking methods are proposed in this endeavor. Feature ranking approaches evaluate features using some statistical characteristics of the data. A scoring function is generally used to measure either correlation between individual feature and target class or scatter of a given feature among data of different class. Every feature is assigned a score and k-top most features are employed for classifier construction.

In many studies on classification of cancer microarray data, gene ranking approaches have been widely investigated. Some commonly used ranking methods for gene selection are Pearson Correlation, Mutual Information, Gini Index, Cochran test statistic, Adjusted Welch test statistic and Brown-Forsythe test statistics. A brief description of these statistical ranking methods is given below.

Let us assume that there are k (≥ 2) distinct classes for the problem under consideration and there are p features/genes and n samples. Suppose X_{fs} is the measurement of the feature f from sample s for $f = 1, 2, ..., p$ and $s = 1, 2, ..., n$. Data can be represented in terms of a matrix X where columns and rows of the matrix X correspond to samples and features respectively. The matrix X is given by

$$X = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ X_{p1} & X_{p2} & \cdots & X_{pn} \end{bmatrix}$$

We assume that the data matrix X is standardized so that the features have mean 0 and variance 1 across samples. Given a fixed feature, let Z_{ij} be the feature from the j^{th} sample of the i^{th} class where Z_{ij} is obtained from the corresponding row of X. We consider the following general model for Z_{ij} :

$$Z_{ij} = \mu_i + \varepsilon_{ij} \quad \text{for } i = 1, 2, \ldots\ldots, k \, ; \, j = 1, 2, \ldots, n_i$$

with $n_1 + n_2 + \cdots + n_k = n$. In the model, μ_i is a parameter representing the mean value of the feature in class i, ε_{ij} are the error terms such that ε_{ij} are independent normal random variables, with expectation (E) and variance(V) given by

$$E(\varepsilon_{ij}) = 0, \; V(\varepsilon_{ij}) = \sigma_i^2 < \infty$$

for $j = 1, 2, \ldots, n_i \quad i = 1, 2, \ldots, k$

We use the test statistics to determine the discriminating genes for microarray classification. Given a test statistic T, we define the discrimination power of a feature as the value of T evaluated at the n levels of the feature. This definition is based on the fact that with larger T the null hypothesis $H_0 : \mu_1 = \mu_2 = \cdots = \mu_k$ will be more likely to be rejected. Therefore, the higher the discrimination power, the more powerful the feature is in discriminating between different sample classes. Finally, we choose those genes as salient features which have high power of discrimination. For the case of homogeneity of variances, the well-known ANOVA F-test is the optimal test to accomplish the task [18]. However, with heterogeneity of the variances, the task is challenging. Therefore, some alternatives to the F-test are worthy of investigation. We considered the following parametric test statistics.

Brown- Forsythe Test Statistic(BF-Test) : This statistics is given by

$$B = \frac{\sum n_i (\bar{Z}_{i.} - \bar{Z}_{..})^2}{\sum (1 - n_i / n) s_i^2} \tag{1}$$

where $\bar{Z}_{i.} = \sum_{j=1}^{n_i} Z_{ij} / n_i$, $\bar{Z}_{..} = \sum_{i=1}^{k} n_i \bar{Z}_{i.} / n$, and $s_i^2 = \sum_{j=1}^{n_i} (Z_{ij} - \bar{Z}_{i.})^2 \big/ (n_i - 1)$.

Under H_0, B is distributed approximately as $F_{k-1,v}$ where

$$v = \frac{\left[\sum (1 - n_i / n) s_i^2 \right]^2}{\sum (1 - n_i / n)^2 s_i^4 / (n_i - 1)}$$

Adjusted Welch Test Statistic(AW-Test): It is a variant of the Welch test statistics[29]. Welch test statistics is given by

$$W = \frac{\sum w_i \left(\bar{Z}_{i.} - \sum h_i \bar{Z}_{i.} \right)^2}{(k-1) + 2(k-2)(k+1)^{-1} \sum (n_i - 1)^{-1} (1 - h_i)^2} \tag{2}$$

Where, $w_i = n_i \big/ s_i^2$ and $h_i = w_i \big/ \sum w_i$. Under H_0, W has an approximate distribution of F_{k-1, v_w} , where

$$v_w = \frac{k^2 - 1}{3\sum(n_i - 1)^{-1}(1 - h_i)^2}$$

The Adjusted Welch statistics [14] described on similar basis is defined by the formula

$$W^* = \frac{\sum w_i^*(\bar{Z}_{i.} - \sum h_i^* \bar{Z}_{i.})^2}{(k-1) + 2(k-2)(k+1)^{-1}\sum(n_i - 1)^{-1}(1 - h_i^*)^2} \tag{3}$$

where $w_i^* = n_i / (\phi_i s_i^2)$ with ϕ_i chosen such that $1 \le \phi_i \le (n_i - 1)/(n_i - 3)$ and $h_i^* = w_i^* / \sum w_i^*$. Under H_0, W^* has an approximate distribution of F_{k-1, v_w^*}, where

$$v_w^* = \frac{k^2 - 1}{3\sum(n_i - 1)^{-1}(1 - h_i^*)^2}$$

In this paper, we choose $\phi_i = (n_i + 2)/(n_i + 1)$, since this choice provides reliable results for small sample sizes n_i and a large number (k) of populations [14].

Cochran Test Statistics(C-Test): This is the quantity appearing as numerator of the Welch test statistics W and is given by

$$C = \sum w_i(\bar{Z}_{i.} - \sum h_i \bar{Z}_{i.})^2 \tag{4}$$

Under H$_0$, C has an approximate distribution of χ^2_{k-1}.

Pearson Correlation (PC): Pearson correlation coefficient [9,20] is used to measure linear dependency between two variables. For ranking of genes, Pearson correlation is calculated between each gene and class label. The Pearson correlation coefficient for a gene vector X_i and the class vector c is given by

$$PC(X_i) = \frac{n\sum X_{ij}c_j - (\sum X_{ij} \sum c_j)}{\sqrt{\left(\left(n\sum X_{ij}^2 - (\Sigma X_{ij})^2\right)\left(n\sum c_j^2 - (\sum c_j)^2\right)\right)}} \tag{5}$$

Its value ranges between -1 and +1. The closer the value is to -1 or 1, the stronger is the correlation among variables. As it approaches zero there is less correlation between the variables. It is capable of measuring both the degree as well as the direction of correlation. Also it is invariant to linear transformations of underlying variables.

However the assumption of linear relationship between the variables is not always true. Sometimes the value of correlation coefficient may mislead because a high value does not always imply a close relationship. Moreover it is sensitive to outliers.

Mutual Information (MI): It is another important ranking method based on information theoretic approach and measures dependency among variables. For ranking of genes, mutual information is calculated between each gene and class label. The mutual information for a gene vector X_i and the class vector c is given by

$$I(X_i,c)=\sum P(X_i,c)\log\frac{P(X_i,c)}{P(X_i)P(c)} \tag{6}$$

Where $P(X_i)$ and $P(c)$ are marginal probability distribution functions for random variables X_i and c respectively and $P(X_i,c)$ is joint probability distribution. For maximum information relevance, the selected features X_i should have largest mutual information $I(X_i,c)$ for target class c , which indicates the largest dependency on the target class. The advantage of mutual information is that it can capture even non linear relationship between the gene and the corresponding class label c.

Gini Index (GI): Gini Index is one of the ranking methods available in RankGene [28]. It was proposed by Breiman et. al [4] which is based on decision trees. It measures the probability of misclassification of a set of instances and is calculated using the following equation

$$GiniIndex=\left(\frac{n_l GiniL+n_r GiniR}{n}\right) \tag{7}$$

where GiniL is the Gini Index of left side of the hyperplane and GiniR is the right side . These two are given by:

$$GiniL=1.0-\sum_{i=1}^{k}\left(\frac{l_i}{n_l}\right)^2 \qquad GiniR=1.0-\sum_{i=1}^{k}\left(\frac{r_i}{n_r}\right)^2$$

Here n_l (n_r) is the number of values in the left (right) partition; l_i (r_i) is the number of values that belong to class i in the left (right) partition. The main advantage is that it is easily interpretable and applicable to large set of data. Its disadvantage is that being a relative measure, it does not keep information about absolute values.

3 Experimental Setup and Results

To determine the appropriate ranking method for selecting discriminatory genes for accurate classification of microarray datasets, we have investigated the above mentioned six popular ranking methods: BF-test, AW-test, C-test, Mutual Information (MI), Gini Index(GI), and Pearson Correlation(PC).

To evaluate the performance, we have used eleven publicly available microarray datasets. Brief description of datasets is given in Table 1. The datasets marked with an asterix are preprocessed as described in the original research work of their respective authors. For NCI dataset, one class contains only two samples, so this class has been removed from the dataset during experiment. Also, since number of samples belonging to each class is very small, 2000 genes with highest variance are considered for experiment. For preprocessing of remaining datasets (Table 2), two cut off values Floor and Ceil are used for each dataset and expression levels below Floor and above Ceil are set to Floor and Ceil respectively. Max/Min ratio and Max-Min difference of a gene across samples were used to filter genes with little variation across samples. Preprocessing is followed by normalization using z-score, so that feature values lie within similar ranges.

Table 1. Datasets Used

S. no.	Dataset	Samples	Original genes	Preprocessed genes	Classes
1.	Prostate [26]	102	12600	5966	2
2.	CNS-v1 [21]	34	7129	2277	2
3.	Colon [1]	62	2000	2000	2
4.	NCI 60 [23]	60	9706	2000	9
5.	CNS-v2 [21]	40	7129	5548	5
6.	Glioma [19]	50	12625	4434	4
7.	SRBCT [15]	83	2308	2308	4
8.	Melanoma [3]	38	8067	8067	3
9.	Leukemia [12]	72	7129	7129	3
10.	GCM [22] *	198	16063	11328	14
11.	CAR [28] *	174	12533	9182	11

Table 2. Preprocessed Data

Dataset name	Floor	Ceiling	Max/Min	Max-Min
Prostate	100	16000	5	50
CNS-v1	20	16000	5	500
CNS-v2	20	16000	5	500
Glioma	20	16000	3	100

K-Nearest Neighbor (KNN), Linear Discriminant Classifier (LDC) and Support Vector Machine (SVM) are used as classifiers which are commonly used by machine learning and data mining communities. The performance is evaluated in terms of classification accuracy and number of relevant genes. The training and test data of CAR and GCM are separately available. Hence, the classification accuracy of these datasets is reported using test data. The classification accuracy of remaining datasets is given in terms of Leave One Out Cross Validation(LOOCV). Each ranking method is applied to a dataset to obtain 60 top ranked genes. These top ranked genes are incrementally included one by one to develop the decision model. At every stage, classification accuracy of the test data is determined. Maximum classification accuracy obtained for each combination of ranking method and a classifier for a dataset are shown in Table 3. The number within parenthesis represents the number of genes corresponding to maximum classification accuracy for a given ranking method and a classifier. For each of the eleven datasets, combination of ranking and classification methods resulting in maximum and the minimum classification accuracy are shown in bold and red respectively in Table 3. The variation in classification accuracy with increase in number of genes as per ranking of the ranking method for only six combinations is depicted in Figure 1. Those combinations where the observations are more marked have been chosen.

The following can be observed from Figure 1 and Table 3:

1. For each dataset, there is significant variation in classification accuracy, which depends on the choice of ranking method and classifier.
2. There is no clear winner among six ranking method. However, for most of the datasets the best performance is achieved with BF, followed closely by MI.
3. The classification accuracy of Prostate, CNS-v1, CNS-v2 and CAR are almost similar for AW and C-test; The classification accuracy of Colon and GCM are almost similar for BF, AW and C-test.
4. BF, AW and C-test show stability in terms of performance *i.e* once a high accuracy is reached, it is maintained. Whereas, erratic spikes are observed in the case of PC and Gini Index.
5. Performance of MI is better with only a few numbers of genes, while other ranking methods show improvement as number of genes increases.
6. MI performs better than PC and Gini Index, and is often comparable with AW, BF and C-test.
7. Initially the performance of Gini is poor but improves with addition of genes beyond 40. However the performance is not better than other ranking methods.
8. The classification accuracy obtained with PC is poor in most of the cases. The performance of PC with LDC is the worst. This signifies relationship between a gene and a class is non-linear in general.
9. On a few datasets, for example Colon, Melanoma, Glioma and CNS-v1, change of classifier has a significant effect. The combination of AW/C with SVM is a better in comparison to AW/C with KNN/LDC.
10. CAR shows a very low minimum accuracy and GCM shows a low maximum accuracy.

Table 3. Classification accuracies of microarray datasets (Highest/lowest accuracy shown in bold/red)

Dataset	Classifier	MI	BF-test	AW-test	C-test	Gini	PC
Prostate	SVM	78.43(38)	**97.06(59)**	96.08(53)	96.08(53)	69.61(22)	71.57(52)
	KNN	72.55(22)	95.1(2)	95.1(58)	95.1(58)	75.49(52)	66.67(7)
	LDC	77.45(53)	95.1(5)	92.16(13)	92.16(13)	73.53(52)	65.69(7)
CNS(v1)	SVM	94.12(8)	94.12(30)	94.12(40)	94.12(40)	97.06(51)	85.29(23)
	KNN	**97.06(12)**	94.12(7)	94.12(18)	94.12(18)	94.12(60)	88.24(45)
	LDC	91.18(1)	94.12(34)	97.06(54)	97.06(54)	85.29(32)	79.41(23)
Colon	SVM	88.71(8)	88.71(5)	88.71(5)	88.71(5)	85.48(5)	90.32(39)
	KNN	**90.32(37)**	88.71(5)	88.71(5)	88.71(5)	85.48(30)	85.48(58)
	LDC	87.1(7)	85.48(13)	85.48(13)	85.48(13)	83.87(7)	79.03(30)
Nci60	SVM	81.03(25)	79.31(27)	72.41(41)	72.41(41)	81.03(56)	68.97(48)
	KNN	74.14(10)	75.86(31)	**82.76(54)**	**82.76(54)**	74.14(58)	68.96(32)
	LDC	63.79(9)	72.41(18)	62.07(7)	62.07(7)	63.79(59)	48.27(58)
CNS- v2	SVM	80(59)	82.5(55)	67.5(35)	67.5(35)	67.5(59)	62.5(54)
	KNN	67.5(30)	80(52)	70(40)	70(40)	60(52)	65(59)
	LDC	80(59)	**85(51)**	57.5(36)	57.5(36)	62.5(50)	67.5(43)
Glioma	SVM	74(27)	78(58)	78(47)	78(47)	68(58)	76(39)
	KNN	78(8)	**88(26)**	74(53)	74(53)	72(10)	70(46)
	LDC	72(8)	88(55)	64(57)	60(55)	76(59)	68(8)
SRBCT	SVM	**100(16)**	100(26)	100(49)	100(49)	90.36(52)	79.52(60)
	KNN	**100(16)**	100(21)	98.8(36)	98.8(36)	89.15(57)	66.27(47)
	LDC	100(24)	100(26)	97.6(19)	97.6(19)	85.54(37)	80.72(59)
Melanoma	SVM	76.32(59)	73.68(50)	**84.21(8)**	**84.21(8)**	60.53(29)	63.16(36)
	KNN	71.05(44)	71.05(13)	81.58(25)	81.58(25)	60.53(5)	50(2)
	LDC	57.89(1)	63.16(1)	63.16(8)	63.16(8)	57.89(1)	50(1)
Leukemia	SVM	68.05(60)	100(18)	98.61(47)	98.61(46)	86.11(60)	66.67(35)
	KNN	75(59)	**100(17)**	98.61(47)	98.61(51)	84.72(60)	68.06(23)
	LDC	63.88(31)	94.44(8)	94.44(17)	94.44(17)	72.22(38)	65.28(26)
GCM	SVM	40.74(32)	40.74(60)	40.74(60)	40.74(60)	44.44(38)	38.88(57)
	KNN	47.83(15)	39.13(60)	39.13(60)	39.13(60)	52.08(43)	36.96(53)
	LDC	47.83(15)	41.3(42)	41.3(42)	41.3(42)	**52.17(43)**	32.61(24)
CAR	SVM	77.03(45)	66.22(55)	17.57(1)	17.57(1)	36.48(4)	68.92(56)
	KNN	70.27(44)	58.11(56)	22.97(1)	22.97(1)	37.84(5)	59.45(50)
	LDC	**79.73(26)**	51.35(32)	21.62(2)	21.62(2)	31.08(6)	55.41(53)

To determine whether the feature ranking methods are significantly different or not, Friedman statistical test is conducted. Friedman test [8,10] is a non-parametric statistical test for hypothesis testing which is distributed according to χ_F^2 with k-1 degrees of freedom, where k is the number of samples. With six ranking algorithms and eleven datasets, each evaluated with three classifiers, the null hypothesis is rejected with α=0.05.

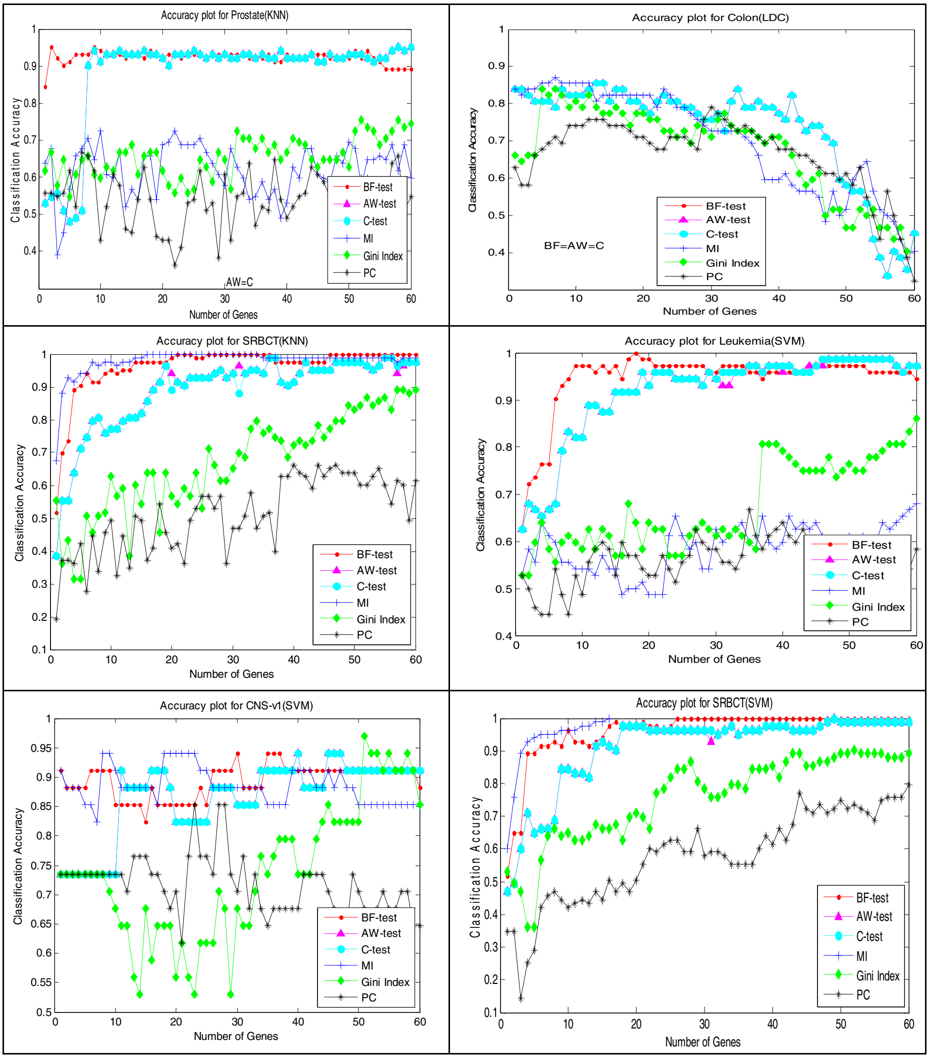


Fig. 1. Variation in classification accuracy with increase in number of genes as per ranking of each ranking algorithm (as per legend) for a few datasets (classifier wise)

4 Conclusion

Microarray genes are often characterized by high dimension and small sample size. Reduction in number of genes and identifying a minimal set of relevant genes in the datasets is an effective way of increasing the classification accuracy of the high dimensional datasets and helps in classifying the samples correctly. For this purpose, many gene ranking methods that are simple and involve only little computational complexity are suggested in literature to obtain a minimal set of relevant genes. In this paper, we have carried out extensive experiments to evaluate the performance of six popular ranking methods on 11 cancer microarray datasets. Our experiments have demonstrated that for each dataset there is significant variation in classification accuracy which depends on the choice of ranking method and classifier. The best performance is achieved with Brown and Forsythe statistic which is closely followed by Mutual Information method. The performance of Gini Index and Pearson Correlation is poor in comparison to other four ranking methods.

References

1. Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D., Levine, A.J.: Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Array. Proc. Nat'l Academy of Science 96(12), 6745–6750 (1999)
2. Bellman, R.: Adaptive Control Processes. In: A Guided Tour, Princeton University Press, Princeton (1961)
3. Bittner, M., Meltzer, P., Chen, Y., Jiang, Y., Seftor, E., Hendrix, M., Radmacher, M., Simon, R., Yakhini, Z., et al.: Molecular classification of cutaneous malignant melanoma by gene expression profiling. Nature 406(6795), 536–540 (2000)
4. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and regression trees. Chapman and Hall, Boca Raton (1984)
5. Brown, M.B., Forsythe, A.B.: The small sample behavior of some statistics which test the equality of several means. Technometrics 16, 129–132 (1974)
6. Cochran, W.: Problems arising in the analysis of a series of similar experiments. J. R. Stat. Soc. Ser. C Appl. Stat. 4, 102–118 (1937)
7. Dechang, C., Zhenqiu, L., Xiaobin, M., Dong, H.: Selecting Genes by Test Statistics. Journal of Biomedicine and Biotechnology 2, 132–138 (2005)
8. Demsar, J.: Statistical Comparisons of Classifiers over Multiple Data Sets. Journal of Machine Learning Research 7, 1–30 (2006)
9. Dowdy, S., Wearden, S.: Statistics for research. Wiley (1983)
10. Friedman, M.: A comparison of alternative tests of significance for the problem of m rankings. Annals of Mathematical Statistics 11, 86–92 (1940)
11. Fu, L.M., Liu, C.S.F.: Evaluation of gene importance in microarray data based upon probability of selection. BMC Bioinformatics 6, 67 (2005)
12. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286(5439), 531–537 (1999)
13. Guyon, I., Elisseff, A.: An Introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182 (2003)
14. Hartung, J., Argac¸, D., Makambi, K.: Small sample properties of tests on homogeneity in oneway ANOVA and meta-analysis. Statist Papers 43, 197–235 (2002)
15. Khan, J., Wei, J.S., Ringner, M., Saal, L.H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C.R., Peterson, C., Meltzer, P.S.: Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat. Med. 7(6), 673–679 (2001)
16. Kohavi, R., John, G.: Wrapper for feature subset selection. Artificial Intelligence 97(1-2), 273–324 (1997)

17. Li, T., Zhang, C., Ogihara, M.: Comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression. Bioinformatics 20, 2429–2437 (2004)
18. Neter, J., Kutner, M.H., Nachtsheim, C.J., et al.: Applied Linear Statistical Models, 4th edn. McGraw-Hill, Chicago (1996)
19. Nutt, C.L., Mani, D.R., Betensky, R.A., Tamayo, P., Cairncross, J.G., Ladd, C., Pohl, U., Hartmann, C., McLaughlin, M.E., Batchelor, T.T., Black, P.M., von Deimling, A., Pomeroy, S.L., Golub, T.R., Louis, D.N.: Gene expressionbased classification of malignant gliomas correlates better with survival than histological classification. Cancer Res. 63(7), 1602–1607 (2003)
20. Pearson, K.: Notes on the History of Correlation. Biometrika 13(1), 25–45 (1920)
21. Pomeroy, S.L., Tamayo, P., Gaasenbeek, M., Sturla, L.M., Angelo, M., McLaughlin, M.E., Kim, J.Y.H., et al.: Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature 415(6870), 436–442 (2002)
22. Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C.H., Angelo, M., Ladd, C., Reich, M., Latulippe, E., Mesirov, J.P., Poggio, T., Gerald, W., Loda, M., Lander, E.S., Golub, T.R.: Multiclass cancer diagnosis using tumor gene expression signatures. Proc. Natl. Acad. Sci. USA 98(26), 15149–15154 (2001)
23. Ross, D.T., Scherf, U., Eisen, M.B., Perou, C.M., Rees, C., Spellman, P., Iyer, V., Jeffrey, S.S., Rijn Van De, W.M., et al.: Systematic Variation in Gene Expression Patterns in Human Cancer Cell Lines. Nature Genet. 24, 227–235 (2000)
24. Shah, S., Kusiak, A.: Cancer gene search with data mining and genetic algorithms. Computers in Biology Medicine 37(2), 251–261 (2007)
25. Shannon, C.E., Weaver, W.: The mathematical theory of Communication. University of Illinois Press, Urbana (1949)
26. Singh, D., Febbo, P.G., Ross, K., Jackson, D.G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A.A., et al.: Gene expression correlates of clinical prostate cancer behavior. Cancer Cell 1(2), 203–209 (2002)
27. Su, A.I., Welsh, J.B., Sapinoso, L.M., Kern, S.G., Dimitrov, P., Lapp, H., Schultz, P.G., Powell, S.M., Moskaluk, C.A., Frierson, H.F., Hampton, G.M.: Molecular classification of human carcinomas by use of gene expression signatures. Cancer Res. 61(20), 7388–7393 (2001)
28. Su, Y., Murali, T.M., et al.: RankGene: identification of diagnostic genes based on expression data. Bionformatics 19(12), 1578–1579 (2003)
29. Welch, B.L.: On the comparison of several mean values: An alternative approach. Biometrika 38, 330–336 (1951)

Assessment of Bayesian Network Classifiers as Tools for Discriminating Breast Cancer Pre-diagnosis Based on Three Diagnostic Methods

Ameca-Alducin Maria Yaneli[1], Cruz-Ramírez Nicandro[2],
Mezura-Montes Efrén[1], Martín-Del-Campo-Mena Enrique[3],
Pérez-Castro Nancy[1], and Acosta-Mesa Héctor Gabriel[2]

[1] Laboratorio Nacional de Informática Avanzada (LANIA) A.C.
Rébsamen 80, Centro, Xalapa, Veracruz, 91000, México
[2] Departamento de Inteligencia Artificial, Universidad Veracruzana
Sebastián Camacho 5, Centro, Xalapa, Veracruz, 91000, México
[3] Centro Estatal de Cancerología: "Miguel Dorantes Mesa"
Aguascalientes 100, Progreso Macuiltepetl,Xalapa, Veracruz, 91130, México

Abstract. In recent years, a technique known as thermography has been again seriously considered as a complementary tool for the pre-diagnosis of breast cancer. In this paper, we explore the predictive value of thermographic atributes, from a database containing 98 cases of patients with suspicion of having breast cancer, using Bayesian networks. Each patient has corresponding results for different diagnostic tests: mammography, thermography and biopsy. Our results suggest that these atributes are not enough for producing good results in the pre-diagnosis of breast cancer. On the other hand, these models show unexpected interactions among the thermographical attributes, especially those directly related to the class variable.

Keywords: Thermography, Breast cancer, Bayesian networks.

1 Introduction

Nowadays, breast cancer is the first cause of death among women worldwide [1]. There are various techniques to pre-diagnose this disease such as auto-exploration, mammography, ultrasound, MRI and thermography [2,3,4,5]. The commonest test for carrying out this pre-diagnosis is mammography [2]; however, due to the different varieties of such disease [3], there are situations where this test does not provide an accurate result [6]. For instance, women younger than 40-years old have more density in their breast: this is an identified cause for mammography not to work properly [7]. In order to overcome this limitation in the pre-diagnosis of breast cancer, a relatively new technique has been proposed as a complement in such pre-diagnosis: thermography [5]. Such technique consists of taking infrared images of the breasts with an infrared camera [8]. Thermography

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 419–431, 2013.

represents a non-invasive, painless procedure, which does not expose the patient to x-ray radiation [7]. Besides, it is cheaper than other pre-diagnostic procedures. Thermography gives mainly information about temperature of the breasts and their corresponding differences. It is argued that lesions in the breasts produce significantly more temperature than healthy, normal breasts [9]. This is because these lesions (or tumors) contain more veins and a have metabolic rate than the surrounding tissue.

Our main contribution is the exploration of the predictive value of the atributes for three different diagnostic methods for breast cancer. With this exploration, we can more easily appreciate the performance of each method regarding accuracy, sensitivity and specificity. Moreover, we can visually identify which thermographic variables are considered, from the point of view of a Bayesian network, more important to predict the outcome.

The rest of the paper is organized as follows. Section 2 describes the state of the art that gives the proper context so that our contribution is more easily identified. Section 3 presents the materials and methods used in our experiments. Section 4 gives the methodology to carry out such experiments and the respective results. Section 5 discusses these results and, finally, section 6 gives the conclusions and identifies some future work.

2 State of the Art

The state of the art of thermography includes introductory investigations, image-based works and data-based works [10,11]. The first ones focus on the explanation of the technique as well as its advantages and disadvantages. A representative work is that of Foster (1998) [6], who points out that thermography may be a potential alternative diagnostic method since it does not produce x-ray radiation. The second ones concentrate on techniques for image processing such as clustering or fractal analyses[12,13]. The work of EtehadTavakol et al. (2008) [12] uses k-means and fuzzy c-means for separating abnormal breast from normal breast. The final ones present statistical and Artificial Intelligence techniques (such as Artificial Neural Networks) [14,15,7,16]. The work of Wishart et al. (2010) [16] performs the comparison between two software that uses AI techniques for analyzing data coming from thermographic images so that diagnoses can be carried out.

Our work is situated in the data-based works and focuses on the exploration of the discriminative power of thermographic atributes for the pre-diagnosis of breast cancer using Bayesian networks.

3 Materials and Methods

3.1 The Database

We used a 98-case database which was provided by an oncologist who specializes in the study of thermography since 2008. The database consists of 77 sick patients and 21 healthy patients. Each of the patients (either sick or healthy) has tests

Table 1. Names, definitions and types of variables of thermography

Variable name	Definition	Variable type
Asymmetry	Degree difference (in Celsius) between the right and the left breasts	Nominal (range [1-3])
Thermovascular network	Amount of veins with highest temperature	Nominal (range [1-3])
Curve pattern	Heat area under the breast	Nominal (range [1-3])
Hyperthermia	Hottest point of the breast	Binary
2c	Degree difference between the hottest points of the two breasts	Nominal (range [1-4])
F unique	Amount of hottest points	Nominal (range [1-4])
1c	Hottest point in only one breast	Binary
Furrow	Furrows under the breasts	Binary
Pinpoint	Veins going to the hottest points of the breasts	Binary
Hot center	The center of the hottest area	Binary
Irregular form	Geometry of the hot center	Binary
Histogram	Histogram in form of a isosceles triangle	Binary
Armpit	Difference degree between the 2 armpits	Binary
Breast profile	Visually altered profile	Binary
Score	The sum of values of the previous 14 variables	Binary
Age	Age of patient	Nominal (range [1-3])
Outcome	Cancer/no cancer	Binary

Table 2. Names, definitions and types of variables of mammography

Variable name	Definition	Variable type
BIRADS	Assigned value in a mammography to measure the degree of the lesion	Nominal (range [0-6])
Clockwise	Clockwise location of the lesion	Nominal (range [1-12])
Visible tumor	Whether the tumor is visible in the mammography	Binary
spiculated edges	Whether the edges of the lesion are spiculated	Binary
Irregular edges	Whether the edges of the lesion are irregular	Binary
microcalcifications	Whether microcalcifications are visible un the mammography	Binary
Asymmetry$_M$	Whether the breast tissue is asymmetric	
distortion	Whether the structure of the breast is distorted	Binary

Table 3. Names, definitions and types of variables of biopsy

Variable name	Definition	Variable type
sizeD	Discretized tumor size	Nominal (range [1-3])
RHP	Types of cancer	Nominal (range [1-8])
SBRdegree	degree of cancer malignancy	Nominal (range [0-3])

for thermography, mammography and biopsy. 28 variables in total form this dataset: 16 belong to thermography, 8 belong to mammography and 3 to biopsy; the last variable taken into account is outcome (cancer and no cancer). This last variable is confirmed by an open biopsy, which is considered as the gold-standard test for diagnosing breast cancer. Table 1 presents the names and a brief description of the corresponding thermographic variables. Table 2 presents the same information for mammographic variables while Table 3 for biopsy variables.

3.2 Bayesian Networks

A Bayesian network (BN) [17,18] is a graphical model that represents relationships of probabilistic nature among variables of interest. Such networks consist of a qualitative part (structural model), which provides a visual representation of the interactions amid variables, and a quantitative part (set of local probability distributions), which permits probabilistic inference and numerically measures the impact of a variable or sets of variables on others. Both the qualitative and quantitative parts determine a unique joint probability distribution over the variables in a specific problem [17,18,19].

Figures 1 and 2 (see section 4) show examples of a BN. One of the great advantages of this model is that it allows the representation of a joint probability distribution in a compact and economical way by making extensive use of conditional independence, as shown in equation 1:

$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i|Pa(X_i)) \tag{1}$$

Where $Pa(X_i)$ represents the set of parent nodes of X_i; i.e., nodes with arcs pointing to X_i. Equation 1 also shows how to recover a joint probability from a product of local conditional probability distributions.

Bayesian Network Classifiers. Classification refers to the task of assigning class labels to unlabeled instances. In such a task, given a set of unlabeled cases on the one hand, and a set of labels on the other, the problem to solve is to find a function that suitably maps each unlabeled instance to its corresponding label (class). As can be inferred, the central research interest in this specific area is the design of automatic classifiers that can estimate this function from data (in our case, we are using Bayesian networks). This kind of learning is known as supervised learning [20,21,22]. The procedures used in these tests are: a) the Naïve Bayes classifier, b) Hill-Climber and c) Repeated Hill-Climber [23,24,21].

a) The Naïve Bayes classifier's (NB) main appeals are simplicity and accuracy: although its structure is always fixed (the class variable has an arc pointing to every attribute). In simple terms, the NB learns for maximum likelihood, from a training data sample, the conditional probability of each attribute given the class. Then, once a new case arrives, the NB uses Bayes' rule to compute the conditional probability of the class given the set of attributes selecting the value of the class with the highest posterior probability.
b) Hill-Climber is a Weka's [23] implementation of a search and scoring algorithm, which uses greedy-hill climbing [25] for the search part and different metrics for the scoring part, such as BIC (Bayesian Information Criterion), BD (Bayesian Dirichlet), AIC (Akaike's Information Criterion) and MDL (Minimum Description Length). For the experiments reported here, we selected the MDL metric. This procedure takes as input an empty graph and a database and applies different operators for building a Bayesian network: addition, deletion or reversal of an arc. In every search step, it looks for a structure that minimizes the MDL score. In every step, the MDL is calculated and procedure Hill-Climber keeps the structure with the best (minimum) score. It finishes searching when no new structure improves the MDL score of the previous network.
c) Repeated Hill-Climber is a Weka's [23] implementation of a search and scoring algorithm, which uses repeated runs of greedy-hill climbing [25] for the search part and different metrics for the scoring part, such as BIC, BD, AIC and

MDL. For the experiments reported here, we selected the MDL metric. In contrast to the simple Hill-Climber algorithm, repeated Hill-Climber takes as input a randomly generated graph. It also takes a database and applies different operators (addition, deletion or reversal of an arc) and returns the best structure of the repeated runs of the Hill-Climber procedure. With this repetition of runs, it is possible to reduce the problem of getting stuck in a local minimum [19].

3.3 Evaluation Method: Stratified K-Fold Cross-Validation

We follow the definition of the cross-validation method given by Kohavi [22]. In k-fold cross-validation, we split the database D in k mutually exclusive random samples called the folds: $D_1, D_2, \ldots, D_k$, where such folds have approximately equal size. We train this classifier each time $i \in 1, 2, \ldots, k$ using $D \setminus D_i$ and test it on Di, which means the whole date minus the corresponding fold. The cross-validation accuracy estimation is the total number of correct classification divided by the sample size (total number of instances in D). Thus, the k-fold cross validation estimate is:

$$acc_{cv} = \frac{1}{n} \sum_{(v_i, y_i) \in D} \delta(I(D \setminus D_{(i)}, v_i), y_i) \tag{2}$$

Where $(I(D \setminus D_{(i)}, v_i), y_i)$ denotes the label assigned by classifier I to an unlabeled instance v_i on dataset $D \setminus D_{(i)}$, y_i is the class of instance v_i, n is the size of the complete dataset and $\delta(i, j)$ is a function where $\delta(i, j) = 1$ if $i = j$ and 0 if $i \neq j$. In other words, if the label assigned by the inducer to the unlabeled instance v_i coincides with class y_i, then the result is 1; otherwise, the result is 0; i.e., we consider a 0/1 loss function in our calculations of equation 2. It is important to mention that in stratified k-fold cross-validation, the folds approximately contain (roughly) the same proportion of classes as in the complete dataset D. For differents classifier, we assess the performance of the classifiers presented in section 3.2 using the following measures [26,27,28,29]:

a) *Accuracy*: the overall number of correct classifications divided by the size of the corresponding test set.
b) *Sensitivity*: the ability to correctly identify those patients who actually have the disease.
c) *Specificity*: the ability to correctly identify those patients who do not have the disease

4 Methodology and Experimental Results

The procedure for making the thermographic study begins with the collection of thermal images. These images are taken from 1 meter away of the patient, depending on her muscular mass in a temperature-controlled room (18-22^o C), with a FLIR A40 infrared camera. Three images are taken for each patient: one frontal and two laterals (left and right). Right after, the breasts are uniformly covered with surgical spirit (using a cotton). Two minutes after, the same three images are taken again. All these images are stored using the ThermaCAM Researcher Professional 2.9 software. Once the images are taken and stored, the specialist analyzes them and fills in the database with the corresponding values for each thermographic variable. He also includes the corresponding values for mammographic and biopsy variables.

We carried out the experiments using the database described in section 3.1 and the three different Bayesian network classifiers presented in section 3.2, which are implemented in Weka [23] (see their parameter set in table 4). For comparison purposes other classifiers were included: Artificial Neural Network and decision trees (ID3 and C4.5) with default parameters. For measuring their accuracy, sensitivity and specificity, we used 10-fold cross-validation as described in section 3.3. The main objective of these experiments is to explore the diagnostic performance of the atributes of thermography, mammography and biopsy, including the unveiling of the interactions among attributes and class.

Table 5 shows the numerical results of thermography, mammography and biopsy alone and thermography and mammography combined for Naïve Bayes, Hill-Climber and Repeated Hill-Climber. Table 6 shows the same results for the Artificial Neural Network and decision trees. Figures 2- 7 show the BN corresponding to Hill-Climber and Repeated Hill-Climber for thermography, mammography and biopsy respectively.

Table 4. Used the following values for Hill-Climber and Repeated Hill-Climber

Parameters	Hill-Climber	Repeated Hill-Climber
The initial structure NB (Naïve Bayes)	False	False
Number of parents	100,000	100,000
Runs	-	10
Score type	MDL	MDL
Seed	-	1
arc reversal	True	True

Table 5. Accuracy, sensitivity and specificity of Naïve Bayes, Hill-Climber and Repeated Hill-Climber for different methods of pre-diagnosis of breast cancer. For the accuracy test, the standard deviation is shown next to the accuracy result. For the remaining tests, their respective 95% confidence intervals (CI) are shown in parentheses.

Method		**Naïve Bayes**			**Hill-Climber**			**Repeated Hill-Climber**	
	Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity
Thermography	68.18% (±12.15)	79%(70-88)	24%(6-42)	78.56% (±3.14)	100%(100-100)	0%(0-0)	78.56% (±3.14)	100%(100-100)	0%(0-0)
Mammography	80.67% (±10.95)	87%(0-0)	52.4%(0-0)	72.56% (±10.30)	84%(76-93)	71%(52-91)	74.56% (±8.26)	87%(80-95)	71%(52-91)
biopsy	99% (±3.16)	99%(80-95)	100%(100-100)	99% (±3.16)	84%(76-93)	100%(100-100)	99% (±3.16)	87%(80-95)	100%(100-100)
Thermography and mammography	77.22% (±14.13)	84%(76-93)	52%(31-74)	69.44% (±10.37)	83%(75-91)	19%(2-36)	70.56% (±11.85)	84%(76-93)	19%(2-36)

Table 6. Accuracy, sensitivity and specificity of Artificial Neural Network, decision tree ID3 and C4.5 for different methods of pre-diagnosis of breast cancer. For the accuracy test, the standard deviation is shown next to the accuracy result. For the remaining tests, their respective 95% confidence intervals (CI) are shown in parentheses.

Method		**Artificial Neural Network**			**Decision Tree ID3**			**Decision Tree C4.5**	
	Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity
Thermography	70.19% (±11.43)	78%(69-87)	33%(13-53)	74.87% (±12.15)	89%(82-96)	52%(31-74)	75.58% (±6.82)	94%(88-99)	5%(-4-14)
Mammography	85.30% (±9.55)	92%(86-98)	67%(47-87)	*73.79% (±12.79)	94%(88-100)	61%(39-84)	*77.96% (±4.36)	97%(94-101)	0%(0-0)
biopsy	100% (±0)	100%(100-100)	100%(100-100)	97.33% (±5.75)	100%(100-100)	100%(100-100)	100%(±0)	100%(100-100)	100%(100-100)
Thermography and mammography	76.11% (±12.91)	87%(80-95)	48%(26-69)	68.67% (±12.56)	76%(67-86)	41%(18-65)	74.36% (±8.70)	94%(88-99)	5%(-4-14)

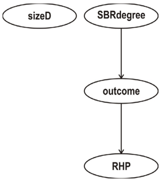


Fig. 1. Bayesian network resulting from running Hill-Climber and Repeated Hill-Climber with the biopsy 98-case database

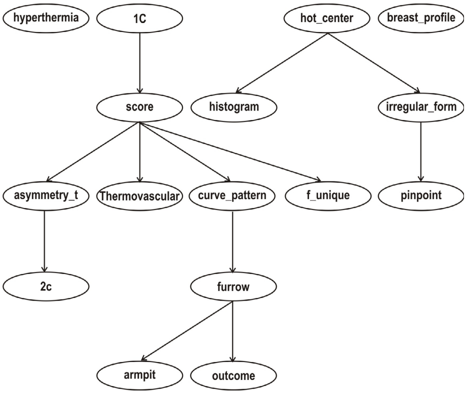


Fig. 2. Bayesian network resulting from running Hill-Climber with the thermographic 98-case database

5 Discussion

Our main objective was explore the predictive value of thermographic atributes for pre-diagnosis of breast cancer. We decided to use the framework of Bayesian networks because its power to visually unveil the relationships not only among

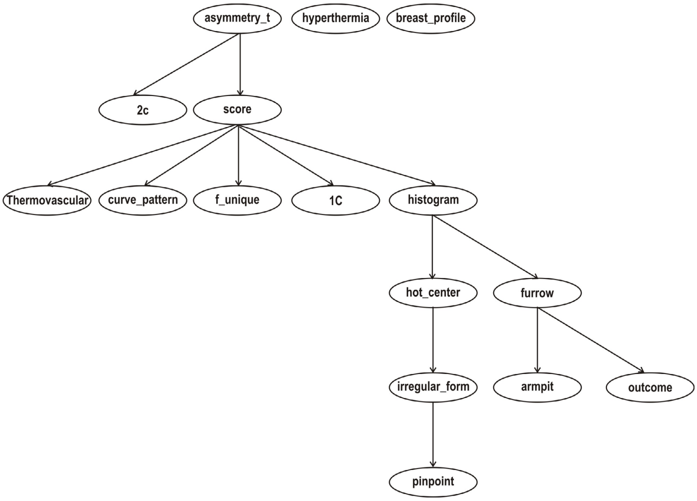


Fig. 3. Bayesian network resulting from running Repeated Hill-Climber with the thermographic 98-case database

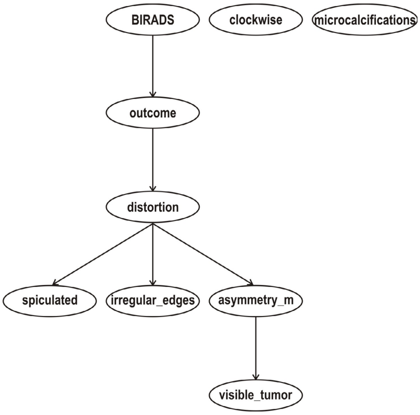


Fig. 4. Bayesian network resulting from running Hill-Climber with the mammography 98-case database

the attributes themselves but also between the attributes and the class. Furthermore, this model allows one to represent the uncertainty usually contained in the medical domain. First of all, let us check the accuracy, sensitivity and specificity performance of the Bayesian network classifiers using for the thermographic atributes (see Table 5). The results of Bayesian networks for thermography and mammography are almost comparable, but for the neural network (see table 6) mammography has an accuracy of 85.30%. In fact, thermography is excellent in identifying cases with the disease (100% sensitivity) but it performs very poorly for detecting healthy cases (0% specificity) for both Hill-Climber and Repeated

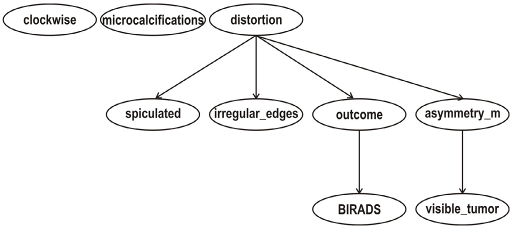


Fig. 5. Bayesian network resulting from running Repeated Hill-Climber with the mammography 98-case database

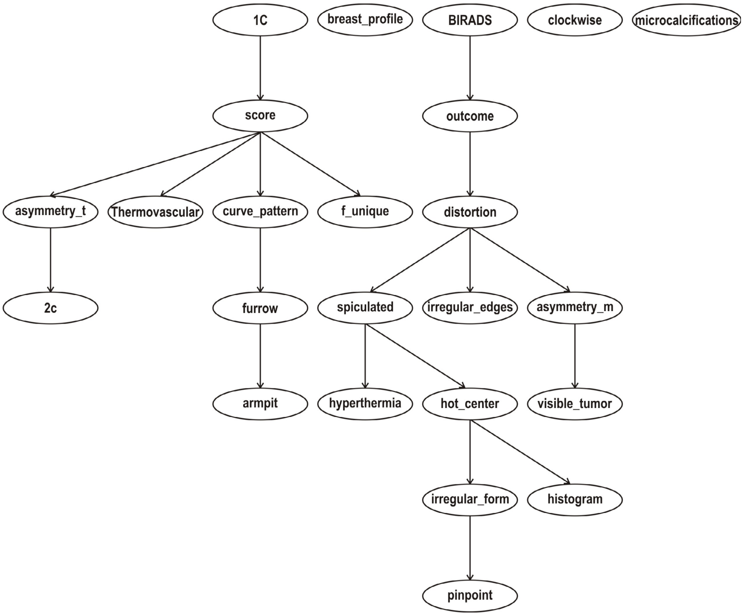


Fig. 6. Bayesian network resulting from running Hill-Climber with the thermographic and mammography 98-case database

Hill-Climber classifiers. And if we compare the results of the table 6 for Neural network, ID3 and C4.5, Bayesian classifiers (78.56%) obtained increased accuracy, as seen in the table 5.

It is remarkable the change in the performance of these two pre-diagnosis techniques with the inclusion of all their respective variables (Naïve Bayes classifier): 24% for thermography and 52.4% for mammography. It seems that this inclusion, far from improving the performance, makes it worse. Coming back to sensitivity and specificity values, it can be argued that in a certain sense, thermography can indeed be useful as a complementary tool for the pre-diagnosis of breast cancer. To see the picture more clearly, imagine a patient with a mam-

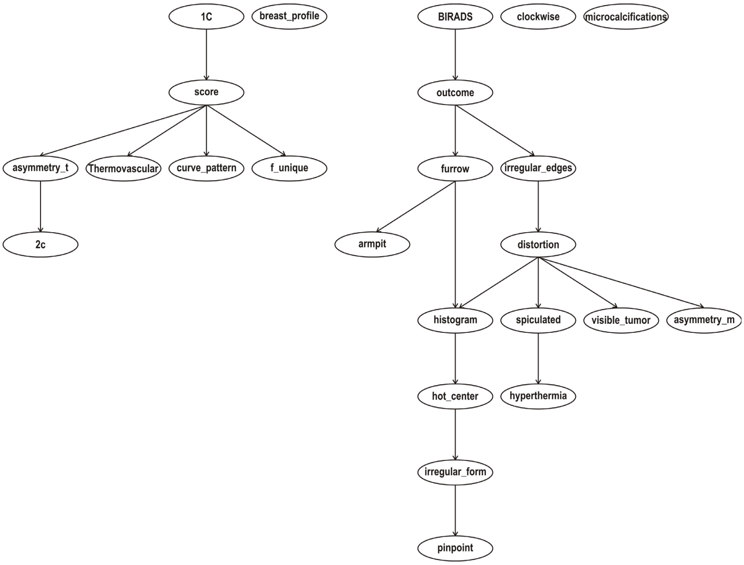


Fig. 7. Bayesian network resulting from running Repeated Hill-Climber with the thermographic and mammography 98-case database

mographic positive result for cancer. In order to be more certain of this result, a thermography can be taken to confirm it since thermography seems to identify without much trouble a sick patient.

The biopsy (see results in the tables 5, 6) is indeed the gold-standard method to diagnose breast cancer. One question that immediately pops up is: why not then always use biopsy to diagnose breast cancer? The answer is because it implies a surgical procedure that involves known risks such as those that have mainly to do with anesthesia apart from the economical costs. Methods such as mammography try to minimize the number of patients undertaking surgery. In other words, if all non-surgical procedures fail in the diagnosis, biopsy is the ultimate resource.

Regarding the unveiling of the relationships among attributes and among the attributes and the outcome, we can detect various interesting issues. For the case of thermography, contrary to what was expected, there is only one variable directly responsible for the explaining the behavior of the outcome: variable furrow (figures 2 and 3). It seems that this variable is enough for obtain the maximum percentage of classification (78.56%) for identifying patients with cancer. For the case of mammography, variables distortion and BIRADS are the only responsible to detect abnormal cases as well as normal cases (figures 4 and 3). This may mean that radiologists could just observe these two variables to diagnose the presence/absence of the disease. For the case of biopsy, we have more arguments to trust in this technique, in spite of its related and well-known risks. Finally, Table 5 suggests not to analyze the thermographic and mammographic variables together but separated: the former decreases accuracy, sensitivity and specificity performance.

6 Conclusions and Future Work

Our results suggest that thermography may have potential as a tool for prediagnosing breast cancer. However, more study and tests are needed. Its overall accuracy and sensitivity values are encouraging; however, on the other hand, its specificity values are disappointing. The Bayesian networks, resultant from running 3 algorithms on the thermography database, give us a good clue for this behavior: it seems that most of thermographic variables are rather subjective, making it difficult to avoid the usual noise in this kind of variables. The present study allows then to think revisiting these variables and the way they are being measured. Such subjectivity does not belong only to thermography but also to mammography: according to the Bayesian network results, just two variables are responsible for explaining the outcome. Indeed, as can be noted from these results, when all mammographic variables are included (Naïve Bayes classifier) the specificity values drop significantly with respect to those when only a subset of such attributes is considered. Moreover, if there were no subjectivity regarding specificity, then Naïve Bayes would perform better. Thus, it seems that there exists an overspecialization of the expert radiologist in the sense of considering all variables for diagnosing patients with the disease but an underspecialization for diagnosing the absence of such a disease. It is important to mention that our database is unbalanced: we need to get more data (healthy and sick cases) so that our conclusions are more certain.

For future work, we firstly propose to add more cases to our databases. Secondly, it would be desirable to have roughly the same number of positive and negative cases. Finally, we recommend a revision on how thermographic variables are being measured.

Acknowledgements. The first, third, and fifth authors acknowledge support from CONACyT through project No. 79809. We are also very grateful to the fourth author, who provided the database.

References

1. Jemal, A., Bray, F., Center, M., Ferlay, J., Ward, E., Forman, D.: Global cancer statistics. CA: A Cancer Journal for Clinicians 61, 69–90 (2011)
2. Geller, B.M., Kerlikowske, K.C., Carney, P.A., Abraham, L.A., Yankaskas, B.C., Taplin, S.H., Ballard-Barbash, R., Dignan, M.B., Rosenberg, R., Urban, N., Barlow, W.E.: Mammography surveillance following breast cancer. Breast Cancer Research and Treatment 81, 107–115 (2003)
3. Bonnema, J., Van Geel, A.N., Van Ooijen, B., Mali, S.P.M., Tjiam, S.L., Henzen-Logmans, S.C., Schmitz, P.I.M., Wiggers, T.: Ultrasound-guided aspiration biopsy for detection of nonpalpable axillary node metastases in breast cancer patients: New diagnostic method. World Journal of Surgery 21, 270–274 (1997)

4. Schnall, M.D., Blume, J., Bluemke, D.A., DeAngelis, G.A., DeBruhl, N., Harms, S., Heywang-Köbrunner, S.H., Hylton, N., Kuhl, C., Pisano, E.D., Causer, P., Schnitt, S.J., Smazal, S.F., Stelling, C.B., Lehman, C., Weatherall, P.T., Gatsonis, C.A.: Mri detection of distinct incidental cancer in women with primary breast cancer studied in ibmc 6883. Journal of Surgical Oncology 92, 32–38 (2005)
5. Ng, E.Y.K.: A review of thermography as promising non-invasive detection modality for breast tumor. International Journal of Thermal Sciences 48, 849–859 (2009)
6. Foster, K.R.: Thermographic detection of breast cancer. IEEE Engineering in Medicine and Biology Magazine 17, 10–14 (1998)
7. Arora, N., Martins, D., Ruggerio, D., Tousimis, E., Swistel, A.J., Osborne, M.P., Simmons, R.M.: Effectiveness of a noninvasive digital infrared thermal imaging system in the detection of breast cancer. The American Journal of Surgery 196, 523–526 (2008)
8. Hairong, Q., Phani, T.K., Zhongqi, L.: Early detection of breast cancer using thermal texture maps. In: Proceedings. 2002 IEEE International Symposium on Biomedical Imaging, pp. 309–312 (2002)
9. Wang, J., Chang, K.J., Chen, C.Y., Chien, K.L., Tsai, Y.S., Wu, Y.M., Teng, Y.C., Shih, T.T.: Evaluation of the diagnostic performance of infrared imaging of the breast: a preliminary study. BioMedical Engineering OnLine 9, 1–14 (2010)
10. Gutierrez, F., Vazquez, J., Venegas, L., Terrazas, S., Marcial, S., Guzman, C., Perez, J., Saldana, M.: Feasibility of thermal infrared imaging screening for breast cancer in rural communities of southern mexico: The experience of the centro de estudios y prevencion del cancer (ceprec). In: 2009 ASCO Annual Meeting, p. 1521. American Society of Clinical Oncology (2009)
11. Ng, E.Y.K., Chen, Y., Ung, L.N.: Computerized breast thermography: study of image segmentation and tempe rature cyclic variations. Journal of Medical Engineering & Technology 25, 12–16 (2001)
12. EtehadTavakol, M., Sadri, S., Ng, E.Y.K.: Application of k- and fuzzy c-means for color segmentation of thermal infrared breast images. Journal of Medical Systems 34, 35–42 (2010)
13. EtehadTavakol, M., Lucas, C., Sadri, S., Ng, E.Y.K.: Analysis of breast thermography using fractal dimension to establish possible difference between malignant and benign patterns. Journal of Healthcare Engineering 1, 27–44 (2010)
14. Ng, E.Y.K., Fok, S.-C., Peh, Y.C., Ng, F.C., Sim, L.S.J.: Computerized detection of breast cancer with artificial intelligence and thermograms. Journal of Medical Engineering & Technology 26, 152–157 (2002)
15. Ng, E.Y.K., Fok, S.-C.: A framework for early discovery of breast tumor using thermography with artificial neural network. The Breast Journal 9, 341–343 (2003)
16. Wishart, G.C., Campisi, M., Boswell, M., Chapman, D., Shackleton, V., Iddles, S., Hallett, A., Britton, P.D.: The accuracy of digital infrared imaging for breast cancer detection in women undergoing breast biopsy. European Journal of Surgical Oncology (EJSO) 36, 535–540 (2010)
17. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann series in representation and reasoning. Morgan Kaufmann Publishers (1988)
18. Neuberg, L.G.: Causality: Models, reasoning, and inference, by judea pearl. Econometric Theory 19, 675–685 (2003)
19. Friedman, N., Goldszmidt, M.: Learning bayesian networks from data. University of California, Berkeley and Stanford Research Institute, pp. 117 (1998)
20. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. TheKaufmann Series in Data Management Systems. Elsevier (2006)

21. Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian network classifiers. Machine Learning 29, 131–163 (1997)
22. Kohavi, R.: A study of cross-validation and bootstrap for accuracy estimation and model selection, pp. 1137–1143. Morgan Kaufmann (1995)
23. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann Series in Data Management Sys. Morgan Kaufmann (2005)
24. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. John Wiley & Sons (2001)
25. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach, 3rd edn. Prentice Hall (2009)
26. Lavrac, N.: Selected techniques for data mining in medicine. Artificial Intelligence in Medicine 16, 3–23 (1999)
27. Cross, S.S., Dubé, A.K., Johnson, J.S., McCulloch, T.A., Quincey, C., Harrison, R.F., Ma, Z.: Evaluation of a statistically derived decision tree for the cytodiagnosis of fine needle aspirates of the breast (fnab). Cytopathology 9, 178–187 (1998)
28. Cross, S.S., Stephenson, T.J., Harrisont, R.F.: Validation of a decision support system for the cytodiagnosis of fine needle aspirates of the breast using a prospectively collected dataset from multiple observers in a working clinical environment. Cytopathology 11, 503–512 (2000)
29. Cross, S.S., Downs, J., Drezet, P., Ma, Z., Harrison, R.F.: Which decision support technologies are appropriate for the cytodiagnosis of breast cancer?, pp. 265–295. World Scientific (2000)

Identification of Risk Factors for TRALI Using a Hybrid Algorithm

María Dolores Torres [1], Aurora Torres[1], Felipe Cuellar[2], María de la Luz Torres [2], Eunice Ponce de León[1], and Francisco Pinales[1]

[1]Universidad Autónoma de Aguascalientes. Av. Universidad # 940, Ciudad Universitaria.
C.P. 20131, Aguascalientes, Ags. México
{mdtorres,atorres,eponce,fjpinales}@correo.uaa.mx
[2] Centenario Hospital Miguel Hidalgo. Galeana Sur # 465, Col. Obraje. C.P. 20230,
Aguascalientes, Ags. México
{cuellarluis,tishats}@hotmail.com

Abstract. This paper presents a hybrid evolutionary algorithm to identify risk factors associated with transfusion related acute lung injury (TRALI). This medical condition occurs mainly in intensive care units and operating rooms, and the main strategy for its treatment is prevention. The proposed algorithm works with information from the model known as "two hits", in which the first hit is the original disease and the second corresponds to the blood transfusion. This algorithm is based on a genetic algorithm hybridized with testor analysis. This research used information from 87 patients treated at the Centenary Hospital Miguel Hidalgo in the city of Aguascalientes, Mexico. As a result of the algorithm's application, it was found that most variables are related to the first hit, while only some of them belong to the second one. The analysis also revealed that some variables physicians believed significant a priori, were not very important; among other discoveries.

Keywords: Genetic Algorithm, Evolutionary Hybrid Algorithm, TRALI's risk factors, Testor, Typical testor.

1 Introduction

Computer sciences and especially techniques from artificial intelligence have always shown their effectiveness and importance in the improving human life conditions. Initially, computation supported the processing of huge quantities of information[1] and subsequently, it was applied in all branches of knowledge as a support tool. Medicine has not been the exception; since the development of the first expert systems, was evident that this area has always been of high interest to the computer sciences; Mycin[2] is an example of this interest [20].

[1] The reader will remain that the national census of the United States of America, 1880 took 7 years for the information to be processed, while the 1890 census, that originally was estimated to take between 10 and 12 years, took a record time of 2 ½ years through the use of punched cards.

[2] Mycin was an early expert system designed to identify bacteria causing severe infections, and to recommend antibiotics. This system was also used for the diagnosis of blood clotting diseases.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 432–445, 2013.

This paper describes the application of a hybrid algorithm in the determination of risk factors for transfusion related acute lung injury, and it reveals several interesting results, according to expert physicians. The used approach is based on a genetic algorithm hybridized with different mechanisms, including the accelerating operator, the improvement mechanisms, and the fitness evaluation function, which depends on the status of testor or typical testor of an individual.

The approached problem has a huge solution space, since it is initially described by 31 variables. This is traduced into 2,147,483,647 possible subsets of variables; consequently the use of powerful computational tools is completely justified.

To better understand the application presented in this paper, below in this section it is included a description of this syndrome, followed by a brief explanation of the main components of the hybridized algorithm. Next sections have been organized in the following way; section 2 presents the used methodology. The core of section 3 is the description of the hybrid evolutionary algorithm. Section 4, describes results and discussion obtained from the application of the algorithm, and finally, conclusions and future work are presented in section 5.

1.1 Transfusion Related Acute Lung Injury (TRALI)

This paper addresses the identification of risk factors for TRALI with preventive purposes. This syndrome is characterized by the development of acute respiratory failure and pulmonary edema within 6 hours after transfusion, with a wide clinical spectrum of presentation. The term "transfusion-related acute lung injury" was coined in 1985 by Popovsky et al., [14].

According to Añon et al., although this is an under-diagnosed medical condition, this syndrome is considered the leading cause of transfusion-related death in the United States, and the second one in the United Kingdom [1]. Its low level of diagnosis, combined with its seriousness, makes it a syndrome that should be avoided, as Cuellar stated [6].

Since the initial description in 1951 [3], non-cardiogenic lung edema related to transfusion has been reported widely using different names, including non-cardiogenic pulmonary edema, pulmonary hypersensitivity and severe allergic pulmonary edema.

Today, this disease has gone from being an almost unknown side effect of transfusion, to the leading cause of death associated with transfusion.

The formal definition of this medical condition was developed by the National Heart Lung and Blood Institute Working Group on TRALI U.S. in 2003 [24], and is described as an acute lung injury (ALI) temporally related to a blood transfusion, specifically, it occurs within the first six hours following a transfusion.

Some of the most important elements of the TRALI´s definition according to medical experts are [10]:

1. Sudden onset of acute lung injury (ALI)
2. Hypoxemia (PaO2/FiO2 (ratio of partial pressure of arterial O2 to the fraction of inspired O2) ≤300 and must be adjusted downward with increasing altitude or SpO2≤90% on room air or other clinical evidence)

3. Bilateral lung infiltrates on frontal chest radiograph
4. No evidence of left atrial hypertension (i.e., transfusion-associated circulatory overload)
5. Occurrence during or within 6 h after completion of transfusion
6. No temporal relationship to an alternative risk factor for ALI
7. New ALI and no other ALI risk factors present including aspiration, multiple trauma, pneumonia, cardiopulmonary bypass, burn injury, toxic inhalation, lung contusion, acute pancreatitis, drug overdose, near drowning, shock and sepsis
8. If one or more ALI risk factors are present, possible TRALI should be diagnosed (in patients with an alternative ALI risk factor, TRALI is still possible).

There are two proposed etiologies for this syndrome: the first is an immune mediated episode produced by transfused antibodies directed toward Human Leukocyte Antigens (HLA) or Human Neutrophil Antigens (HNA). The second, is a model consisted of two events: the first of them is related to the clinical picture of the patient (first hit), and the second one is the transfusion of blood products (second hit). This model commonly known as "two hits" was taken as the basis for the classification of the studied variables. Later the reader will see that we refer to variables such as those related with the first hit, those related to the second hit and other variables. TRALI is frequently associated with the transfusion of plasma products, but can also occur in recipients of packed red blood cells due to the residual plasma present in the unit. According to Palfi et at., transfusion of blood products obtained from multiparous women was associated with a significantly higher incidence of hypotension, decreased PaO2 relationship / FIO2 and proinflammatory cytokines [13]. Therefore, transfusion of blood components obtained from these donors is a higher risk of inducing immune-mediated TRALI [12]. Also previous transfusion or transplantation can lead to donor sensitization.

Several studies have explored the prevention of TRALI using three main strategies:

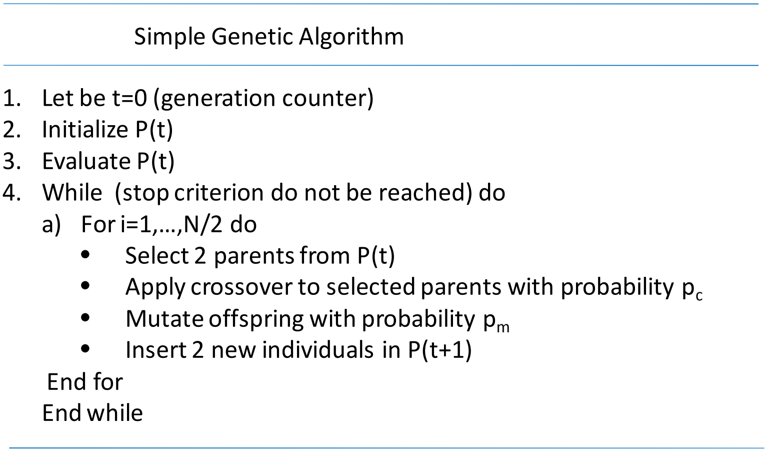
- Establishing a policy of exclusion of donors.
- Establishing and following strict storage criteria of blood components
- And avoiding unnecessary transfusions.

However, while these policies are completely established on Mexican hospitals, is necessary to have tools that enable health specialists to prevent this condition with a certain degree of confidence.

1.2 Genetic Algorithm

Genetic algorithms (GAs) were initially developed by Holland and his students, around the seventies [9]. Their study was motivated by the huge success of the process of natural-systems adaptation. Subsequently, the ideas developed by these early researchers were applied to the fields of optimization and machine learning [8]. The success of these applications was so extraordinary, that in a few years GAs became the first choice for solving complex problems.

Genetic algorithms are the most popular class of so-called evolutionary algorithms whose inspiration source has been the biological evolution. In its simplest form, a genetic algorithm can be described by the algorithm known as SGA. This algorithm is shown in Figure 1.



```
Simple Genetic Algorithm

1. Let be t=0 (generation counter)
2. Initialize P(t)
3. Evaluate P(t)
4. While (stop criterion do not be reached) do
   a) For i=1,…,N/2 do
      • Select 2 parents from P(t)
      • Apply crossover to selected parents with probability $p_c$
      • Mutate offspring with probability $p_m$
      • Insert 2 new individuals in P(t+1)
   End for
   End while
```

Fig. 1. SGA

Although the general mechanism of these algorithms is simple, it has been theoretically demonstrated that when they use elitist selection mechanisms, they converge to the global optimum of certain functions whose domain can be an arbitrary space. Rudolph in 1996 generalized previous developments in convergence theory for binary search spaces, and Euclidean general search spaces [15].

In order to improve the performance of the genetic algorithm, it was hybridized with exploration and exploitation techniques developed in 2010 by Torres [22] and also used in [23]. These techniques were used for the selection of the subset of variables that better describe a phenomenon. Another important concepts added to the GA are testor and typical testor. This kind of hybridization has proved to be very effective in both supervised and unsupervised learning problems as Torres et al, already evidenced [21].

Hybridization of metaheuristics with different algorithmic ideas and sometimes with mechanisms from other areas is not a new idea; over the last years, a large number of algorithms that does not correspond to a pure approach have been reported. Hybrid metaheuristics have become a very promising field [4].

The main idea of hybridized metaheuristics is to take advantage of their potentialities while avoiding theirs weaknesses using mechanisms or techniques from other sources.

Below concepts of testor and typical testor will be described. These concepts are very important because they were the basis for the identification of the most TRALI-related variables.

1.3 Testors and Typical Testors

The concept of typical testor appeared in the middle of the fifties in Russia [5]. One of its first applications was failure detection in electrical circuits, followed by problems as supervised classification, and variable selection related with the geological area [2]. Among the group of researchers who worked with this tool, the research of Dmitriev Zhuravlev et al., is very remarkable because they pioneered the use of typical testors for feature subset selection [7].

According to Torres [22], an easy way to describe testor is: a set of features required to distinguish two objects that belong to different classes. This set not necessarily has to be minimal. On the other hand, typical testor is the minimum set of features required to distinguish two objects belonging to different classes. In other words, if a feature is eliminated from a typical testor, it will be not more a testor [17].

Let us suppose U is a collection of objects, and these objects are described by a set of n features, and these objects belong to k classes. In order to determine the testors and typical testors of this collection is need to have a structure called basic matrix, which is based in the difference matrix.

The difference matrix is made starting from the comparison of each feature of the objects from the same class against objects from the other classes. Difference matrix (DM) is the matrix whose content let us distinguish objects from different classes. DM is binary coded, using 1 if difference exists and 0 if difference doesn't exist.

Once the DM has been constructed, the basic matrix (BM) has to be created. BM is constituted by all the basic rows from DM.

Let ip and iq, to be two rows from DM: ip is a basic row from DM, if there is not any row lesser than ip in DM. We say that ip<iq if $\forall i \quad ip_i \leq iq_i$ and $\exists j$ such as $ip_j \neq iq_j$.

The subset of features T of a certain BM is a testor, if when eliminate from BM all features except those who belong to T, there is not any zero row. The set T is typical, if by eliminating a feature $j \in T$, T is no more testor. The set of all typical testors of MD is equal to the set of all typical testors of MB [11].

Due to determining all typical testors from a certain BM is a very complex problem, several researchers have developed special algorithms like BT [18], TB [18], REC [19] and LEX [17] among others. This problem has an exponential complexity and it depends on the size of the BM [16].

In order to apply the concepts of testor and typical testors, it is important to establish how comparisons are going to be made. One of the most common comparison criterion is known as strict equality, because is applicable to any kind of domain. The way this kind of comparison is performed is described below.

Let us suppose $\Omega=\{O_1, O_2,\ldots,O_m\}$ is a set of m objects and $I(O_1), I(O_2),\ldots,I(O_m)$ their descriptions on terms of their features, $R=\{x_1, x_2, \ldots, x_n)$, where each feature x_j, has associated a set of acceptable values M_j i.e. $I(O_i)=(x_1(O_i), x_2(O_i), \ldots, x_n(O_i)$, and $x_i(O_i) \in M_j$, j=1, 2, …, n and i=1, 2, …, m (Santiesteban and Pons, 2003).

Each type of variable depends of the nature of its set of acceptable values; a set M_j could be {0,1}, the set of real numbers, the set integer numbers, etc.

The strict comparison criterion is described in the next function:

$$C_k(x_k(O_i), x_k(O_j) = 0 \text{ if } x_k(O_i) = x_k(O_j) \quad ; 1 \text{ other wise} \tag{1}$$

In this work, this criterion was used with the whole set of variables. In order to facilitate the variable managing, each feature was discretized into classes according to the recommendation of health experts.

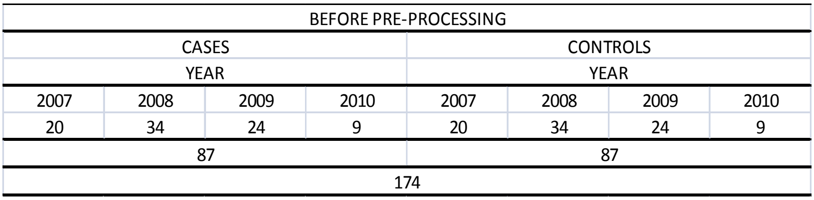
2 Methodology

The current research was made in the “Centenary Miguel Hidalgo” hospital located in Aguascalientes City in Mexico. This is one of the most important health care specialist hospitals in the center of Mexico. The collecting task finished in year 2011. And it constitutes the total information from years 2007, 2008, 2009 and also 2010.

As already mentioned, the research presented here, is based on the medical model known as “two hits” that consists of two main groups of variables that can be related with TRALI.

These two groups are: first hit variables, and second hit variables. Nevertheless, researchers found a little new group of variables than can complete the model already mentioned.

At the beginning, the historic archive of patients was reviewed looking for TRALI cases. The period of time considered covers from 2007 to 2010, because this syndrome is rare and the collecting of cases is slow.



BEFORE PRE-PROCESSING							
CASES				CONTROLS			
YEAR				YEAR			
2007	2008	2009	2010	2007	2008	2009	2010
20	34	24	9	20	34	24	9
87				87			
174							

Fig. 2. Cases and controls before pre-processing phase

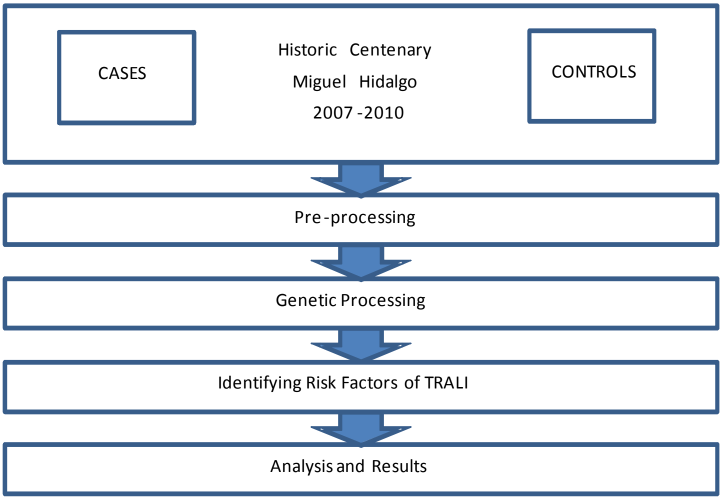
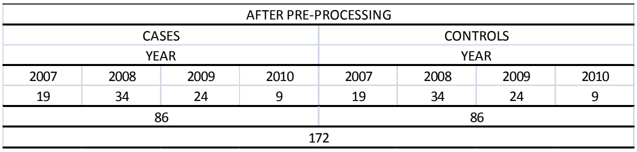


Fig. 3. General Approach

In figure 2 can be seen the number of cases and controls that were obtained from the historic archive of Miguel Hidalgo hospital. 87 cases and 87 controls were collected in the way this figure illustrates. The number of cases was the total of TRALI cases presented in those years and per each case one control was included.

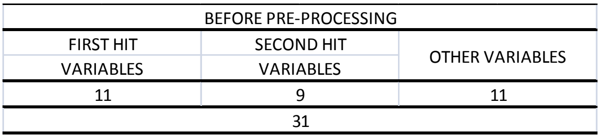
In order to make easier the assessment of the general approach, figure 3 shows a scheme of it. As the reader can see, figure 3 reveals the process done with data. Once data were compiled, a pre-processing phase was completed. As a result of this phase, the number of cases and also controls were reduced because one case had missing values in one variable. So the total number of cases and controls used from this stage on ahead are shown in figure 4.



AFTER PRE-PROCESSING							
CASES				CONTROLS			
YEAR				YEAR			
2007	2008	2009	2010	2007	2008	2009	2010
19	34	24	9	19	34	24	9
86				86			
172							

Fig. 4. Cases and controls after pre-processing phase

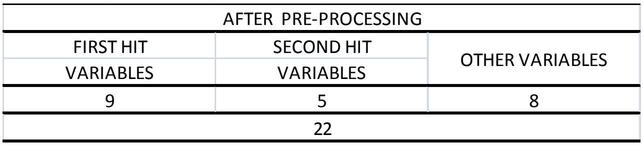
The pre-processing phase also influenced the number of variables in the research because the variables who did not contribute to risk of developing TRALI were eliminated. At the beginning, researchers had 31 variables grouped as figure 5 shows.



BEFORE PRE-PROCESSING		
FIRST HIT	SECOND HIT	OTHER VARIABLES
VARIABLES	VARIABLES	
11	9	11
31		

Fig. 5. Variables before the pre-processing phase

After the pre-processing phase, the number of variables kept was only 22 and they were distributed according to figure 6.



AFTER PRE-PROCESSING		
FIRST HIT	SECOND HIT	OTHER VARIABLES
VARIABLES	VARIABLES	
9	5	8
22		

Fig. 6. Variables after the pre-processing phase

Once the pre-processing phase was completed, the genetic processing began. This process is presented in detail in the hybrid evolutionary algorithm section. (Section 3).

The hybrid evolutionary algorithm gave us the whole set of typical testors associated to the learning matrix. We can say that each typical testor is formed by a set of variables that being together, better describe the chance of presenting TRALI by a patient.

Each typical testor was studied and they together let us computing the informational weight associated to each single variable. This way, not only the relationship between the variable and the syndrome (TRALI) is known, but also its dimension.

When the measurement of the influence each variable has for presenting the syndrome is obtained, expert physicians analyzed the results and fixed a specific cutoff for the most important variables identified as the main risk factors for TRALI.

3 Hybrid Evolutionary Algorithm

The determination of risk factors for TRALI is performed by a system based on a genetic algorithm called TRALI-GA. Previous experiments [21] suggested that mechanisms used to hybridize this metaheuristic, work better on it than in a univariate marginal distribution algorithm. TRALI-GA and its components are presented in the follow subsections.

3.1 TRALI-GA

The Evolutionary Hybrid Genetic Algorithm for the identification of risk factors for TRALI is shown below:

```
Algorithm - TRALI-GA
begin
  Initial pre-processing.
  Generating Difference Matrix phase.
  Generating Basic Matrix phase.
  Generate initial population (randomly).
  Apply accelerating operator.
  Compute fitness.
  If necessary
    Apply improvement mechanism.
    Compute fitness.
 End if.
  population ← new initial population
  Repeat
     Begin /* New Generation */
        Repeat
         Begin /* reproductive cycle for pairs of
              individuals */
           Apply selection operator.
           Apply crossover operator.
           Apply accelerating operator.
           Compute fitness.
           If necessary
```

```
            Apply improvement mechanism.
             Compute fitness.
          End if.
           Count = Count+1;
         End.
        Until Count = (generation size/ 2).
        Order of population by fitness.
        Apply elitism.
        Population ← new population
     End.
  Until (stopping criterion is reached)
  Final set of typical testors analysis.
  Compute informational weight for each feature
  Get final features subset selection.
End
```

As can be seen on previous pseudo code, the first step consisted of pre-processing, is made for reducing the size of the problem in terms of number of variables. After the pre-processing and based on related literature, three groups of variables were identified:

1. First hit variables
2. Second hit variables
3. Other variables

Once the three groups of variables were identified, and some variables were eliminated, the remained set was discretized. After that, the difference matrix was obtained. This matrix is formed by the differences found between each case and the others in the learning matrix.

Next step consists in the generation of the basic matrix; this phase consists on finding the basic difference between each individual in the difference matrix versus all others. This matrix conserves only the basic differences in the difference matrix. Some differences can be considered redundant while others are not. This concept can be reviewed in the section 1.3; where basic differences are represented as basic rows.

When the basic matrix is obtained, the evolvable process begins. Each individual is coded by means of a binary string whose size corresponds to the number of variables. Each bit in the string takes the value 1 if the variable that corresponds to it is present or 0 if it is absent.

The evolvable process starts with the generation of a random population that is potentialized through the accelerating operator, which conserves the vital information a typical testor has to have according with a fast analysis of the basic matrix. This operator applies a logic OR to the actual chromosome [22].

Once the accelerating operator is applied, the fitness of the individual is computed. This calculus consists of three possible weights: the maximum value of an individual is assigned to its fitness when it is a typical testor. A medium weight is given when the individual is a testor but not a typical testor. Finally, the minimum weight is obtained for those individuals that are not typical testors, neither simple testors.

Fitness function assignation (ff-value), is performed according to next function.

$$\textit{ff-value} = \begin{cases} 5 & \textit{if the individual is not a testor or typical testor} \\ 10 & \textit{if the individual is a testor} \\ 20 & \textit{If the individual is a typical testor} \end{cases} \tag{2}$$

Considering the value of the fitness function (based on the individual's condition), the algorithm resolve if an improvement is required. Obviously, when an individual is a typical testor, none improvement is required, but if the individual is weighted as a simple testor, a reduction in the number of variables selected is needed; on the other hand, if the individual is weighted with the minimum value, an increment in the number of variables selected as a components of the individual is required for its improvement. So, the improvement mechanism works as a guided mutation (increasing or reducing the number of variables of the individual). A complete description of the whole set of mechanisms used in this work, can be consulted in Torres et al., 2012 [23].

After the improvement phase was performed (if it was needed), the population is sorted by fitness beginning with best individuals and ending with the worst ones. The new population is created by means of elitism and the application of the complete set of operators to the current population. Then the iterative process begins. This process consists on the generation of new populations that continually improve the previous ones. When stopping criterion is reached, the final set of typical testors is obtained and the informational weight is computed. This last computation let us know the importance of each variable.

The TRALI-GA was hybridized using a local mechanism that let us improve a specific solution (individual), looking for a typical testor when we have a single testor. This could be done because we know that a simple testor have inside as a minimum one typical testor; so the mechanism try to find it deleting possible redundant variables.

The algorithm was also strengthening using elitism to accelerate and to guarantee its convergence. Another special feature of this algorithm consists on the use of a global search operator obtained from a simple analysis of the basic matrix. This way, we rescue the better of the interior and exterior scale algorithms for finding typical testors.

The use of a logical combinatory focus, joined with a genetic algorithm and the special mechanisms, constitutes an interesting hybridization for supervised learning.

As an immediate result of the proposed system, we obtained an efficient exploration and exploitation of the search space.

4 Results and Discussion

The analysis of the results from TRALI-GA, revealed that the most important variables physician have to take care of, are grouped in three categories. Each category is presented at a different time. They are: the first hit variables, the second hit variables and other variables that were found in this research.

The original number of variables was 31, after the preprocessing this number was reduced to 22. Once the whole process finished, we only kept 15 variables.

The experimental results shown that even though genetic algorithms are known as rapid convergence algorithms, the set of mechanisms added, let it conserves the diversity required for looking typical testor in the whole solutions space.

Experiments indicate that the algorithm found the whole set of typical testors in all the executions. Results are described from ten sequential executions of TRALI-GA.

As mentioned in methodology, once the algorithm provided the risk factors for TRALI and their respective weigh (computed as the informational weight), expert physicians established the cutoff point for accepting a variable as an important risk factor. The cutoff point was fixed over 50%.

The resulting variables associated with the first hit were reduced to 8 and they are presented in table 1. This group of variables can be detected in the original disease of a patient. This disease is what causes the patient to be transfused; and could be a surgery, sepsis, trauma or any other medical affection.

Table 1. First Hit Variables

Variable	Weight attached
Age	100.00
Base Disease	80.06
PaO2/FiO2	100.00
History of heart attack	60.54
Hematological malignancy	55.08
Lung disease	52.56
Previous Surgery	53.31
Previous transfusion	52.10

Variables corresponding to the second hit are shown in table 2. These variables were reduced to 4. The second hit in the model of two hits, is the surgery in which the patient is blood transfused. Many of times, the surgery can be associated to a polytrauma because a crash car, another kind of accident or disease.

Table 2. Second Hit Variables

Variable	Weight attached
More than 3 globular packages	50.50
More than 3 units of fresh frozen plasma	57.10
Multiparous donor	51.47
Fresh frozen plasma dated with more than 2 weeks	57.07

Besides first and second hits variables, researchers wanted to know if some other suggestive variables could contribute to suffer TRALI; so, the genetic process looked for 8 variables that were labeled as "other variables". This group was reduced to 3 and they are presented in table 3.

Table 3. Other Variables

Variable	Weight attached
Pulmonary edema	52.01
Diuretic therapy	52.00
Smoking habits index greater than 20	57.61

Based on the obtained results, we can conclude that TRALI is presented mostly in patients with predisposition and presence of risk factors related with previous cardiac or lung disease, smoking, and previous hematological malignancies. We also can say that risk increases if the blood units used to obtain the frozen fresh plasma have more than 2 weeks in storage. "TRALI" presented in our hospitable environment is more related with the use of fresh frozen plasma than the use of globular packages. This relationship can be associated with the time of storage of the blood component (unfortunately a very frequent situation).

Blood components donated by multiparous women had less relationship with TRALI than the one expected by physicians (probably because of the deficient record of donors in the blood bank). Risk factors related to the medical history of a patient: cardiac history, previous lung disease and hematological malignancy represent a comparable risk to presenting TRALI.

5 Conclusions and Future Work

Nowadays, medicine is one of the most notorious areas for applying intelligent techniques. The reason is simple: when preserving health and saving life is possible, every scientist is pleasure to collaborate. Artificial intelligence is being applied to medicine as a natural tool to discover, to strengthen or reject health theories.

The identification of the risk factors for TRALI, is a challenging task for the artificial intelligence area because it constitutes a very difficult to diagnose syndrome. Actually, nowadays the Centenary Miguel Hidalgo Hospital does not routinely runs the tests required to diagnose this medical condition.

As mentioned in the introduction, for long time TRALI was considered as a unknown secondary effect associated to blood transfusion, even though currently it is recognized as the leading cause of transfusion-related death in the United States of America, and the second one in the United Kingdom.

Although TRALI is not easy to diagnose, physicians can take care of critical risk factors with an insignificant cost and big benefits; and this is why many researches in health care area, consider it a very transcendent topic.

We can also say that hybridization improves the results for the problem we attacked. We know that including all the knowledge we can to our algorithm, produces better results since search space is reduced. As an important part of the used system, it can be stated that the use of typical testors, the use of the accelerating operator, and also the use of the improvement mechanism, make possible to have a powerful tool to solve complex real life problems.

Finding typical testors of a dataset is an exponential problem, so, applying a metaheuristic in the solution is well justified. Informational variable weight is also a very important concept, since it let us to qualify the importance of each variable.

The application of the described method is also promising to other diagnostic tasks in medicine.

References

1. Añón, J.M., García de Lorenzo, A., Quintana, M., González, E., Bruscas, M.J.: Lesión pulmonar aguda producida por transfusión. Publicado en Med Intensiva 34(02), 139–149 (2010)
2. Alba, C.E., Santana, R., Ochoa, R.A., Lazo, C.M.: Finding Typical Testors By Using an Evolutionary Strategy. In: Proceedings of V Iberoamerican Workshop on Pattern Recognition, Lisbon, Portugal, pp. 267–278 (2000)
3. Barnard, R.D.: Indiscriminate transfusion: a critique of case reports illustrating hypersensitivity reactions. NY State. J. Med. 51, 2399–2402 (1951)
4. Blum, C., Roli, A., Sampels, M. (eds.): Hybrid Metaheuristics. An Emerging Approach to Optimization". SCI, vol. 114. Springer (2008)
5. Cheguis, I.A., Yablonskii, S.V.: "About testors for electrical outlines. Uspieji Matematicheskij Nauk 4 (66), 182-184 (1955) (in Russian)
6. Cuellar. L.F.: Factores de riesgo asociados a TRALI en el paciente politraumatizado.?'Es la transfusión masiva el único factor predisponente? Tesis de Especialidad, Centenario Hospital Miguel Hidalgo y Universidad Autónoma de Aguascalientes. México, Enero (2012)
7. Dmitriev, A.N.: On the mathematical principles of patterns and phenomena classification. Diskretnyi Analiz 7, 3–15 (1966) (In Russian)
8. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading (1989)
9. Holland, J.H.: Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. The University of Michigan Press, Ann Arbor (1975); 2rd edn. The MIT Press, Cambridge (1992)
10. Kleinmann, S., Caulfield, T., Chan, P., Davenport, R., McFarland, J., McPhedran, S., et al.: Towards an understanding of transfusion-related acute lung injury: statement of a consensus panel. Transfusion 44, 1774–1789 (2004)
11. Lazo-Cortés, M., Ruiz-Shulcloper, J.: Determining the feature relevance for non classically described objects and a new algorithm to compute typical fuzzy testors. Pattern Recognition Letters 16, 1259–1265 (1995)

12. Muller, J.Y.: TRALI: from diagnosis to prevention. Transfus. Clin. Biol. 12(2), 95–102 (2005)
13. Palfi, M., Berg, S., Ernerudh, J.: Randomized Controlled Trial of Transfusion-Related Acute Lung Injury: Is plasma from multiparous blood donors dangerous? Transfusion 41, 317–322 (2001)
14. Popovsky, M.A., Moore, S.B.: Diagnostic and pathogenetic considerations in transfusion-related acute lung injury. Transfusion 25(6), 573–577 (1985)
15. Rudolph, G.: Convergence of Evolutionary Algorithms in General Search Spaces. In: Proceedings of the Third IEEE Conference on Evolutionary Computation (1996)
16. Sanchez-Díaz, G., Lazo-Cortés, M.: CT-EXT: An Algorithm for Computing Typical Testor Set. In: Rueda, L., Mery, D., Kittler, J. (eds.) CIARP 2007. LNCS, vol. 4756, pp. 506–514. Springer, Heidelberg (2007)
17. Santiesteban, A.Y., Pons, P.: LEX: Un Nuevo Algoritmo para el Cálculo de los Testores Típicos. Revista Ciencias Matemáticas 21(1), 85–95 (2003) ISSN: 0256-5374
18. Shulcloper, J.R., Bravo, M.A., Lazo, C.: Algoritmos BT y TB para el cálculo de todos los test típicos. Revista Ciencias Matemáticas VI(2), 11–18 (1985)
19. Shulcloper, J.R., Alba, C., Lazo, C.: Introducción al reconocimiento de Patrones: Enfoque Lógico Combinatorio". Serie Verde (51), 188 (1995)
20. Shortliffe, E.H.: Consultation systems for physicians: the role of artifitial intelligence techniques. In: Proceedings Third National Conference of the Canadian Society for Computational Studies of Intelligence, Victoria, British Columbia (May 1980)
21. Torres, D., Ponce-de-León, E., Torres, A., Ochoa, A., Díaz, E.: Hybridization of Evolutionary Mechanisms for Feature Subset Selection in Unsupervised Learning. In: Aguirre, A.H., Borja, R.M., Garciá, C.A.R. (eds.) MICAI 2009. LNCS, vol. 5845, pp. 610–621. Springer, Heidelberg (2009)
22. Torres, M.D.: Metaheurísticas Híbridas en Selección de Subconjuntos de Características para Aprendizaje no Supervisado. Tesis Doctoral. Universidad Autónoma de Aguascalientes, México (May 2010)
23. Torres, M.D., Torres, A., Ortiz Zezzatti, C.A., Sentí, E.E., Díaz, E.D., Landín, C.J., Amador, C.E.: Hybrid Algorithm Applied to the Identification of Risk Factors on the Health of Newly Born in Mexico". In: Ortiz Zezzatti, C., Chira, C., Hernandez, A., Basurto, M. (eds.) Logistics Management and Optimization through Hybrid Artificial Intelligence Systems, pp. 83–112. Information Science Reference, Hershey (2012), doi:10.4018/978-1-4666-0297-7.ch004
24. Toy, P., Popovsky, M.A., Abraham, E., Ambruso, D.R., Holness, L.G., Kopko, P.M., et al.: National Heart Lung and Blood Institute working group on TRALI, Transfusion-related acute lung injury: Definition and review. Crit. Care Med. 33, 721–726 (2005)

Generation and Exploitation of Semantic Information Using an Epidemiological Relational Database as a Primary Source of Information

David González-Marrón[1,2],Miguel González-Mendoza[2], and Neil Hernández-Gress[2]

[1] Instituto Tecnológico de Pachuca,Carretera México-Pachuca Km 81.5,Pachuca, Hidalgo, México
dgonzalez@itpachuca.edu.mx
[2] Tecnológico de Monterrey, Carretera Lago de Guadalupe Km 2.5,
Atizapán de Zaragoza, Edo. de México, México
{mgonza,ngress}@itesm.mx

Abstract. Recently, many researchers are trying to automate the automatic production of ontologies, realizing extraction of the information contained in databases using reverse engineering, however, because of the continuous evolution of standards in relational databases, not all the characteristics presented by modern databases managers are still supported. Besides as databases can be implemented in several ways in relational databases using different commands, this frequently causes differences when creating ontologies. In this paper, is presented how this approach can be applied to an epidemiological database and the advantages of use, when realizing query exploitation.

Keywords: Heterogeneous Databases, Semantic Web, Ontologies.

1 Introduction

Most of the information accessible in the web, is extracted from databases using specially designed programs. The amount of information existent in this databases is more than 70% the information of the web, this database information is commonly denominated as the "deep web" [1] because it is not accessible through general search engines. The semantic web is looking for mechanisms that permit to find, to share, and to combine the web information more easily [2]. The use of ontologies permits increment interoperability between several systems, and at the same time permits the use of the semantic technology, incrementing the potential of the formulated queries. In this job, an analysis of several mechanisms required to produce automatically ontologies is realized. An analysis of the advantages and disadvantages of exploiting semantic information for a common user is realized, it is described also the process required to produce ontologies, considering the RDF files integration as one of the fundamental Semantic Web activities. Since this paper is more related with data stored in databases, a mechanism to interact with this information is described; it is mentioned one of the most important approaches used by researchers when mapping the SQL databases schemas, and the subsequent process of transforming it into an ontology.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 446–457, 2013.

The process of extracting information using the information stored in SQL is a complex task, a relationship between SQL and the semantic web can be established, starting in SQL with the relational model, continuing with the table definition, posteriorly with constraints in insertions and updates and finishing with triggers, and in the semantic web starting with RDF files (the most simple semantic format), continuing with the RDFS (oriented to describe the Schema of the RDF), posteriorly with a most powerful language OWL (Ontology Web Language, suited to describe the relation between the semantic data) and finishing with Rules that permit to realize inference in the semantic data. The state of the art is the generation of basic ontologies, this ontologies are named by Sequeda et al. [3] as “putative ontologies”. Since they are simple ontologies that require refinement by domain experts. This paper is organized as follows: In section 2 a brief description of the alternatives to realize DB integration is realized; in section 3 are mentioned some advantages of realizing semantic integration; in section 4 several considerations about ontologies formulation are mentioned, as well as the problematics to formulate ontologies based in relational DB's; in section 5 is described how the epidemiological relational DB is converted into semantic information; in section 6 are mentioned the possibilities of query exploitation using the semantic information.

2 Database Integration

The integration of databases in a semantic environment, is considered a difficult task because the great number of variables involved in this activity, such as: Data incompatibility, data distribution, different semantic meaning, mechanisms of security, mapping tasks for databases, different data representation and data accuracy. In this review, we get focused in the use of ontologies to achieve interoperability in databases, using a semantic approach. It is explained the reverse engineering process applied to databases, in order to manage the content of an entire database in semantic files basically in the OWL language. Two classical approaches are derived from this process. The first one is related to replication of the total information contained in the database, and the second one more related to the semantic associations contained in every table that conforms the database, keeping data where are produced.

Several tools and protocols exist, the most significant advances in this area, are done by the OMG [4]. In the figure below, a description of different alternatives to create semantic information by this group is given.

2.1 Approach to Access Data of Relational Databases Using Semantic Means

Two main alternatives exist to realize this activity, the first choice is the creation of static maps of the database schema, this process is still under development by many researchers and it is very promising. The second one is a semiautomatic process, which transforms the information into a semantic format. Below is presented a brief description for every alternative.

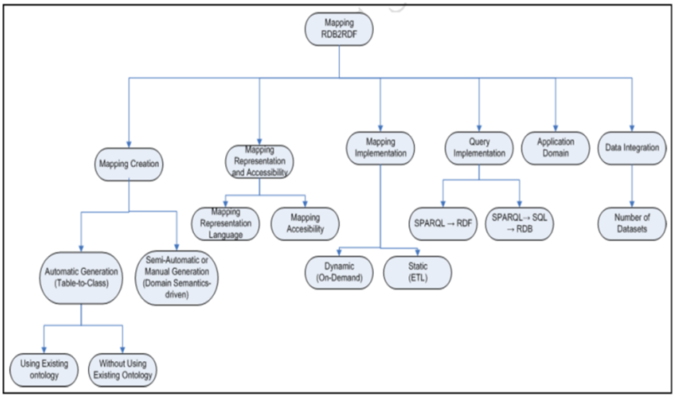


Fig. 1. Different alternatives to create and exploit semantic information

Static Maps

In this method, the mapping is realized over the database schema, without realizing physical copies of information. The structure of the relational database tables and attributes are mapped into corresponding classes and properties in a user-defined ontology. The relational database schema is a tuple R (U, D, dom, I). Where: R is known as a relation is defined as a set of tuples that have the same attributes, and a tuple usually represents information about one object; U is a finite set of attributes in a relation; D is the domain of the database; dom represents a function that maps U to D, where every attribute A*i* in U must have valid value in D*i*; I is a finite set of integrity constraint, which restricts data instances stored in database, two different kinds of integrities can be applied (entity integrity and referential integrity). Also, two main parts composed the process, the first one related with the data stored in DBs and the second one, more related with how data interrelates. Although the process of generating an ontology can be automated, actually the results produced by the technology require a refinement by experts in databases and in the areas of the domain of discourse is required.

Database Conversion into a Semantic Format

This method produces a conversion of the database and replicates data into a different format, commonly into triplets conformed by three elements (Subject, Predicate, and Object). Here, this conversion produces a great increment of the information. This conversion generally is realized into a RDF (Resource Description Framework) format, which is stored into repositories for being exploited using SPARQL (Sparql Protocol and RDF Query Language), several tools exist to store the RDF files produced into a relational database, in order to keep the data stored into a single platform (database manager). The ontology creation process can make use of a domain ontology, if used, the semantic information obtained will contain information about the domain selected, the URIs (Uniform Resource Identifiers) include information about the domain, in any case the information produced will consist by triplets in a common RDF format, for both cases, using a domain ontology or not, the assumption of a closed world (CW) is considered.

3 Why Use Semantic Integration?

The general aspects underlying database applications, is that they need semantic integration because of the structured data with more than one representation, applications must solve heterogeneities with respect to the schemas and their data to allow their manipulation or to enable the transformation of data and queries between schemas [5]. This task is related to conversion of data for queries exploitation. A conceptualization for query exploitation is showed in figure 2, in which a user demands a query that requires the integration of several databases. Since most of the time data are not compatibles, the use of semantic integration is proposed as an alternative to realize this integration.

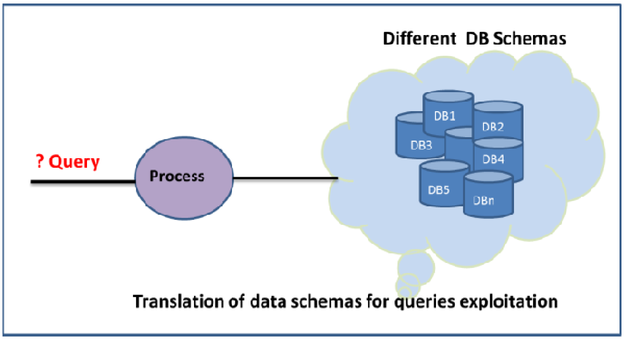


Fig. 2. Conceptual view of semantic query exploitation

3.1 Alternatives for Data Integration

Two alternatives exist to achieve data integration, the most common is the Static, using an “Extract, Transform, and Load” (ETL) process, which produces data conversion from a staging area to a target representation. The other approach is Dynamic (Data Virtualization), in which the data remain in its source, and a conceptual data view is produced as needed.

- Traditional ETL. Is a variation of the ETL paradigm, in which data sources are used to support business applications, and these applications often require many different formats, files, structures, and even different encoding. Normally there is a tendency to normalize data before attempting to integrate them into a target downstream system.
- Data Virtualization. Is opposed to the traditional method of extracting data from various sources and temporarily stores those data sets at a staging area. This method called Data Federation or Data Virtualization allows the source data set to remain in their original locations.
- Semantic Data Integration. An innovative approach which can be used to integrate data, it is an active area in different areas as Databases, information-integration and ontologies. It permits interoperability between different systems.

Semantic integration involves also techniques for matching databases schemas to answer complex queries that require multiple sources. Here, the data are transformed in order to exploit more the semantics of data, two different alternatives are of potential use. The first is similar to the ETL approach, but using semantic data organization, creating a replica of the data. The second is related to the approach used by the data virtualization, creating mapping files necessaries to locate data, but leaving data at original locations

3.2 Why Semantic Integration

The necessity of using semantic information is because the representation of data and the information itself is often bound tightly together, additionally because that information often lacks context. Most developers often think not on the data themselves, but in the structure of those data (data types, schemas, file formats, relational database constructs, and other structures that don't pertain directly to the information). In tightly coupled architectures, data structures are needed since they provide systems a way of interacting with the information they are being fed. However, in standards based with loosely coupled architecture, when the barriers to application integration are uninvolved, instead of being helpful, these different data structure representations, causes obstacles to data integration. It is because of this; that systems transcend just coupling the application interface and providing loose coupling at the semantic level [6].

4 Ontology Formulation

The most common definition of an ontology is proposed by Gruber: "*a formal, explicit specification of a shared conceptualization*" [7]. *Formal* means that the specification is encoded in a language whose formal properties are well understood; an *explicit specification* means that concepts and relationships in an abstract model are given explicit names and definitions. A *conceptualization* in this context, is referred to an abstract model of how people think about things, it is usually restricted to a particular subject area. *Shared* means that the main purpose of an ontology is to be used and reused across different applications and communities.

Several reasons make necessary the creation of an ontology: To share common understanding of the structure of information between people and software agents, to enable reuse of domain knowledge, to make domain assumptions explicit, to separate domain knowledge from the operational knowledge, and to analyze domain knowledge. There has been much discussion on what exactly is an 'ontology', however, there is coincidence with then next two approaches: a) Vocabulary of terms that refer to the things of interest in a given domain and b) Some specification of meaning for the terms grounded in some form of logic.

An ontology together with a set of concrete instances of the class constitutes a knowledge base. The taxonomy of classes is a primary focus in every ontology. It is necessary to emphasize that the ontology construction consists of the next six steps [8]:

1. Specification of the purpose, usage, scope and degree of formality of the ontology.
2. Data collection using various eliciting methods.
3. Conceptualization of domain terms (preliminary ontology).
4. Integration to other ontologies.
5. Formalization in an ontology language.
6. Evaluation of completeness, consistency and redundancy.

These steps must be considered in order to produce an ontology suited to the purpose pursued.

4.1 Problematic to Model Ontologies Based in Relational Databases

There are some impedance existent problems when a conversion to semantic data is required, ten different cases of primary and foreign keys combination must be considered, when realizing direct mapping. In the research realized by Sequeda et al. [3], are mentioned the basic considerations required to produce an automatic tool able to produce automatic ontologies. Other important variable to consider is the SQL evolution since its first version in 1986 (SQL-86) up to the next expected release (SQL:2011) , since the first version, the complexity process of interpreting in a simple way the information stored in the database into semantic data has been increasing. Since the first version of SQL in 1986, the language has evolved, permitting support of: Distributed databases, XML, recursive triggers, object-oriented technology. Besides of increasing the semantic relation between data. Many of the SQL DDL analyzers do not support, the different versions existent between these standards, even the SQL92 it is considered an important challenge for semantic automation, however, actually not exists a single product able to support triggers using semantic technology, this because of the complexity involved in these commands. Additionally every database software producer, implements the software with slight differences, requiring that a version for every database engine be developed.

5 Epidemiological Database

The epidemiological database SINAVE [9] is conformed by several databases (AIDS, Tuberculosis, Cancer, Measles, Diabetes, and other diseases), the database is constituted by almost 300 tables. One table is frequently used by more than one disease, this is caused because personal information about patients (such as living places, hospital care centers and information related) is shared frequently between these databases. For this paper, the Tuberculosis database was selected because all this information is available, was validated for experts and contains enough data to validate the method proposed.

A simplified model conformed by seven tables is used for making the conversion to semantic data.

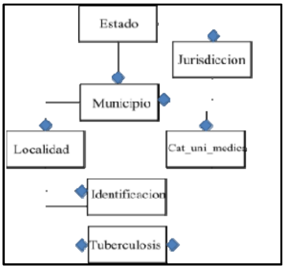
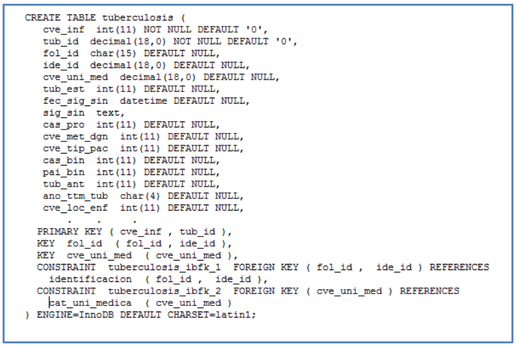


Fig. 3. Relational Model of the tuberculosis database using ELKA representation

As can be seen exists relation between tables, permitting data integration. This information is of fundamental importance to produce semantic information, this database should be transformed to semantic data using reverse engineering, this process can be realized using products developed by the semantic community. One of the most interesting aspects when analyzing the database design, is shown in the next figure.



```
CREATE TABLE tuberculosis (
   cve_inf  int(11) NOT NULL DEFAULT '0',
   tub_id  decimal(18,0) NOT NULL DEFAULT '0',
   fol_id  char(15) DEFAULT NULL,
   ide_id  decimal(18,0) DEFAULT NULL,
   cve_uni_med  decimal(18,0) DEFAULT NULL,
   tub_est  int(11) DEFAULT NULL,
   fec_sig_sin  datetime DEFAULT NULL,
   sig_sin  text,
   cas_pro  int(11) DEFAULT NULL,
   cve_met_dgn  int(11) DEFAULT NULL,
   cve_tip_pac  int(11) DEFAULT NULL,
   cas_bin  int(11) DEFAULT NULL,
   pai_bin  int(11) DEFAULT NULL,
   tub_ant  int(11) DEFAULT NULL,
   ano_ttm_tub  char(4) DEFAULT NULL,
   cve_loc_enf  int(11) DEFAULT NULL,
       .    .    .
  PRIMARY KEY ( cve_inf , tub_id ),
  KEY  fol_id  ( fol_id , ide_id ),
  KEY  cve_uni_med  ( cve_uni_med ),
  CONSTRAINT  tuberculosis_ibfk_1  FOREIGN KEY ( fol_id ,  ide_id ) REFERENCES
    identificacion  ( fol_id ,  ide_id ),
  CONSTRAINT  tuberculosis_ibfk_2  FOREIGN KEY ( cve_uni_med ) REFERENCES
    cat_uni_medica  ( cve_uni_med )
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Fig. 4. Partial description of the tuberculosis table

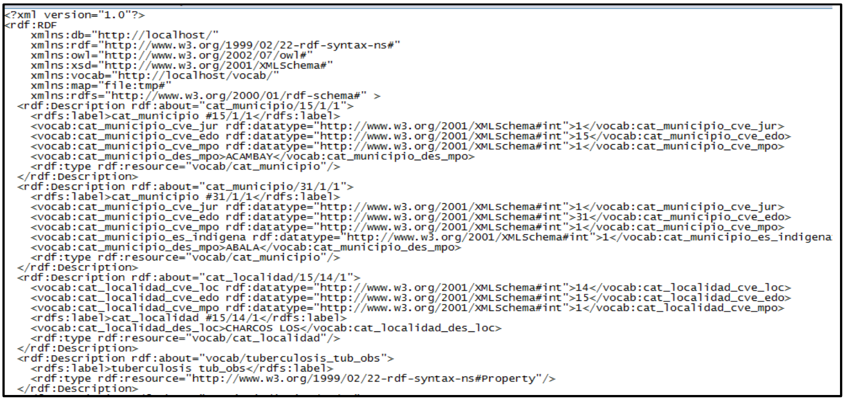
The most interesting part when analyzing relational databases is related to primary and foreign keys, as can be seen in the figure 4, the primary key is a composed key conformed by the fields (cve_inf and tub_id), contains two foreign keys named as *tuberculosis_ibfk_1* and *tuberculosis_ibfk_2*, these two foreign keys associate a relation of the table tuberculosis with other two tables named as: identificacion and cat_uni_medica.

When the data stored in the DB is converted to semantic data, one alternative is to convert everything to triplets of information, in this case, the d2rserver was used to realize this action producing an RDF file as can be seen in figure 5, it is important to mention that the amount of data produced is considerably bigger than the required in the relational model, this is done because relational databases are designed to optimize space of stora-ge, and the semantic storage is not designed with this purpose.

This data produced as rdf, needs and additional process to convert it to a useful ontology, the potential of the RDF data model is well suited to merge data (triples) from arbitrary sources. It is common to have unconnected descriptions of the same object in different sources. Because of this it is required to relate unconnected descriptions of the same object to share and interrelate information, one possible way to make this process possible is to relate information using commands of OWL as (owl:sameAs, or using the owl:InverseFunctionalProperty commonly named as IFP) which are useful to establish equivalence between data.

The first command sameAs is useful to relate synonyms based in the structure of the data, the second one is to indicate a relation between the data based on the data values. One example for the first definition can be the *curp* used in one database and the *regist_pobl* used in other source, for the second example the association is given when some individuals have the same value in a specific property, for example *rfc*.

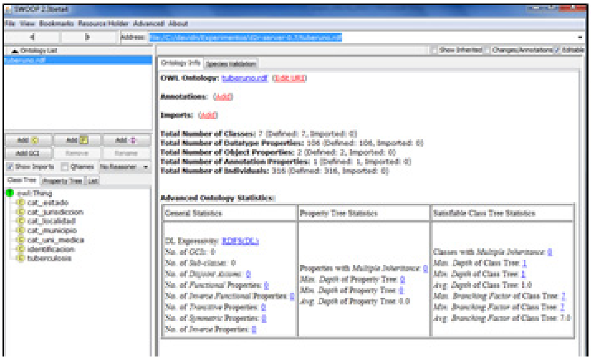
An example of the file produced using reverse engineering is showed below.



```
<?xml version="1.0"?>
<rdf:RDF
    xmlns:db="http://localhost/"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:vocab="http://localhost/vocab/"
    xmlns:map="file:tmp#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" >
  <rdf:Description rdf:about="cat_municipio/15/1/1">
    <rdfs:label>cat_municipio #15/1/1</rdfs:label>
    <vocab:cat_municipio_cve_jur rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</vocab:cat_municipio_cve_jur>
    <vocab:cat_municipio_cve_edo rdf:datatype="http://www.w3.org/2001/XMLSchema#int">15</vocab:cat_municipio_cve_edo>
    <vocab:cat_municipio_cve_mpo rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</vocab:cat_municipio_cve_mpo>
    <vocab:cat_municipio_des_mpo>ACAMBAY</vocab:cat_municipio_des_mpo>
    <rdf:type rdf:resource="vocab/cat_municipio"/>
  </rdf:Description>
  <rdf:Description rdf:about="cat_municipio/31/1/1">
    <rdfs:label>cat_municipio #31/1/1</rdfs:label>
    <vocab:cat_municipio_cve_jur rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</vocab:cat_municipio_cve_jur>
    <vocab:cat_municipio_cve_edo rdf:datatype="http://www.w3.org/2001/XMLSchema#int">31</vocab:cat_municipio_cve_edo>
    <vocab:cat_municipio_cve_mpo rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</vocab:cat_municipio_cve_mpo>
    <vocab:cat_municipio_es_indigena rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</vocab:cat_municipio_es_indigena
    <vocab:cat_municipio_des_mpo>ABALA</vocab:cat_municipio_des_mpo>
    <rdf:type rdf:resource="vocab/cat_municipio"/>
  </rdf:Description>
  <rdf:Description rdf:about="cat_localidad/15/14/1">
    <vocab:cat_localidad_cve_loc rdf:datatype="http://www.w3.org/2001/XMLSchema#int">14</vocab:cat_localidad_cve_loc>
    <vocab:cat_localidad_cve_edo rdf:datatype="http://www.w3.org/2001/XMLSchema#int">15</vocab:cat_localidad_cve_edo>
    <vocab:cat_localidad_cve_mpo rdf:datatype="http://www.w3.org/2001/XMLSchema#int">1</vocab:cat_localidad_cve_mpo>
    <rdfs:label>cat_localidad #15/14/1</rdfs:label>
    <vocab:cat_localidad_des_loc>CHARCOS LOS</vocab:cat_localidad_des_loc>
    <rdf:type rdf:resource="vocab/cat_localidad"/>
  </rdf:Description>
  <rdf:Description rdf:about="vocab/tuberculosis_tub_obs">
    <rdfs:label>tuberculosis tub_obs</rdfs:label>
    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  </rdf:Description>
```

Fig. 5. Piece of information produced as result of converting RDB to an RDF file

As can be seen triplets of information are produced, the URIs increment considerably the space required in this format, also the purpose of this format is more related to permit an automatic interchange between computers and not between humans. In order to make more legible this information, and to increment the interoperability between systems, several graphical editors have been created as Protegé, Smooth, Topbraid and others. An example of interpreting graphically the produced file with the Smooth editor is shown below.



OWL Ontology: tuberuno.rdf (Edit URI)

Annotations: (Add)

Imports: (Add)

Total Number of Classes: 7 (Defined: 7, Imported: 0)
Total Number of Datatype Properties: 106 (Defined: 106, Imported: 0)
Total Number of Object Properties: 2 (Defined: 2, Imported: 0)
Total Number of Annotation Properties: 1 (Defined: 1, Imported: 0)
Total Number of Individuals: 316 (Defined: 316, Imported: 0)

Advanced Ontology Statistics:

General Statistics	Property Tree Statistics	Satisfiable Class Tree Statistics
DL Expressivity: RDFS(DL) No. of GCIs: 0 No. of Sub-classes: 0 No. of *Disjoint Axioms*: 0 No. of *Functional Properties*: 0 No. of *Inverse Functional Properties*: 0 No. of *Transitive Properties*: 0 No. of *Symmetric Properties*: 0 No. of *Inverse Properties*: 0	Properties with *Multiple Inheritance*: 0 *Max. Depth of Property Tree*: 0 *Min. Depth of Property Tree*: 0 *Avg. Depth of Property Tree*: 0.0	Classes with *Multiple Inheritance*: 0 *Max. Depth of Class Tree*: 1 *Min. Depth of Class Tree*: 1 *Avg. Depth of Class Tree*: 1.0 *Max. Branching Factor of Class Tree*: 7 *Min. Branching Factor of Class Tree*: 7 *Avg. Branching Factor of Class Tree*: 7.0

Fig. 6. Using the graphic editor Swoop, to realize the smushing process

In the figure 6 can be seen that the seven tables existent in the relational database schema, were transformed into classes. Using these tools it is considerably easier to realize the smushing process (process of refinement and association in ontologies). However, much information concerning to how the database schema was designed is stored in the database, and because of this, the process can be automated. It is considered a good alternative the use of a database schema to produce an ontology, because the DB schemas are formal documents that have been designed to capture a real world modeled by Information Systems specialists. Several authors [5][10][11][12][13][14] are producing different Semantic Information based in a same DB schema, this is not suitable, because causes uncertainty about the results obtained, however, this is actually happening and an agreement between these leaders must be done, in order to produce the same results always. OWL ontologies are formed by a sequence of annotations, axioms, and facts. The most essential elements are axioms and facts, which provide information about classes, properties and individuals According to Kashyap [15] the ontology construction from a relational DB requires analyzing the DB schema to determine keys, foreign keys and inclusion dependencies. In the figure 7 some of the 86 elements that conform DatatypeProperties and ObjectProperties of the ontology are shown. However, as can be seen the name of the

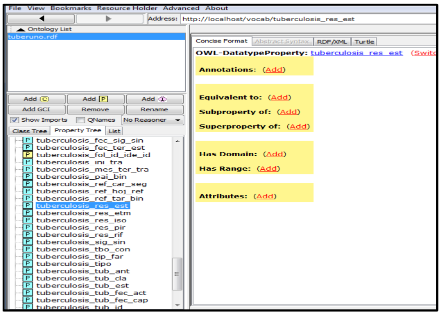


Fig. 7. Tuberculosis ontology using a CW assumption

elements that conform the tuberculosis ontology are given considering a close world CW (conception), because of this, the names used to describe the properties of objects and datatypes were defined only to be of practical use and efficiency in Spanish, the analysis realized was considering the point of view of the databases analysts, and a description of the meaning of every name was used to permit interchange concepts with the system users. On the opposite side considering an open world (OW) the analysis is directed to permit the data interchange between different actors because of this, the analysis, is realized considering the interoperability of the ontology. This two different points of views OW and CW consider different depth level of analysis, the OW is more standardized and consider a deeper analysis, for example the ontologies produced by the IDO developers [19] supported by the Burroughs Wellcome Fund and the National Institute of Allergy and Infectious Diseases. Produce ontologies based in a core ontology denominated IDO (Infection Disease Ontology), the ontology for tuberculosis and others shown in the figure 8, are parts of the subdomain specific IDO, still in development.

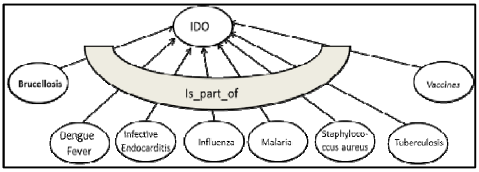


Fig. 8. Relation existent between IDO's ontologies

The Approach used by the CW existent in databases is more practical, looks for the development of useful systems able to permit support decisions, considers more important the optimization of data storage and the effectiveness of the software produced, it is not focused in getting compatibility with other systems. In the Tuberculosis database analyzed, the name of the properties were given using Spanish abbreviations, useful to database designers to differentiate concepts. At this moment the elaboration of tuberculosis ontology is in process, and the smushing process is realized in two different ways, using an ontology editor for OWL-RDF files, and manual modification for map files, the considerations for this elaboration are based in the inclusion of hyponyms and hypernyms. This association requires the consideration of similarities, we are considering the use of two types of similarities (string and synonyms) because of the complexity involved in the process. The inconvenient of using the semantic approach is precisely the complexity involved in the process, because it is required some refinement to be efficient in storage and speed execution, up to date the technology developed, it is in full swing, and is a growing area that need to be developed to facilitate interoperability between systems.

6 Exploitation Using a Semantic Approach

Additional information to exploit data can be obtained from ontologies because of the hierarchical structure of data contained in it, with this information is possible to realize data aggregation. The hierarchy, makes possible the inclusion of information related with the area to increase the power of the queries. Some examples of the queries that can be answered using semantic exploitation are: a) *Indicate the name of the people with tuberculosis in a particular municipality*; b) *Indicate the name of people with tuberculosis in a specified State*; c) *Total of sick people with symptomatology of "frequent coughing at nights" in a particular region of the country*; d) *Indicate the name of sick people and region of residence with contagious diseases with conceptual confusion inferior to 90%.*

Using relational query exploitation, the designer of the queries, must be concerned with the physical database schema in every design, in order to produce well-formed relational queries, one of the advantages that the semantic exploitation gives is that the user does not need to know every particular database design, but only the model of semantic data representation conformed by triplets. A lot of information requested using a semantic approach, it is not available in the relational model, but thanks to ontologies, new useful information can be added to the system

7 Conclusions

Extraction of information stored in relational databases using reverse engineering to transform it into semantic information, requires that the database be at least in 3NF (third normal form), because this ensure that the design is realized adequately, This process (3NF) involves having all the information atomized, an adequate use of primary keys and a correct association between tables must be assured. Once the extraction of information is realized, an additional processing must be done in order to get a useful ontology. It is necessary to work with the steps recommended by Du Bois [8] for ontology creation, this is not a trivial task. The ontology elaboration requires to parse the SQL script and start identifying the relevant elements needed to create the ontology. The incubator group W3C [4] still research thorough the RDB2RDF group how to improve the automatic conversion of data, the integration of information from several repositories, storage optimization, and query optimization Important initiatives as the project SQL2OWL looks of interests [3]. However, still it is not available to be evaluated.

It is important to conclude that from two alternatives to produce ontologies, we consider that "Direct-mapping" as the most important alternative to be analyzed for database designers. It is important to mention that the automatic generation of ontologies is still in development [17], a standardization is fundamental in this area, even the ontologies produced manually are not totally accepted by the community of ontology producers as can be seen in the research done by Todorov and Geibel [16]. The analyzed tools, recently produced by developers in the semantic web area, still require to realize certain activities manually, the W3C community in semantic web has developed important products as the D2RSERVER [18], this software able to transform relational data into RDF formats, produces direct mapping of relational schemas, but it is not specifically an ontology language. Also, several initiatives are proposed to improve queries performance and support of queries when exists incomplete information. One of the beauties of this semantic integration is the increment of possibilities for query exploitation, in terms of simplicity and great potential to establish relations between data.

References

1. An, J., Geller, J., Wu, Y., Choon, S.: Semantic Deep Web: Automatic Attribute Extraction from the Deep Web Data Sources (2007) ACM 1-59593-480
2. Berners Lee, T., Fischetti, M.: Weaving the web, vol. 12. Harper, SanFrancisco (1999)
3. Sequeda, J., Tirmizi, S., Corcho, O., Miranker, D.: Survey of directly Mapping SQL databases to the Semantic Web. The Knowledge Engineering Review 26(4), 445–486 (2011)
4. W3C Semantic Web Activity, `http://www.w3.org/2001/sw/`
5. Doan, A., Halevy, A.: Semantic-Integration Research in the Database Community. AI Magazine 26(1) (2005)
6. Noy, N.F.: Semantic Integration: A Survey of Ontology-Based Approaches. SIGMOD Record 33(4) (December 2004)
7. Gruber, T.: A translation approach to portable ontology specifications. Knowledge Acquisition 5, 199–220 (1993)

8. Du Bois, B.: Towards an ontology of factors influencing reverse engineering. In: STEP 2005: Pro-ceedings of the 13th IEEE International Workshop on Software Technology and Engineering Practice, USA, pp. 74–80 (2005)
9. Organización Panamericana de la Salud, Oficina Regional de la Organización Mundial de la Salud, OMS/OPS Estrategia de Cooperación con el País México (Noviembre 2005), http://www.who.int/countryfocus/cooperation_strategy/ccs_mex_es.pdf
10. Astrova, I.: Reverse Engineering of Relational Databases to Ontologies. In: Bussler, C.J., Davies, J., Fensel, D., Studer, R. (eds.) ESWS 2004. LNCS, vol. 3053, pp. 327–341. Springer, Heidelberg (2004)
11. Bucella, A., Penabad, M.R., Rodriguez, F.J., Fariña, A.C., Cechich, A.: From relational databases to OWL ontologies. In: Proceedings of the 6th Russian Conference on Digital Libraries, Pushchino Rusia (2004)
12. Li, M., Du, X., Wang, S.: Learning ontology from relational database. In: Proceedings of the 4th international Conference on Machine Learning and Cybernetics, Guangzhou, China (2005)
13. Lubyte, L., Tessaris, S.: Automatic Extraction of Ontologies Wrapping Relational Data Sources. In: Bhowmick, S.S., Küng, J., Wagner, R. (eds.) DEXA 2009. LNCS, vol. 5690, pp. 128–142. Springer, Heidelberg (2009)
14. Stojanovic, L., Stojanovic, N., Volz, R.: A reverse engineering approach for migrating data intensive web sites to the Semantic Web. In: Proceedings of the IFIP 17th World Computer Congress – Tc12 Stream on Intelligent Information Processing, Deventer, The Netherlands (2002)
15. Kashyap, V.: Design and creation of ontologies for environmental information retrieval. In: 12th Workshop on Knowledge Acquisition Modeling and Management (KAW 1999), Banff, Canada (October 1999), http://sern.ucalgary.ca/ksi/kaw/KAW99/papers/Kashyap1/kashyap.pdf
16. Todorov, K., Geibel, P.: Ontology Mapping via structural and Instance-Based Similarity Measures. In: Third International Workshop on Ontology Matching, OM 2008 (2008)
17. O'Leary, D.: Different Firms, Different Ontologies, and No one Best Ontology. IEEE Intelligent Systems, 72–78 (September/October 2000)
18. W3C RDB2RDF Incubator Group (January 08, 2009), http://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF_SurveyReport.pdf
19. IDO developers, The infectious Disease Ontology, http://infectiousdiseaseontology.org/page/Main_Page (last access June 2012)

A Graph Cellular Automata Model to Study the Spreading of an Infectious Disease

Maria Jose Fresnadillo Martínez[1], Enrique García Merino[2],
Enrique García Sánchez[1], Jose Elias García Sánchez[1],
Angel Martín del Rey[3], and Gerardo Rodríguez Sánchez[4]

[1] Department of Preventive Medicine, Public Health and Clinic Microbiology
Faculty of Medicine, University of Salamanca
C/Alfonso X "El Sabio" sn, 37007-Salamanca, Spain
{jofrema,engarsan,joegas}@usal.es

[2] I.E.S. Politécnico de Soria, Soria, Spain
engarme@gmail.com

[3] Department of Applied Mathematics
E.P.S. de Ávila, Universidad de Salamanca
C/Hornos Caleros 50, 05003-Ávila, Spain
delrey@usal.es

[4] Department of Applied Mathematics
E.P.S. de Zamora, Universidad de Salamanca
Avda. Cardenal Cisneros 34, 49022-Zamora, Spain
gerardo@usal.es

Abstract. A mathematical model based on cellular automata on graphs to simulate a general epidemic spreading is presented in this paper. Specifically, it is a SIR-type model where the population is divided into susceptible, infected and recovered individuals.

1 Introduction

As is well known infectious diseases are those caused by pathogens (virus, bacteria, epiphytes) or parasites (protozoans, worms) and which can spread in the population. They have been an human enemy from time immemorial. Epidemics and pandemics can place sudden and intense demands on health systems: communicable diseases such as measles, influenza, tuberculosis, etc. are a common fact of modern life. Currently, they are events of concern and interest to many people worldwide: Remember epidemics such as Lyme diseases, toxic-shock syndrome, hepatitis C and E, AIDS, SARS, the Ebola virus, Avian Flu, and more recently the outbreak due to N1H1 virus. They can disrupt economic activity and development. The effects of high disease mortality on mean life span and of disease debilitation and mortality on the economy in afflicted countries are considerable.

As a consequence, the importance of understanding the dynamics and evolution of infectious diseases is steadily increasing in the contemporary world. The study, design and analysis of mathematical models to simulate epidemic spreading has a long history (see [9,19] although the crucial moment in the mathematical epidemiology was reached in 1927 when Kermack and McKendrick (see [13])

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 458–468, 2013.

introduced its famous model based on a system of ordinary differential equations to study the transmission of the Great Plague occurred in London from 1665 to 1666. It is the first compartmental model; the population is divided into different compartments or classes: susceptibles (individuals which are susceptible to the disease), infected (individuals which have been infected by the disease and are infectious) and recovered (individuals which are removed from infected compartment). Consequently, it is a SIR model where susceptible individuals are infected and the individuals leaving the infective compartment become immune, dead or removed by an isolation policy.

Since then, several mathematical models have been appeared in the literature (see [12,23] and references therein). The majority of mathematical models to simulate epidemic spreading are based on the use of differential equations (see, for example, [6] and references therein). Unfortunately, these models exhibit some important drawbacks since they do not take into account spatial factors such as population density, they neglect the local character of the spreading process, they do not include variable susceptibility of individuals, they cannot comprehensively depict complex contagion patterns (which are mostly caused by the human interaction induced by modern transportation), etc. As a consequence, this can lead to unrealistic results, such as, for example, endemic patterns relaying on very small densities of individuals, which are called "atto-foxes" or "nano-hawks" (see [16]).

Other mathematical models are based on a particular type of finite state machines called cellular automata. Cellular automata (CA for short) are simple models of computation capable to simulate physical, biological or environmental complex phenomena (see, for example, [22,25]).

CA were introduce by J. von Neumann and S. Ulam in the 50's and their motivation was to obtain a better formal understanding of biological systems that are composed of many identical objects that are relatively simple. The pattern evolution of a cellular automata is the result of the interactions of its objects. Cellular automata have been studied from a dynamical system perspective, from a logic, automata and language theoretic perspective and through ergodic theory.

Roughly speaking, a cellular automaton consists of a discrete spatial lattice of sites called cells, each one endowed at each time t with a state from a finite state set. The state of each cell is updated in discrete time steps according to a local transition function which depends on the states of the cells in some neighborhood around it. As the lattice is finite, some type of boundary conditions must be imposed. As is mentioned above, the usual topologies of CAs are chains and regular lattices; nevertheless, particular properties of two-dimensional lattice space exhibit some drawbacks: connection topology among the cells is restricted to predetermined homogeneous lattice, etc. As a consequence, although that is the standard paradigm for cellular automata, other topologies must be considered when the phenomenon to simulate therefore requires it. In this sense the topologies based on graphs are very important and useful: the nodes of the graph stand for the cells of the CA, and the neighborhood of a particular cell/node is constituted by the nodes adjacent to that one.

The CA-based models for epidemiological spreading eliminate the last mentioned shortcomings exhibited by the models based on ODEs, and are specially suitable for computer simulations. They have been used by several researches as an efficient alternative method to simulate epidemic spreading (see, for example, [1,3,5,7,8,17,18,20,24]), apart from another works appeared in the life sciences and computing literature). Of special interest are the CA-epidemic proposals modeling the motion of individuals (see, for example [2,4,14]). In the majority of these CA models the individuals are assumed to be distributed in the cellular space (defined as an homogeneous lattice) such that each cell stands for an individual of the population.

Here, we introduce a new mathematical model to simulate epidemic spreading. It is a SIR model and is based on cellular automata on graphs. In each cell several individuals are considered instead of only one individual, as is stated in the majority of proposals appeared in the literature. Consequently, each cell stands for a town or a city and its state is obtained from the fraction of the number of individuals which are susceptible, infected, or recovered from the disease.

The model introduced in this work can be considered as the continuation and improvement of the models shown in previous works of the authors. Specifically, in [10,11] a SIS and SIR models based on cellular automata endowed with the traditional topology (Von Neumann and Moore neighborhoods) was presented. Also, an improved SIS model was published in [15] considering topologies based on graphs. There are few works dealing with the use of cellular automata to simulate epidemic spreading considering each cell as an urban centre or a portion of land. Maybe the first paper was due to Sirakoulis, Karafyllidis and Thanailakis (see [21]) and in this work the basis concepts was stated although some drawbacks (related to the motion of individuals) was also presented.

The rest of the paper is organized as follows: In section 2 the basic theory about cellular automata on graphs is stated; The mathematical model to simulate the epidemic spreading is introduced in section 3. An illustrative simulation is presented in section 4, and finally, the conclusions and further work is presented in section 5.

2 Cellular Automata on Graphs

A graph G is a pair (V, E) where $V = \{v_1, v_2, \ldots, v_n\}$ is an ordered non-empty finite set of elements called nodes (or vertices), and E is a finite family of pairs of elements of V called edges. Two nodes of the graph, $v_i, v_j \in V$, are said to be adjacent (or neighbors) if there exists an edge in E of the form (v_i, v_j). We consider undirected graphs, that is, $(v_i, v_j) = (v_j, v_i) \in E$. A graph G is called simple if there is not two edges of G with the same ends and no loops exist, *i.e.* edges whose start and end is located at the same node.

If $V = \{v_1, \ldots, v_n\}$, the adjacency matrix of G is the $n \times n$ matrix, $A = (a_{ij})$, where

$$a_{ij} = \begin{cases} 1, \text{if } (v_i, v_j) \in E \\ 0, \text{if } (v_i, v_j) \notin E \end{cases}$$

As this work deals with undirected graphs, the adjacency matrix is symmetric.

The neighborhood of a node $v \in V$, N_v, is the set of all nodes of G which are adjacent to v, that is, $N_v = \{u \in V \text{ such that } (v, u) \in E\}$. The degree of a node v, d_v, is the number of its neighbors.

A cellular automaton on an undirected graph $G = (V, E)$ is a 4-tuple $\mathcal{A} = (V, S, N, f)$. The set V defines the cellular space of the CA such that each node stands for a cell the cellular automaton. S is the finite set of states that can be assumed by the nodes at each step of time. The state of the node v at time step t is denoted by $s_v^t \in S$, and it changes accordingly to the local transition function f. N is the neighborhood function which assigns to each node its neighborhood, that is:

$$\begin{aligned} N \colon V &\to 2^V \\ v_i &\mapsto N(v_i) = N_{v_i} = \left\{v_{i_1}, v_{i_2}, \ldots, v_{i_{d_v}}\right\} \end{aligned}$$

Finally, the local transition function f calculates the state of every node at a particular time step $t+1$ from the states of the its neighbors at the previous time step t, that is:

$$s_v^{t+1} = f\left(s_{v_{i_1}}^t, s_{v_{i_2}}^t, \ldots, s_{v_{i_{d_v}}}^t\right) \in S,$$

where $N_v = \left\{v_{i_1}, v_{i_2}, \ldots, v_{i_{d_v}}\right\}$.

3 The SIR Mathematical Model

In the mathematical epidemiological model introduced in this work the population is divided into three classes: those who are susceptible to the disease, those who are infected and those who have recovered and are immune to the disease. Moreover, the population is located at city centres which stand for the nodes of a graph G. If there is some type of transport connection between two of these cities, the associated nodes are connected by an edge. The following assumptions are also made:

1. The population of each node remains constant over time, that is, no births or deaths are taking into account. Moreover, the population distribution is inhomogeneous where P_u is the number of individuals of the node $u \in V$, and $P = \max\{P_u, u \in V\}$.
2. The transmission of the disease is through direct physical contact: touching an infected person, including sexual contact.
3. The population are able to move from its node to another one and return to the origin node at every step of time.

As the model introduced in this work is a SIR model, then the state of the node $u \in V$ at time step t is the triple $s_u^t = (S_u^t, I_u^t, R_u^t) \in Q \times Q \times Q = S$, where $S_u^t \in [0, 1]$ stands for the fraction of susceptible individuals of the node u at time t, $I_u^t \in [0, 1]$ stands for the fraction of infected individuals of the node u at time

t, and $R_u^t \in [0,1]$ stands for the fraction of recovered individuals of the node u at time step t. Consequently, the transition function of the CA is as follows:

$$\begin{aligned} s_u^t &= f\left(s_{v_1}^{t-1}, \ldots, s_{v_{d_u}}^{t-1}\right) = \left(S_u^t, I_u^t, R_u^t\right) \\ &= \left(\left(d \circ f_S\right)\left(s_{v_1}^{t-1}, \ldots, s_{v_{\gamma_u}}^{t-1}\right), \left(d \circ f_I\right)\left(s_{v_1}^{t-1}, \ldots, s_{v_{\gamma_u}}^{t-1}\right), \left(d \circ f_R\right)\left(s_{v_1}^{t-1}, \ldots, s_{v_{\gamma_u}}^{t-1}\right)\right) \end{aligned}$$

The ground where the epidemic is spreading is modeled as a weighted graph where each node stands for a city or a town, and the arc between two nodes represents the connection between the corresponding cities. In this sense, the connection factor between the nodes u and v is the weight associated to the edge $(u,v) \in E$ and it is denoted by w_{uv}. It depends on the transportation capacity of the public and non-public transport; Consequently:

$$w_{uv} = \frac{h_{uv}}{\max\left\{h_{xy}, \forall x, y \in V\right\}} \in [0,1],$$

where h_{uv} is the total amount of population wich move from u to v during a time step.

The infected individuals of u at time step t is given by the sum of the following terms:

- The infected individuals at the previous time step which have not been recovered.
- The susceptible individuals which have been infected during the time step. In this case we have to take into account the recovery rate $r \in [0,1]$. These new sick individuals of u can be infected both by the infected individuals of u or by the infected individuals of the neighbor nodes of u which have moved to u during the time step. In the first case, only the rate of transmission, $p \in [0,1]$, is involved, whereas in the second case we have to consider the connection factors between the nodes, and the population and movement factor of each node. Moreover we also consider the susceptible individuals of u moved to a neightbor node during the step of time and infected in this neighbor node by its corresponding infected individuals; in this case $\eta_u \in [0,1]$ yields the portion of moved susceptible individuals from u to its neighbor nodes. Note that $\sum_{v \in V_u} \eta_{uv} = \eta_u$.

As a consequence the mean-field equation for infected individuals is the following:

$$\begin{aligned} f_I\left(s_{v_1}^{t-1}, \ldots, s_{v_{\gamma_u}}^{t-1}\right) = {} & (1-r)\, I_u^{t-1} + p\,(1-\eta_u)\, S_u^{t-1} I_u^{t-1} \\ & + p\,(1-\eta_u)\, S_u^{t-1} \sum_{v \in V_u} \frac{P_v}{P} w_{vu} I_v^{t-1} \\ & + p S_u^{t-1} \sum_{v \in V_u} (1 - w_{vu})\, \eta_{uv} I_v^{t-1}. \end{aligned} \tag{1}$$

On the other hand, the susceptible individuals of each node is given by the difference of the susceptible individuals of the node at the previous time step and the susceptible individuals which have been infected as is mentioned above. As a consequence, the following equation holds:

$$\begin{aligned} f_S\left(s_{v_1}^{t-1},\ldots,s_{v_{\gamma_u}}^{t-1}\right) &= S_u^{t-1} - p\left(1-\eta_u\right)S_u^{t-1}I_u^{t-1} \\ &- p\left(1-\eta_u\right)S_u^{t-1}\sum_{v\in V_u}\frac{P_v}{P}w_{vu}I_v^{t-1} \\ &- pS_u^{t-1}\sum_{v\in V_u}\left(1-w_{vu}\right)\eta_{uv}I_v^{t-1}. \end{aligned} \tag{2}$$

Finally, the recovered individuals of a node at a particular time step is given by the recovered individuals at the previous time step plus the infected individuals which have been recovered during the time step, that is:

$$f_R\left(s_{v_1}^{t-1},\ldots,s_{v_{\gamma_u}}^{t-1}\right) = R_u^{t-1} + rI_u^{t-1}. \tag{3}$$

Note that, as a simple calculus shows:

$$I_u^t + S_u^t + R_u^t = I_u^{t-1} + S_u^{t-1} + R_u^{t-1} = P_u, \tag{4}$$

and consequently equation (3) can be substitute for the following equation:

$$f_R\left(s_{v_1}^{t-1},\ldots,s_{v_{\gamma_u}}^{t-1}\right) = 1 - I_u^t - S_u^t. \tag{5}$$

Moreover, as

$$\begin{aligned} f_S\left(s_{v_1}^{t-1},\ldots,s_{v_{\gamma_u}}^{t-1}\right) &\in [0,1], \\ f_I\left(s_{v_1}^{t-1},\ldots,s_{v_{\gamma_u}}^{t-1}\right) &\in [0,1] \end{aligned} \tag{6}$$

and $f_R\left(s_{v_1}^{t-1},\ldots,s_{v_{\gamma_u}}^{t-1}\right) \in [0,1]$, then a discretization function $d\colon [0,1] \to Q$ must be used in order to get a finite state set. In our case, the discretization function used is the following:

$$\begin{aligned} d\colon\ [0,1] &\longrightarrow Q \qquad (7) \\ x &\longmapsto d(x) = \frac{[100\cdot x]}{100} \end{aligned}$$

where $[m]$ stands for the nearest integer to m. As a consequence,

$$Q = \{0, 0.01, 0.02, \ldots, 0.99, 1\}. \tag{8}$$

4 An Illustrative Simulation

In this simulation we will suppose that the epidemic is spreading over n cities forming a complete graph K_n. A complete graph is a graph in which each pair of graph nodes is connected by an edge (that is, each city is connected with each others). The complete graph with n nodes is denoted by K_n and has $\frac{n(n-1)}{2}$ edges. The adjacency matrix A of the complete graph K_n takes the particularly simple form of all 1s with 0s on the diagonal.

For the sake of simplicity this example deals with the complete graph K_6, that is, only $n = 6$ cities are involved in the spreading of the epidemic: $u_1, \ldots, u_6$. In Figure 1 the graph topology of this example is shown.

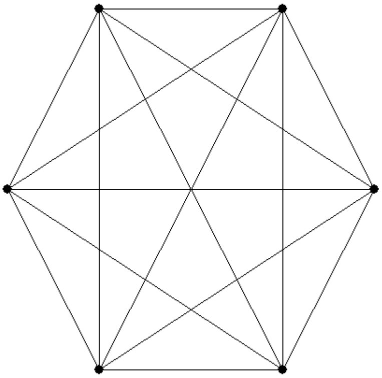


Fig. 1. Complete graph K_6

Moreover, the parameters used in this example are merely illustrative and they do not correspond to a particular infectious disease. We will consider the following initial configuration:

$$S_{u_1}^0 = 0.8, I_{u_1}^0 = 0.2, R_{u_1}^0 = 0,$$
$$S_{u_i}^0 = 1, I_{u_i}^0 = R_{u_i}^0 = 0, \ 2 \leq i \leq n.$$

That is, there is only one node at time $t = 0$ with infected population. Moreover, the parameters used are:

$$p = 0.5, r = 0.6,$$
$$\eta_{u_i} = 0.25, \ 1 \leq i \leq 6 \text{ with } \eta_{u_i v} = 0.05 \ \forall v \in N_{u_i}.$$

Note that it is assume that $\eta_{u_i v} = \eta_{u_i}/d_{u_i}$ for each i. Moreover, let us suppose that the population of each node is the same: $P_{u_i} = 100$ with $1 \leq i \leq 6$, and also the transport capacity between two nodes is the same: $w_{u_i u_j} = 1$ for $1 \leq i, j \leq 6$. Note that this example deals with an homogeneous-symmetric case.

In Figure 2, the evolution of the number of susceptible, infected and recovered individuals is shown.

In Table 1, the necessary conditions for epidemic spreading from a single node u to a neighbor node v are shown. In the first, second and third column some different and arbitrary values of the parameters p, η_u and w_u are taken. In the

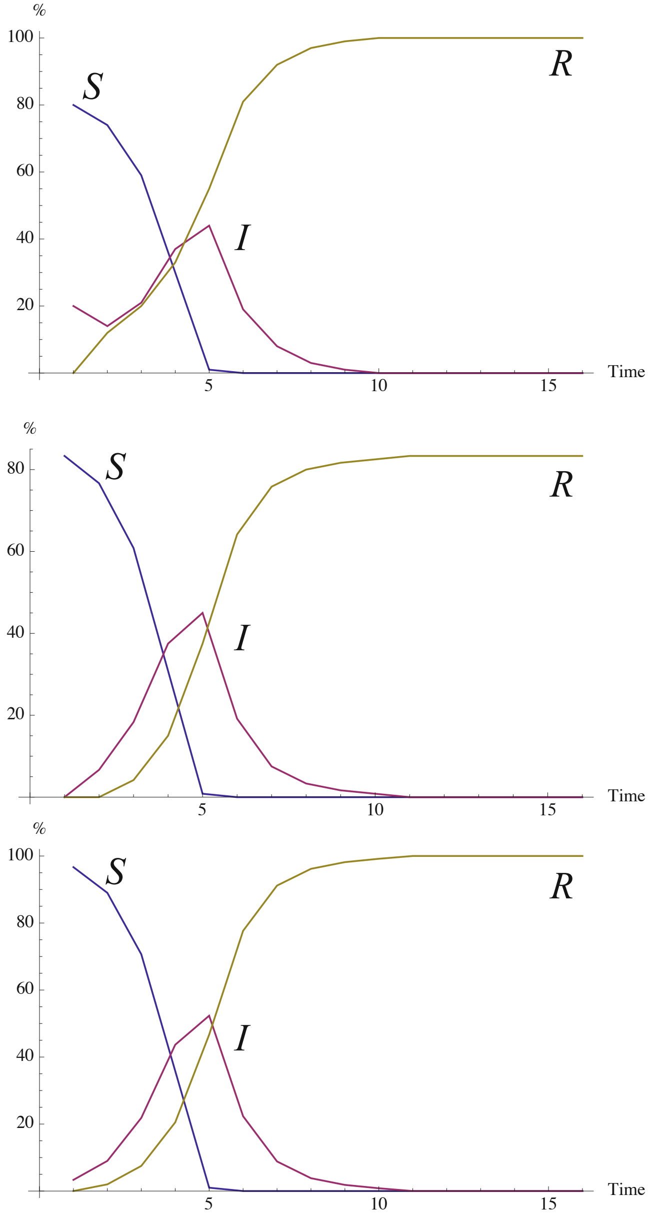


Fig. 2. Top: Evolution of the proportion of the susceptible, infected and recovered population in the node u_1. Middle: Evolution of the susceptible, infected and recovered population in the nodes $u_2, \ldots, u_6$. Bottom: Evolution of total number of susceptible, infected and recovered individuals.

fourth column, the minimun state of the node u to produce epidemic spreading is shown, and finally in the fifth column the state of the neighbor node (when the spreading occurs) at the following state of time is given.

Table 1. Necessary conditions for epidemic spreading in the case of K_6

p	η_u	w_u	I_u^0	I_v^1
0.25	0.25	0.5	0.84	0.08
0.25	0.25	1	0.53	0.1
0.25	0.5	1.	0.8	0.1
0.5	0.25	0.5	0.47	0.09
0.5	0.25	1	0.27	1
0.5	0.5	0.5	0.67	0.09
0.5	0.5	1	0.4	0.1
0.5	0.75	1	0.8	0.1
0.75	0.25	0.5	0.33	1
0.75	0.25	1	0.18	0.1
0.75	0.5	0.5	0.47	1
0.75	0.5	1	0.27	0.1
0.75	0.75	0.5	0.84	0.09
0.75	0.75	1	0.53	0.1
1	0.25	0.5	0.25	0.1
1	0.25	1	0.13	0.1
1	0.5	0.5	0.36	0.1
1	0.5	1	0.2	0.1
1	0.75	0.5	0.67	0.1
1	0.75	1	0.4	0.1

5 Conclusions

In this work a new SIR-epidemiological model based on cellular automata on graphs has been proposed. The main characteristics of this model are the following:

- Each node of the graph stand for a group of individuals placed on a city or town.
- These individuals are classified into three compartments: susceptible, infected and recovered. As a consequence, the state of the each node at a particular time step is the 3-tupla formed by the portion of susceptible, infected and recovered individuals at this time.
- It is suppose that the transmission of the disease is through direct physical contact between an infected and a susceptible individual.
- The population is able to move from a node to another one.
- The local transition function of the cellular automata is non-linear and it involves the following parameters: the recovery rate, the rate of transmission, the movement factor for susceptible individuals, the connection factor between the nodes and the population of each node.

The laboratory simulations obtained seem to be in agreement with the expected behavior of a real epidemic.

Future work will aim to extend the paradigm presented in this work to other compartmental models as SIRS, SEIR, etc. Moreover, the study of the introduction in the model of new parameters and vaccination effect must be taken into account.

Acknowledgements. This work has been supported by Fundación "Memoria D. Samuel Solórzano Barruso" (University of Salamanca, Spain) under grant YA7Y / 463AC06.

References

1. Ahmed, E., Agiza, H.N.: On modelling epidemics including latency, incubation and variable susceptibility. Physica A 253, 247–352 (1998)
2. Ahmed, E., Elgazzar, A.S.: On some applications of cellular automata. Physica A 296, 529–538 (2001)
3. Beauchemin, C., Samuel, J., Tuszynski, J.: A simple cellular automaton model for influenza A viral infections. J. Theor. Biol. 232, 223–234 (2005)
4. Boccara, N., Cheong, K.: Critical behavior of a probablistic automata network SIS model for the spread of an infectious disease in a population of moving individuals. J. Phys. A-Math. Gen. 26, 3707–3717 (1993)
5. Boccara, N., Cheong, K., Oram, M.: A probabilistic automata network epidemic model with birds and deaths exhibiting cyclic behaviour. J. Phys A-Math. Gen. 27, 1585–1597 (1994)
6. Diekmann, O., Heesterbeek, J.O.P.: Mathematical Epidemiology of Infectious Diseases: Model Building, Analysis and Interpretation. Wiley (2000)
7. Fuentes, M.A., Kuperman, M.N.: Cellular automata and epidemiological models with spatial dependence. Physica A 267, 471–486 (1999)
8. Fuks, H., Lawniczak, A.T.: Individual-based lattice model for spatial spread of epidemics. Discrete Dyn. Nat. Soc. 6, 191–200 (2001)
9. Hamer, W.H.: Epidemic disease in England. Lancet 1, 733–739 (1906)
10. Martín del Rey, A., Hoya White, S., Rodríguez Sánchez, G.: A Model Based on Cellular Automata to Simulate Epidemic Diseases. In: El Yacoubi, S., Chopard, B., Bandini, S. (eds.) ACRI 2006. LNCS, vol. 4173, pp. 304–310. Springer, Heidelberg (2006)
11. Hoya White, S., Martín del Rey, A., Rodríguez Sánchez, G.: Modeling epidemics using cellular automata. Appl. Math. Comput. 186, 193–202 (2007)
12. Keeling, M.J., Rohani, P.: Modeling Infectious Diseases in Humans and Animals. Princeton University Press (2008)
13. Kermack, W.O., McKendrick, A.G.: Contributions to the mathematical theory of epidemics, part I. Proc. Roy. Soc. Edin. A 115, 700–721 (1927)
14. Mansilla, R., Gutierrez, J.L.: Deterministic site exchange cellular automata model for the spread of diseases in human settlements. Complex Systems 13, 2 (2001)
15. Fresnadillo, M.J., García, E., García, J.E., Martín, Á., Rodríguez, G.: A SIS Epidemiological Model Based on Cellular Automata on Graphs. In: Omatu, S., Rocha, M.P., Bravo, J., Fernández, F., Corchado, E., Bustillo, A., Corchado, J.M. (eds.) IWANN 2009, Part I. LNCS, vol. 5518, pp. 1055–1062. Springer, Heidelberg (2009)

16. Molisson, D.: The dependence of epidemic and population velocities on basic parameters. Math. Biosci. 107, 255–287 (1991)
17. Quan-Xing, L., Zhen, J.: Cellular automata modeling of SEIRS. Chinese Phys. 14, 1370–1377 (2005)
18. Ramani, A., Carstea, A.S., Willox, R., Grammaticos, G.: Oscillating epidemics: a discrete-time model. Physica A 333, 278–292 (2004)
19. Ross, R.: The prevention of malaria, 2nd edn. Murray (1911)
20. Rousseau, G., Giorgini, R., Livi, H., Chaté, H.: Dynamical phases in a cellular automaton model for epidemic propagation. Physica D 103, 554–563 (1997)
21. Sirakoulis, G.C., Karafyllidis, I., Thanailakis, A.: A cellular automaton model for the effects of population movement and vaccination on epidemic propagation. Ecol. Model. 133, 209–223 (2000)
22. Toffoli, T., Margolus, N.: Cellular Automata Machines: A New Environment for Modeling. The MIT Press (1987)
23. Vynnycky, E., White, R.G.: An Introduction to Infectious Disease Modelling. Oxford University Press (2010)
24. Willox, R., Grammaticos, B., Carstea, A.S., Ramani, A.: Epidemic dynamics: discrete-time and cellular automaton models. Physica A 328, 13–22 (2003)
25. Wolfram, S.: A New Kind of Science. Wolfram Media Inc. (2002)

Modelling Interpersonal Relations in Surgical Teams with Fuzzy Logic

Andrzej Romanowski, Pawel Wozniak, Tomasz Jaworski,
Pawel Fiderek, and Jacek Kucharski

Institute of Applied Computer Science, Lodz University of Technology
androm@kis.p.lodz.pl
http://ubicomp.pl

Abstract. This paper reports on the design and development of a novel interactive system for managing this cooperation between remote wards in a children's hospital. This work focuses on the design of a decision-making system that aids in assembling surgical teams. Surgical team rosters were modelled using fuzzy inference methods in order to include existing personal preferences and to achieve greater efficiency in using hospital's resources. The design of the system is based on user studies and simulations were performed to test the feasibility of the solution. In this interdisciplinary endeavour, the research team combined knowledge from the fields of interaction design, ubiquitous computing, operations research and fuzzy modelling to tackle a real-life problem in a medical safety-critical setting.

1 Introduction

The research described in this paper is a product of a cooperation scheme with one of Poland's leading institutions in the field of prenatal cardiology, a very young and rapidly developing field of medical science. The research team has spent a considerable amount of time with the physicians at the Polish Mothers Memorial Hospital (PMMH) discussing their daily practices and identifying their needs in terms of IT.

Several times a month, during an ultrasound or echocardiography examination, an extraordinary procedure is invoked in one of the hospital's diagnosis units. This action may require at least three wards in two buildings to cooperate, the details of previous work on improving the process are available in [1].

This paper focuses on solving a related problem of choosing the proper personnel for a given procedure, especially in a case where surgeons of different specialities work towards a common goal. User studies conducted by the research team give reason to suspect that the automation of the surgical team scheduling process and optimising operating theatre occupancy may significantly contribute to increasing the operational efficiency of the unit under investigation.

I. Batyrshin and M. González Mendoza (Eds.): MICAI 2012, Part I, LNAI 7629, pp. 469–479, 2013.

2 Problem Background

2.1 Infrastructural Obstacles to Seamless Cooperation

The PMMH is a facility of considerable size consisting mainly of the paediatric hospital and the maternity hospital. These are located in separate buildings and connected with an underground tunnel (See Fig. 1). The physical distance between the two locations is the first obstacle on the road to seamless cooperation. The other barriers are of administrative and financial nature.

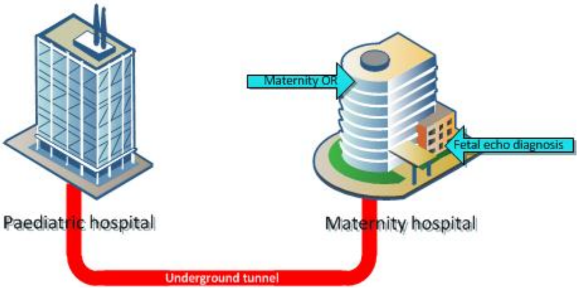


Fig. 1. Placement of the key divisions within the hospital

2.2 Assembling Surgical Teams

Currently, there are no dedicated means of communication between the three institutions involved that would help managing interinstitutional involvement in surgeries. There is no system in place that would clearly indicate that an exceptional case is in progress and the communication is prone to the usual technical difficulties connected with using phones. The vast experience of the physicians and nurses is the most important factor as virtually no technical assistance is provided for the task of promptly assembling the required medical team. The personnel at the hospital use their exceptional organisational skills and the safety of the patient is always assured, yet this part of the process can be easily identified as an area that can be improved.

2.3 Related Work

Optimising hospital resources and perfecting clinical schedules is a well-analysed problem and several approaches have been used. Hospitals are generally considered a safety-critical environment and clinical systems are usually subject to evaluation limitations [2]. Several theoretical models have been developed. Usually this works focus on: limiting costs and optimising resources in long-term planning [3], predicting surgery times [4] or improving schedules [5].

Context-aware applications along with pervasive computing can benefit significantly from applying artificial intelligence methods. Among others, fuzzy set theory and fuzzy logic can be used as a tool for representing and processing an imprecise contextual information [6]. Nowadays, fuzzy techniques are an inherent element of ambient intelligence systems [7] and situation-aware systems [8], [9]. Moreover, some recent works (*e.g.* [10], [11] and [12]), prove that the fuzzy set theory and fuzzy logic may be a team building tool. There exists a story of applying fuzzy inference in medical environments [13].

2.4 User Study Methodology

The information on the day-to-day operations in the PMMH and other units of interest was obtained through working closely with the doctors through extensive needfinding. The research team conducted numerous interviews, shadowing sessions and panel discussions to identify the shortcomings of the present IT solutions and find areas for potential improvement. The prototype design takes the physicians' input into account as the doctors have been involved in the process of creating the solution.

3 Overall Solution Description

The solution facilitates notifying the institutions, assembling medical teams and exchanging opinions between physicians. To eliminate time loss, it was decided to use context-aware technologies as their application in medical environments may be highly beneficial to the medical staff [14]. As shown in [15] activity management combined with interface familiarity may play a crucial role in this case. Furthermore, an inference system that aids assembling surgical teams is in place.

The system itself is a quite straightforward design, largely limited due to our determination to minimise the interference with the hospital's regular operations. The overall infrastructure with reference to the hospital room can be seen in Fig. 2.

3.1 Improvements

A prototype system has been implemented and preliminary tests have been conducted. The system is to make several improvements in the clinical process.

Notifications and Communication. The first area for improvement was notifying the doctors of the emergency in progress. The system can be used to receive the notification of an emergency and, instead of issuing a ward-wide alarm, the nurses can quickly assemble a surgical team and notify the squad by automated text messages (as per request of the medical staff). The process is aided by the fuzzy inference system.

Bluetooth proximity enables the surgeons to continue with their ward duties and only refer to the touchscreens when new information is available.

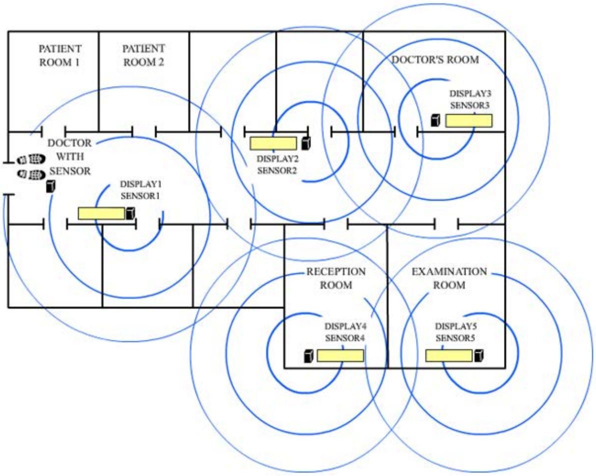


Fig. 2. Context-aware infrastructure distributed in the clinical environment

Sharing Imaging Data. Bluetooth proximity is also the key to how imaging data is shared within the system. The surgeon scheduled to perform is only required to approach the touchscreen to receive all the available data. Having been notified by the system, the surgeon may review the streamed images at the screen and discuss the process with the examiner.

Scheduling Surgeries and Managing Surgical Teams. The implemented system includes a tool for aiding ward management by assembling surgical teams and managing operating theatre schedules. This paper features a thorough description of the solution as it uses fuzzy inference for assuring that proper personnel is selected at appropriate times.

4 A System for Choosing the Proper Surgical Team

Fig. 3 presents a functional diagram of the surgical team manager software. This tool helps choosing surgical team rosters, taking several factors into account, *e.g.* issues related to procedure complexity or time limitations.

Moreover, the module is designed to record the teams' performances in surgery, so storing new data in the knowledge base. A feedback loop is included in the design for future improvement of the system based on interpersonal relation modifications inferred from past surgery data.

4.1 Description of the Decision Process

The system consists of several decision blocks which utilise both crisp rules and fuzzy inference. The overall composition can be seen in Fig. 3. In each step, the

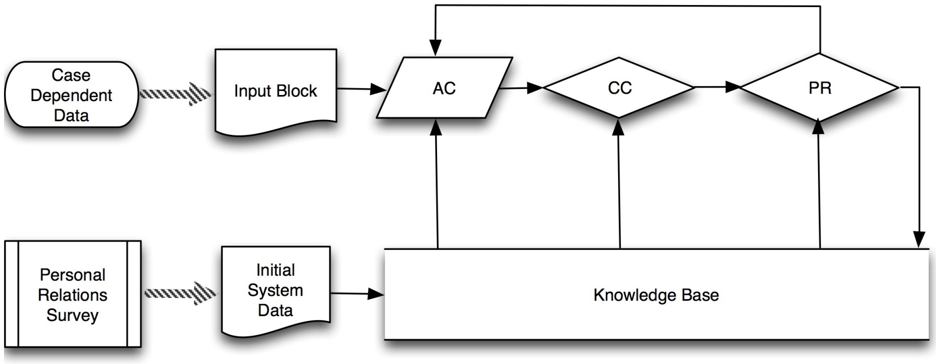


Fig. 3. An overview diagram of the surgical team manager. AC — availability check, CC — choice of consultant, PR — personal relations.

potential staff pool is reduced with a complete surgical team at the end of the process. The individual blocks are described below.

The system is provided with direct, case-specific input from the head nurse. It also contains information on the specific conditions of the ward (*e.g.* maximum operating capacities) and the current schedule of both surgeon and operating theatres. The initial input includes the type of the surgery, the severity of the case and any special circumstances.

AC (availability check, crisp logic) is a decision step where the entire staff pool is analysed for minimum required skills and current availability. The teams must be chosen from doctors and staff who are not performing other surgeries and not on holiday. In this manner, the staff pool is limited to those members whose participation requires advanced decision making.

In *CC* (choice of consultant, fuzzy logic), the system limits the potential staff pool by comparing the complexity of the procedure with the experience of the staff thus assuring proper supervision by a consultant. The knowledge base is consulted for past success rates and current work loads in order to find the most suitable lead surgeon.

PR (personal relations, fuzzy logic) is responsible for making inferences based on the personal relation between the staff. The method used is described further in this chapter and it is the core issue addressed in this paper.

When a satisfactory surgical team is assembled, the suggestion is communicated to the user and the personnel can be notified by means of SMS messages or fixed-line telephony. A feedback loop is used in the design since the head nurse may want to alter the squad of the teams or the reasoning system may produce unsatisfactory results in the first instantiation (*e.g.* a fully competent team is assembled, yet there is a known rivalry between the chief surgeon and the chosen assistant). A given team member is eliminated form the process (by means of choosing the one whose schedule is most busy or experience most limited) and the reasoning procedure is repeated.

4.2 Modelling the Staff's Personal Preferences with Fuzzy Logic

Due to the inherently approximate nature of personal relationships within surgical team, teh authors suggest using an approch that provides a framework for approximate reasoning — fuzzy logic [16]. In case of the system under consideration, degrees of truth are used to describe social relations between the potential members of the surgical team. For instance, let us assume that a social relation matrix is available for almost all potential team members. Such a matrix would contain a set of values that one could interpret as a personal preference relation between persons A and B, defined as $LIKE(A, B)$ with the following properties:

- if person A does not know person B then

$$LIKE(A, B) = 0 \tag{1}$$

- personal preferences are not always reciprocated and may be asymmetric

$$\exists(A, B \in Personnel) : LIKE(A, B) \neq LIKE(B, A) \tag{2}$$

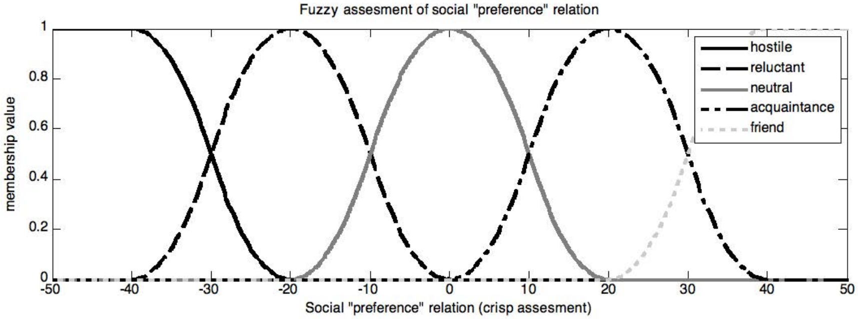


Fig. 4. The fuzzy variable $liking_F$ and its labels (linguistic variables)

A sample social preference relation matrix for an anonymised personnel group is presented in Table 1. This example assumes that the relation is defined on range from -50 (profound hostility) to 50 (declared friendship).

It is worth noting that the values contained within the table where obtained using a single-survey study. One may try obtain more precise values using data mining techniques on past success rates or apply more complex psychological analyses. This discussion, however, is beyond the scope of this article. To utilize the fuzzy logic approach, a the fuzzy variable $liking_F$ is defined with reference to the fuzzy sets (labels), depicted in Fig. 4.

At this point, every crisp social preference relation is expressed in terms of $liking_F$ (a fuzzy description). The following form is used:

$$liking^F(l) = \{\mu_{friend}(l); \mu_{acquaintance}(l); \mu_{neutral}(l); \mu_{reluctance}(l); \mu_{hostility}(l)\} \tag{3}$$

Table 1. The social preference matrix for potential surgical team members

		Brenda	Amias	Caryl	Rosalin	Arn	Eddy	Emma	Zaria	Zoey	Radcliff	Davey	Mike
Surgeons	Brenda		45	-25	28	0	30	15	35	3	9	15	25
	Amias	35		10	30	30	15	30	35	30	20	0	35
	Caryl	-30	40		0	35	20	25	-10	35	5	30	-15
Anesth.	Rosalin	25	35	-15			0	0	0	25	15	10	0
	Arn	-10	40	30			0	0	0	5	-10	0	35
Assistants	Eddy	35	10	35	-10	0		10	15				
	Emma	15	30	20	0	0	15		20				
	Zaria	35	35	5	30	20	10	0					
Staff	Zoey	25	15	10	5	5					30	15	30
	Radcliff	5	-10	0	15	0				25		-5	15
	Davey	0	0	4	0	0				0	0		30
	Mike	25	35	-15	0	35				30	15	30	

where μ_N is a characteristic function (*i.e.* the membership function) of the fuzzy set N and l is the $LIKE$ relation from Table 1.. With such a definition, every relation $LIKE(\circ)$ has its individual fuzzy description $liking^F$ which consists of 5 values.

For instance, let us consider two surgeons (Caryl and Amias) with defined social preference relations $LIKE(Caryl, Amias) = 40$ and $LIKE(Amias, Caryl) = 10$. On the basis of Eq. 3 and Fig. 4, one can infer that Caryl considers Amias more a friend ($\mu_{friend} = 0.6$; $\mu_{colleague} = 0.4$) than an acquaintance, however Amias is rather neutral to Caryl ($\mu_{neutral} = 0.6; \mu_{acquaintance} = 0.4$).

When every personal preference relation in Table 1 can be expressed with fuzzy logic, a measure of the surgical team's *cooperative capability* can be derived. The measure the authors used is:

$$G^F(T) = \bigvee_{(A,B) \in P(T)} mutual^F_{liking^F}(A,B) \tag{4}$$

where T denotes the surgical team, (A, B) are pairs of team members, $P(T)$ is the set of all team member pairs of T. The fuzzy operator $\wedge$ was implemented in a classical way — by using max. Similarly, the operator $\vee$ (used below) utilises min. In Eq. 4 the cooparative capability of the team is expressed as a fuzzy sum of the relation $liking^F$. The new operator $mutual^F_R$ defined in Eq. 5 can be interpreted as a mutualisation of the relation R between persons A and B. The mutualisation is accomplished by means of selecting the most pessimistic version of the personal preference relation $liking^F$. In:

$$mutual^F_{liking^F}(A,B) = liking^F(min(LIKE(A,B), LIKE(B,A))) \tag{5}$$

min is not fuzzy operator, but an implementation of the worst case of the *liking* relation. For Caryl and Amias (see Table 1), $mutual^F$ will be equal to:

$$mutual^F_{liking^F}(Caryl, Amias) = liking^F(10) = \left\{ \begin{array}{l} \mu_{friend} = 0; \\ \mu_{acquaintance} = 0.5; \\ \mu_{neutral} = 0.5; \\ \mu_{reluctance} = 0 \\ \mu_{hostility} = 0 \end{array} \right\} \quad (6)$$

The approach can be further extended by introducing weights to the relations between particular team members. A sample solution is presented in Fig. 5. It can be clearly seen that the personal relations between the lead surgeon and the practitioner is key while less importance is assigned to the personal preferences of the assisting staff. The presented weight are based on user studies performed by the authors. Eq. 4 can be extended with the use of relation weights by using:

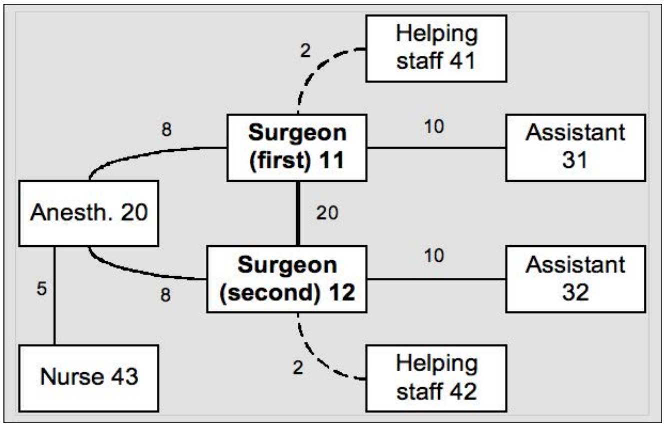


Fig. 5. Sample weight distribution of personal preference relations in a surgical team

$$G^F(T) = \left\{ \underset{L \in LABELS(liking^F)}{\mu_L} : \frac{\sum_{(A,B)\in P(T)} W(A,B)\mu_{mutual^F_{liking^F}(A,B),L}}{\sum_{(A,B)\in P(T)} W(A,B)} \right\} \quad (7)$$

where W(A,B) is the weight of the personal preference relation between A and B, assuming that $W(A,B) = W(B,A)$. Eq. 7 is a weighted average of the importance of the personal preference relation with weight defined in Fig. 5.

Finally, in order to obtain a scalar measure of the cooperative capability of the team the fuzzy quantity $G^F(T)$ must be defuzzyfied. The authors suggest using the Center of Gravity method for performing this task.

$$G(T) = \frac{\sum_x \bigvee_{L \in LABELS(G^F(T))} \mu_{G^F(T),L}(x)x}{\sum_x \bigvee_{L \in LABELS(G^F(T))} \mu_{G^F(T),L}(x)} \quad (8)$$

The value obtained as a result of Eq. 8 will belong to the same interval as the values of the *liking* relation in Table 1. The lower the measure of the operational capability $G(T)$ of the team, the lower the probability of assembling such a team in practice should be.

5 Simulation Results and Discussion

5.1 Evaluation

The authors performed three types of numerical simulations to evaluate the feasibility of the system. The experiment uses the following assumptions:

- A surgical team consists of 8 members — 2 surgeons, 1 anaesthesiologist, 2 assisting nurses, 3 members of technical staff on standby
- The available staff pool consists of 12 people

Using elementary combinatorics the total number of 1728 possible teams was computed. All of these rosters were evaluated using the suggested reasoning techniques. Table 2 presents the outcomes of the simulation. By analysing the

Table 2. An overview of simulations results for an eight-member surgical team chosen from a staff pool of 12 in terms of the operational efficiency $G(T)$. LS — lead surgeon, SS — secondary surgeon, AN — anaesthesiologist, A1, A2— surgical assistants, S1, S2, S3 — assisting staff.

$G(T)$	Team count	Two sample surgical teams for a given $G(T)$							
		LS	SS	AN	A1	A2	S1	S2	S3
34.26	6	Amias	Caryl	Arn	Zaria	Eddy	Zoe	Davey	Mike
		Caryl	Amias	Arn	Eddy	Zaria	Davey	Radcliff	Mike
32.52	6	Brenda	Amias	Rosalin	Eddy	Emma	Mike	Radcliff	Zoey
		Amias	Brenda	Rosalin	Emma	Zaria	Radcliff	Mike	Zony
32.14	12	Brenda	Amias	Rosalin	Eddy	Zaria	Davey	Radcliff	Zoey
		Amias	Brenda	Rosalin	Emma	Zaria	Radcliff	Davey	Zoey
31.59	24	Brenda	Amias	Rosalin	Eddy	Emma	Davey	Mike	Radcliff
		Brenda	Amias	Rosalin	Eddy	Emma	Mike	Zoey	Radcliff
31.16	4	Amias	Caryl	Rosalin	Zaria	Eddy	Mike	Davey	Zoey
		Caryl	Amias	Rosalin	Eddy	Zaria	Davey	Radcliff	Zoey
-3.43	36	Brenda	Caryl	Arn	Eddy	Zaria	Davey	Mike	Radcliff
		Caryl	Brenda	Arn	Zaria	Eddy	Zoey	Radcliff	Davey
-3.35	12	Brenda	Caryl	Arn	Eddy	Zaria	Mike	Davey	Zoey
		Brenda	Caryl	Arn	Eddy	Zaria	Mike	Davey	Radcliff
-2.34	20	Caryl	Brenda	Rosalin	Eddy	Emma	Davey	Zoey	Radcliff
		Caryl	Brenda	Rosalin	Eddy	Emma	Davey	Radcliff	Zoey
-2.33	4	Brenda	Caryl	Rosalin	Emma	Eddy	Davey	Radcliff	Zoey
		Brenda	Caryl	Rosalin	Emma	Eddy	Mike	Radcliff	Zoey
-2.32	12	Brenda	Caryl	Rosalin	Emma	Eddy	Radcliff	Mike	Davey
		Caryl	Brenda	Rosalin	Eddy	Emma	Radcliff	Mike	Davey

results of the simulation, the relationship matrix (Table 1) and the weight distributions (Fig. 5), it can be observed that the surgical staff can be classified into two borderline cases. The first case concerns team variants where Caryl and Brenda are fond of Amias and the feelings are reciprocated. The second case involves the hostile relationship between Caryl and Brenda.

Even though such situations are clearly undesirable in a hospital setting, they play a crucial role in the algorithm and largely affect the operational efficiency $G(T)$. Table 2 shows that teams containing the Amias-Caryl and Amias-Brenda pairs received best grades, while those that include the Caryl-Brenda duo are deemed least appropriate. The two surgeons are at the core of the team and this fact translates to the central value of this pair when calcuating operational efficiency. Other personnel have limited influence on the final $G(T)$. In several cases, the precise roles of team members do not affect the overall grade (*e.g.* there are 24 teams with ($G(T) = 31.59434$). In such cases, scheduling may be deciding factor or the teams can be chosen at random.

5.2 Discussion

While evaluating the interactive system is a risky and complicated endeavour in a hospital environment, several components of the solution can be safely tested. A further investigation on how data sharing and notifications work in a clinical environment is still in progress, but the surgery team management solution was evaluated using generated mock sets based on real-life data. Furthermore, a mock classification of surgery types and severities was created. The simulated performance of the system was satisfactory with personal conflicts being avoided at all times.

The possibility of introducing the solution to the clinical environments raises several questions. Firstly, test subjects have reported that the task of providing the system with data on the procedure to be performed was a particularly complex task. Furthermore, the personnel responsible for assembling the surgical team is often unable to assess the severity of the case. Consequently, additional input from a physician is required. This, in turn, may contribute to decreasing the operational efficiency of the unit.

6 Conclusions

This paper introduced a system that aids in assembling surgical teams. The key contribution is the idea of including the social relationships between surgical staff in the decision process. This is facilitated by the application fuzzy logic machine reasoning to model the interpersonal relations within a surgical clinic.

The outcomes of the project are a product of extensive cooperation with doctors for one of Poland's largest maternity hospitals. The problem of choosing proper personnel for surgical procedures has a particular impact on the wards' operational efficiency on a daily basis.

The suggested system design address these issues and provides a convenient automated tool for use in medical work environments. The feasibility of the solution is proven by simulation experiments.

It is believed that, in the proximate future, the knowledge base of the system will be extended as result of the ongoing process of digitising medical records. Consequently, it can be concluded that a future scenario of inter-speciality co-operation within clinical hospitals will include the automatic selection and notification of relevant personnel for a given procedure.

References

1. Romanowski, A., Wozniak, P.: Medical imaging data sharing for emergency cases. In: Romanowski, A., Sankowski, D. (eds.) Computer Science in Novel Applications. Lodz Univeristy of Technology Press (2012)
2. Magrabi, F.: Using cognitive models to evaluate safety-critical interfaces in healthcare. In: CHI 2008 Extended Abstracts on Human Factors in Computing Systems, CHI EA 2008, pp. 3567–3572. ACM, New York (2008)
3. Harper, P.: A framework for operational modelling of hospital resources. Health Care Management Science 5, 165–173 (2002)
4. Khanna, S., Cleaver, T., Sattar, A., Hansen, D., Stantic, B.: Multiagent Based Scheduling of Elective Surgery. In: Desai, N., Liu, A., Winikoff, M. (eds.) PRIMA 2010. LNCS, vol. 7057, pp. 74–89. Springer, Heidelberg (2012)
5. Devi, S.P., Rao, K.S., Sangeetha, S.S.: Prediction of surgery times and scheduling of operation theaters in optholmology department. J. Med. Syst. 36(2), 415–430 (2012)
6. Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., Riboni, D.: A survey of context modelling and reasoning techniques. Pervasive Mob. Comput. 6(2), 161–180 (2010)
7. Acampora, G., Loia, V.: A proposal of ubiquitous fuzzy computing for ambient intelligence. Inf. Sci. 178(3), 631–646 (2008)
8. Ye, J., Dobson, S., McKeever, S.: Review: Situation identification techniques in pervasive computing: A review. Pervasive Mob. Comput. 8(1), 36–66 (2012)
9. Anagnostopoulos, C.B., Ntarladimas, Y., Hadjiefthymiades, S.: Situational computing: An innovative architecture with imprecise reasoning. J. Syst. Softw. 80(12), 1993–2014 (2007)
10. Topaloglu, S., Selim, H.: Nurse scheduling using fuzzy modeling approach. Fuzzy Sets Syst. 161(11), 1543–1563 (2010)
11. Wi, H., Oh, S., Mun, J., Jung, M.: A team formation model based on knowledge and collaboration. Expert Syst. Appl. 36(5), 9121–9134 (2009)
12. Strnad, D., Guid, N.: A fuzzy-genetic decision support system for project team formation. Appl. Soft Comput. 10(4), 1178–1187 (2010)
13. Sadegh-Zadeh, K.: The fuzzy revolution: Goodbye to the aristotelian weltanschauung. Artif. Intell. Med. 21(1-3), 1–25 (2001)
14. Bardram, J.E.: Applications of context-aware computing in hospital work: examples and design principles. In: Proceedings of the 2004 ACM Symposium on Applied Computing, SAC 2004, pp. 1574–1579. ACM, New York (2004)
15. Bardram, J.E.: Activity-based computing for medical work in hospitals. ACM Trans. Comput.-Hum. Interact. 16(2), 10:1–10:36 (2009)
16. Zadeh, L.A.: Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A. Zadeh. World Scientific Publishing Co., Inc., River Edge (1996)

Author Index

Acevedo, Elena I-183
Acevedo, Marco Antonio I-183
Acosta-Mesa, Héctor Gabriel I-419
Agrawal, R.K. I-87, I-205, I-407
Aguilar Noriega, Leocundo II-411
Aguilar-Ponce, Ruth M. I-171
Alba, Alfonso I-171
Alemán, Yuridiana I-97
Alexandropoulou, Stavroula II-12
Álvarez Salgado, Carlos Fco II-411
Ameca-Alducin, Maria Yaneli I-419
Annicchiarico, Roberta I-395
Ansari, Sara I-51
Arce-Santana, Edgar I-171
Armbruster, Dieter II-421

Baltazar, Arturo I-261
Bandyopadhyay, Sivaji I-73, II-26, II-36
Barragán, Irving II-338
Barrón-Estrada, María Lucía II-444
Barrué, Cristian I-395
Barták, Roman I-359
Baruch, Ieroham II-304
Bello, Rafael I-39
Bello-García, Marilyn I-39
Bermeo, Nestor Velasco I-298
Bogdan, Karina Olga Maizman I-383
Bracho-Rios, Josue II-127

Caballero-Morales, Santiago Omar II-175
Caballero-Mota, Yailé I-27
Camarena-Ibarrola, Antonio I-119
Carlos, González-Rojas I-194
Carrasco-Ochoa, Jesús A. I-61
Carrión, Vicente II-81
Castillo, Oscar II-247, II-259, II-362, II-374
Castro, Alexander García I-298
Castro, Juan R. II-411, II-456
Castro-Manzano, José Martín I-321, I-334
Castro-Sánchez, Noé I-1
Cervantes, Leticia II-362
Chaaraoui, Alexandros Andre II-163
Chanona-Hernández, Liliana II-1
Climent-Pérez, Pau II-163
Conant-Pablos, Santiago E. I-146, I-347
Cortés, Ulises I-395
Cruz-Álvarez, Víctor Ricardo I-216
Cruz-Cortés, Nareli II-115
Cruz-Ramírez, Nicandro I-419
Cruz-Ramos, Marco Polo I-249
Cuellar, Felipe I-432

Dagan, Ido II-12
Das, Dipankar I-73
da Silva, Valdinei Freire I-371, I-383
de Amorim, Renato Cordeiro I-15
de León, Eunice Ponce I-432
del Rey, Angel Martín I-458
Dempsey, Morgan II-421
Díaz-Rangel, Ismael I-1
Dueñas, Irais Heras I-298

Enrique, Guzmán-Ramírez I-194
Esparza, Carlos II-271
Esteves, Claudia I-227

Felipe, Federico I-183
Fiderek, Pawel I-469
Flórez-Revuelta, Francisco II-163
Fuentes, Olac II-329

García-Lorenzo, María Matilde I-27, I-39
García-Moreno, Angel-Iván I-285
García Vazquez, Mauricio J. I-227
García-Vázquez, Mireya S. I-158
Garrido, Leonardo II-433
Gaxiola, Fernando II-259
Gelbukh, Alexander I-1, I-73, II-1, II-26, II-36
Gomez, Juan Carlos II-91
Gómez-Adorno, Helena I-97
González, Fabio II-271
González, Juan Pablo Nieto II-236, II-317

González-Barbosa, Erick-Alejandro I-285
Gonzalez-Barbosa, José-Joel I-285
González-Campos, Guillermo II-200
González-Estrada, Luis II-283
Gonzalez-Hernandez, Loreto II-127
González-Marrón, David I-446
González-Mendoza, Miguel I-311, I-446
González-Sanmiguel, Gustavo II-283
Gordillo, J.L. I-249, I-273
Gordon, Juan I-1
Grau, Isel II-188
Grau, Ricardo II-188
Guadalupe Martínez-Villaseñor, María de Lourdes I-311

Hassard, Christian I-249, I-273
Hayet, Jean-Bernard I-227
Hernández, J. Alejandro II-350
Hernández, Sergio II-81
Hernández Aguirre, Arturo II-223
Hernández-Gress, Neil I-446
Hernandez-Manzano, Sergio-Miguel II-304
Herrera Guzmán, Rafael II-223
Hurtado-Ramos, Juan-Bautista I-285

Ignacio, Arroyo-Fernández I-194

Jaiswal, Ajay I-87
Jaworski, Tomasz I-469
Jesús, Linares-Flores I-194

Katrenko, Sophia II-12
Kaur, Baljeet I-407
Kempf, Karl G. II-421
Khoudour, Louahdi I-131
Klockmann, Heidi II-12
Kriheli, Boris II-399
Kucharski, Jacek I-469
Kumar, Nitin I-87
Kuri-Morales, Angel II-292

La Cruz, Azael Martinez-De II-200
León, Maikel II-188
León-Borges, Jose A. II-115
Levner, Eugene II-399
Li, Hongmin II-421
Licea, Guillermo II-456
Liu, Wen Yu II-386
López-Arévalo, Iván I-107
Lopez-Arevalo, Ivan II-104
Lorena Villarreal, B. I-273
Loya, Nahun I-97
Luévano-Hipólito, Edith II-200

Madrigal, Jorge Francisco I-227
Magaña-Lozano, Dulce J. I-347
Mar, Oscar I-227
Mariaca-Gaspar, Carlos-Román II-211
Martín-Del-Campo-Mena, Enrique I-419
Martínez, Ángel I-183
Martínez, Luis G. II-456
Martínez, Maria Jose Fresnadillo I-458
Martínez-Rodríguez, José-Lázaro I-107
Martínez-Trinidad, José Fco. I-61
Martínez-Velasco, Antonio I-395
Mateos, Luis A. I-239
Medina, Joselito II-338
Medina-Pagola, José E. I-61
Medina-Urrea, Alfonso II-46
Melin, Patricia II-247, II-259, II-362, II-374
Méndez-Cruz, Carlos-Francisco II-46
Mendoza, Miguel González I-298
Merino, Enrique García I-458
Mezura-Montes, Efrén I-216, I-419
Minami, Renato I-371
Miranda-Jiménez, Sabino I-1
Moens, Marie-Francine II-91
Montes-Gonzalez, Fernando I-216
Moreno-Ahedo, Luis II-386
Moreno-Cruz, Jacob II-304

Nápoles, Gonzalo II-188
Navarro, César I-261
Navarro-Barrientos, Jesús Emeterio II-421
Neme, Antonio II-81
Neogi, Snehasis II-26
Nuñez, Omar I-119
Núñez, Rafael II-271

Oleksiy, Pogrebnyak I-194
Olmos Pineda, Iván I-97
Ornelas-Rodríguez, Francisco-Javier I-285
Ortiz-Rodríguez, Floriberto II-211

Padilla-López, Jose Ramón II-163
Pakray, Partha II-26, II-36
Pal, Santanu II-36
Palafox Maestre, Luis E. II-411
Parra, Ricardo II-433
Parvin, Hamid I-51
Parvin, Sajad I-51
Pérez, Yasmín Hernández II-444
Pérez-Castro, Nancy I-419
Pérez-Cortés, Elizabeth II-151
Pérez-Suárez, Airel I-61
Pinales, Francisco I-432
Pinto, David I-97
Poria, Soujanya I-73, II-36

Ramírez, Gabriel II-139
Ramírez-Acosta, Alejandro Alvaro I-158
Ramírez Cruz, José Federico II-329
Ramirez-Pedraza, Alfonso I-285
Rana, Bharti I-205
Rangel-Valdez, Nelson II-127
Reyes-García, Carlos Alberto II-444
Rodríguez, Angel II-283
Rodríguez, Julio-César Tovar II-211
Rodríguez-Díaz, Antonio II-456
Rodriguez-Tello, Eduardo II-139
Romanowski, Andrzej I-469
Romero, Rodrigo II-329
Romero-Monsivais, Hillel II-139
Romero-Salcedo, Manuel II-350
Rovenský, Vladimír I-359
Ruichek, Yassine I-131
Ruiz-Mireles, Karina II-104

Salas, Reynaldo II-456
Salmane, Houssam I-131
Sánchez, Daniela II-247
Sánchez, Enrique García I-458
Sánchez, Gerardo Rodríguez I-458
Sánchez, Jose Elias García I-458
Santiago, Elizabeth II-350
Santos, José I-216
Sardana, Manju I-407
Seck-Tuoh, Juan Carlos II-338
Serrano Rubio, Juan Pablo II-223
Shamshurin, Ivan II-58
Sidorov, Grigori I-1, II-1
Sierra, Gerardo II-46
Sosa-Rodríguez, María Esther II-151
Sosa-Sosa, Victor II-104
Sosa-Sosa, Víctor-Jesús I-107
Stamatatos, Efstathios II-1
Stern, Asher II-12
Suárez-Guerra, Sergio I-1

Terashima-Marín, Hugo I-146, I-347
Toledo, Assaf II-12
Torres, Aurora I-432
Torres, María de la Luz I-432
Torres, María Dolores I-432
Torres-Jimenez, Jose II-127
Torres-Moreno, Juan-Manuel II-46
Torres-Nogales, Alan I. I-146
Torres-Treviño, Luis Martin II-200, II-283
Treesatayapun, Chidentree I-261
Treviño, Alejandro I-1
Trujillo-Romero, Felipe II-175

Uribe, Diego II-69

Valdez, Fevrier II-247, II-259, II-362, II-374
Vega, Israel Cruz II-386
Velasco, Aaron II-329
Velasco-Hernández, Jorge X. II-350
Velásquez, Francisco I-1
Velasquez, Francisco II-1
Velasquillo, Luis G. II-350
Vigueras-Gómez, Javier Flavio I-171
Villanueva, Pedro Pérez II-236
Villuendas-Rey, Yenny I-27
Vincze, Markus I-239
Viveros-Jiménez, Francisco I-1, II-115

Winter, Yoad II-12
Wozniak, Pawel I-469

Zatarain-Cabada, Ramón II-444