Advances in Intelligent Systems and Computing 318

Ana Fred
Maria De Marsico *Editors*

Pattern Recognition Applications and Methods

International Conference, ICPRAM 2013 Barcelona, Spain, February 15–18, 2013 Revised Selected Papers

Springer

Advances in Intelligent Systems and Computing

Volume 318

Series editor

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

About this Series

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

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

Advisory Board

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

Ana Fred · Maria De Marsico
Editors

Pattern Recognition Applications and Methods

International Conference, ICPRAM 2013 Barcelona, Spain, February 15–18, 2013 Revised Selected Papers

Springer

Editors
Ana Fred
Instituto de Telecomunicações, Instituto Superior Técnico
Technical University of Lisbon
Lisbon
Portugal

Maria De Marsico
Department of Computer Science
Sapienza University of Rome
Roma
Italy

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-12609-8 ISBN 978-3-319-12610-4 (eBook)
DOI 10.1007/978-3-319-12610-4

Library of Congress Control Number: 2014956204

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

Preface

This book contains the extended and revised versions of a set of selected papers from the second International Conference on Pattern Recognition (ICPRAM 2013), held in Barcelona, Spain, from 15 to 18 February 2013.

ICPRAM was organized by the Institute for Systems and Technologies of Information, Control and Communication (INSTICC) and was held in cooperation with the Association for the Advancement of Artificial Intelligence (AAAI).

The hallmark of this conference was to encourage theory and practice to meet at a single venue. The International Conference on Pattern Recognition Applications and Methods aims at becoming a major point of contact between researchers, engineers, and practitioners in the areas of Pattern Recognition. The focus is on contributions describing applications of Pattern Recognition techniques to real-world problems, interdisciplinary research, experimental and/or theoretical studies yielding new insights that advance Pattern Recognition methods. The final ambition is to spur new research lines and provide the occasion to start novel collaborations, most of all in interdisciplinary research scenarios. In fact, in many respects, there is still a frequent distance between more theoretical and more application-oriented researchers. However, theory finds a significant test bed in practical applications, while any technological design must rely on solid theoretical foundations.

The second edition of ICPRAM received 180 paper submissions from 43 countries, in all continents. This result is encouraging in itself since it testifies the interest of the research community in the cultural project sketched above. To evaluate each submission, a double-blind paper review was performed by the Program Committee, whose members are highly qualified researchers in ICPRAM topic areas. Based on the classifications provided, only 66 papers were selected for oral presentation (23 full papers and 43 short papers) and 40 papers were selected for poster presentation. The full paper acceptance ratio was about 13 %, and the

total oral acceptance ratio (including full papers and short papers) was close to 37 %. These strict acceptance ratios show the intention to preserve a high-quality forum which we expect to grow over the next years.

We deeply thank the authors, whose research and development efforts are recorded here.

December 2013

Ana Fred
Maria De Marsico

Organization

Conference Chair

Ana Fred, Instituto de Telecomunicações, Instituto Superior Técnico, Technical University of Lisbon, Portugal

Program Chair

Maria De Marsico, Sapienza Università di Roma, Italy

Organizing Committee

Marina Carvalho, INSTICC, Portugal
Helder Coelhas, INSTICC, Portugal
Vera Coelho, INSTICC, Portugal
Andreia Moita, INSTICC, Portugal
Bruno Encarnação, INSTICC, Portugal
Ana Guerreiro, INSTICC, Portugal
André Lista, INSTICC, Portugal
Raquel Pedrosa, INSTICC, Portugal
Vitor Pedrosa, INSTICC, Portugal
Cláudia Pinto, INSTICC, Portugal
Cátia Pires, INSTICC, Portugal
Susana Ribeiro, INSTICC, Portugal
Sara Santiago, INSTICC, Portugal
Margarida Sorribas, INSTICC, Portugal
José Varela, INSTICC, Portugal
Pedro Varela, INSTICC, Portugal

Program Committee

Shigeo Abe, Japan
Gady Agam, USA
Mayer Aladjem, Israel
Guillem Alenya, Spain
Genevera Allen, USA
Ethem Alpaydin, Turkey
Francisco Martínez Álvarez, Spain
Jesús Angulo, France
Annalisa Appice, Italy
Antonio Artés-Rodríguez, Spain
Thierry Artières, France
Kevin Bailly, France
Gabriella Sanniti di Baja, Italy
Emili Balaguer-Ballester, UK
Vineeth Nallure Balasubramanian, USA
Christian Bauckhage, Germany
Luis Baumela, Spain
Jon Atli Benediktsson, Iceland
Charles Bergeron, USA
André Bergholz, Germany
Monica Bianchini, Italy
Isabelle Bloch, France
Anselm Blumer, USA
Liefeng Bo, USA
Joan Martí Bonmatí, Spain
Gianluca Bontempi, Belgium
Patrick Bouthemy, France
Francesca Bovolo, Italy
Paula Brito, Portugal
Lorenzo Bruzzone, Italy
Hans du Buf, Portugal
Samuel Rota Bulò, Italy
Javier Calpe, Spain
Rui Camacho, Portugal
Gustavo Camps-Valls, Spain
Ramón A. Mollineda Cárdenas, Spain
Marco La Cascia, Italy
Oscar Castillo, Mexico
Rui M. Castro, The Netherlands
Michele Ceccarelli, Italy
Jocelyn Chanussot, France
Snigdhansu Chatterjee, USA

Frederic Chazal, France
Chi Hau Chen, USA
Seungjin Choi, Korea, Republic of Korea
Jesús Cid-Sueiro, Spain
Juan Manuel Corchado, Spain
Antoine Cornuejols, France
Tom Croonenborghs, Belgium
Jesús Manuel de la Cruz, Spain
Justin Dauwels, Singapore
Jeroen Deknijf, Belgium
Thorsten Dickhaus, Germany
Carlos Diuk, USA
Petros Drineas, USA
Gideon Dror, Israel
Artur Dubrawski, USA
Carl Henrik Ek, Sweden
Deniz Erdogmus, USA
Francesc J. Ferri, Spain
Mario Figueiredo, Portugal
Maurizio Filippone, UK
Gernot A. Fink, Germany
Vojtech Franc, Czech Republic
Damien François, Belgium
Katrin Y. Franke, Norway
Ana Fred, Portugal
Élisa Fromont, France
Sabrina Gaito, Italy
Vicente Garcia, Spain
Giorgio Giacinto, Italy
Giuliano Grossi, Italy
Sébastien Guérif, France
Jose Antonio Martin H., Spain
Amaury Habrard, France
Barbara Hammer, Germany
Dongfeng Han, USA
Edwin Hancock, UK
Mark Hasegawa-Johnson, USA
Francisco Herrera, Spain
Tom Heskes, The Netherlands
Laurent Heutte, France
Anders Heyden, Sweden
Kouichi Hirata, Japan
Su-Yun Huang, Taiwan
Laura Igual, Spain
Nazli Ikizler-Cinbis, Turkey

Jose M. Iñesta, Spain
Akihiro Inokuchi, Japan
Alan J. Izenman, USA
Saketha Nath Jagarlapudi, India
Robert Jenssen, Norway
Alfons Juan, Spain
Ata Kaban, UK
Yasushi Kanazawa, Japan
Kristian Kersting, Germany
Yunho Kim, USA
Walter Kosters, The Netherlands
Constantine Kotropoulos, Greece
Irene Kotsia, UK
Sotiris Kotsiantis, Greece
Konstantinos Koutroumbas, Greece
Oleksiy Koval, Switzerland
Adam Krzyzak, Canada
Gautam Kunapuli, USA
Jaerock Kwon, USA
Helge Langseth, Norway
Qi Li, USA
Aristidis Likas, Greece
Hantao Liu, UK
Gaelle Loosli, France
Stephane Marchand-Maillet, Switzerland
Elena Marchiori, The Netherlands
Gian Luca Marcialis, Italy
Maria De Marsico, Italy
J. Francisco Martínez-Trinidad, Mexico
Peter McCullagh, USA
Majid Mirmehdi, UK
Piotr Mirowski, USA
Suman Kumar Mitra, India
Rafael Molina, Spain
Giovanni Montana, UK
Igor Mozetic, Slovenia
Arthur Munson, USA
Laurent Najman, France
Yuichi Nakamura, Japan
Michele Nappi, Italy
Claire Nédellec, France
Zoran Nenadic, USA
Hannes Nickisch, Germany
Vicente Palazón-González, Spain
Apostolos Papadopoulos, Greece

Emilio Parrado-Hernández, Spain
Marcello Pelillo, Italy
Pedro Albertos Pérez, Spain
Gabriel Peyre, France
Frederick Kin Hing Phoa, Taiwan
Sergey Plis, USA
Sylvia Pont, The Netherlands
Philippe Preux, France
Lionel Prevost, France
Peihua Qiu, USA
Emmanuel Rachelson, France
Subramanian Ramamoorthy, UK
Jan Ramon, Belgium
Soumya Ray, USA
Nicola Rebagliati, Italy
Daniel Riccio, Italy
François Rioult, France
David Masip Rodo, Spain
Marcos Rodrigues, UK
Juan J. Rodríguez, Spain
Lior Rokach, Israel
Rosa María Valdovinos Rosas, Mexico
Juho Rousu, Finland
Yvan Saeys, Belgium
Lorenza Saitta, Italy
Luciano Sanchez, Spain
J. Salvador Sánchez, Spain
Antonio-José Sánchez-Salmerón, Spain
Michele Scarpiniti, Italy
Paul Scheunders, Belgium
Tanya Schmah, Canada
Thomas Schoenemann, Germany
Friedhelm Schwenker, Germany
Pedro Garcia Sevilla, Spain
Katsunari Shibata, Japan
Tania Stathaki, UK
Vassilios Stathopolous, UK
Stefan Steidl, Germany
Masashi Sugiyama, Japan
Shiliang Sun, China
Yajie Sun, USA
Alberto Taboada-Crispí, Cuba
Ichiro Takeuchi, Japan
Toru Tamaki, Japan
Lijun Tang, USA

Ryota Tomioka, Japan
Fabien Torre, France
Andrea Torsello, Italy
Godfried Toussaint, UAE
Giorgio Valentini, Italy
Antanas Verikas, Sweden
Michel Verleysen, Belgium
Cinzia Viroli, Italy
Jordi Vitrià, Spain
Thomas Walsh, USA
Joost van de Weijer, Spain
Pierre Armand Weiss, France
David Windridge, UK
Jianxin Wu, Singapore
Xin-Shun Xu, China
Josiane Zerubia, France
Albrecht Zimmermann, Belgium
Reyer Zwiggelaar, UK

Auxiliary Reviewers

Zahid Akthar, Italy
Davide Ariu, Italy
Andrea Baisero, Sweden
Kris Cuppens, Belgium
Salvatore Frandina, Italy
Luca Ghiani, Italy
Peter Karsmakers, Belgium
Stefano Melacci, Italy
Axel Plinge, Germany
Leonard Rothacker, Germany
Claudio Saccà, Italy
Zhaolong Shen, USA
Roberto Tronci, Italy
ZiyanWu, USA

Invited Speakers

Alberto Sanfeliu, Universitat Politècnica de Cataluyna, Spain
Tomaso Poggio, CBCL, McGovern Institute, Massachusetts Institute of Technology, USA
Mário Figueiredo. Technical University of Lisbon—IST, Portugal
Bikash Sabata, Ventana Medical Systems, USA

Contents

Part I
Theory and Methods

A Two-Part Approach to Face Recognition: Generalized Hough Transform and Image Descriptors

Marian Moise, Xue Dong Yang and Richard Dosselmann

Abstract This research considers a two-part approach to the problem of face recognition. The first part, based on a variant of the generalized Hough transform, takes a global view of the matter, specifically the edges that make up a sketch of a face. The second component, on the other hand, examines the local features of a given face using a novel image descriptor, known as the gradient distance descriptor. The proposed technique performs well in testing. Moreover, this method does not require any training and may be extended to general object recognition.

Keywords Face recognition · Generalized Hough transform · Image descriptors.

1 Introduction

Of all topics in computer vision, few have received as much attention in recent years as *face recognition* [1]. A comparatively straightforward task for a human observer, this problem is not easily handled by a machine. Humans have a seemingly innate ability to recognize not only faces, but objects in general, with only a fleeting glance of the person, or item, in question [2]. This appears to be true regardless of the subject's pose, illumination or whether or not their face is partially occluded. This research presently focuses on a particular application of face recognition, namely one of video surveillance in which there is a given database of known, or target, individuals. The goal is, thus, one of matching faces of known individuals against those in images of a given scene. This is accomplished by way of a two-part approach, in turn offering both a global, as well as local, perspective on the matter. The global shape of a face in a given image is resolved using a variant of the *generalized Hough transform*

M. Moise (✉) · X.D. Yang · R. Dosselmann
Department of Computer Science, University of Regina, Regina, SK S4S 0A2, Canada
e-mail: moise20m@cs.uregina.ca

X.D. Yang
e-mail: yang@cs.uregina.ca

R. Dosselmann
e-mail: dosselmr@cs.uregina.ca

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_1

[3] (GHT). At the same time, local features are assessed using an original *image descriptor* [4], known as the *gradient distance descriptor* (GDD). The proposed solution functions well when tested over the popular *Yale face database* [5]. This particular database enables a person to temporarily avoid issues of face alignment, cropping and background removal. In any case, unlike many existing methods, this approach does not require any training data. What's more, this technique can be extended to general object recognition. Following a brief overview of the existing literature in this area, the new ideas of this research are presented in Sect. 2, along with a discussion of the test results in Sect. 3. This is subsequently followed by some closing statements and insight in Sect. 4. Note that this paper expands on the ideas of Ref. [6], ideas that were originally documented in Ref. [7].

Both the generalized Hough transform and image descriptors, though not specifically the GDD, have been employed in various vision tasks in the past. In an early application, the GHT was used to recognize handwritten Chinese characters [8]. More recent efforts have centered on template-based matching [9], as well as matching hand-drawn illustrations of objects, such as bottles, cars, horses, saxophones and watches, to real images of these items [10]. Face recognition surfaces in Ref. [11], in which real-time face detection and tracking is carried out using the GHT. Multiple faces are detected using *Hough forests* [12] in Ref. [13]. Predating many of these works are the classic notions of *Eigenfaces* [14] and *Fisherfaces* [15], with Fisherfaces generally assumed to be the stronger of these two. Both of these techniques are evaluated, later in Sect. 3, in relation to the new method of Sect. 2. Further approaches to face recognition may be found in such surveys as Refs. [16–19].

An overview of previous efforts relating to descriptors begins with the *locally adaptive regression kernel* [20] (LARK) descriptor. It is obtained from an assortment of other descriptors by way of *principal component analysis* [21] (PCA). This descriptor, along with the three described in a moment, are compared against the new descriptor of Sect. 2.2. In Ref. [22], individual portions of two images are compared using a descriptor based on the *matrix cosine similarity* [23] (MCS) measure, the second descriptor taken up by this research. The third technique, namely the *self similarities local descriptor* [24] (SSLD), enables one to find similarities in images in which there are differences in colors and textures [24]. This makes it useful in challenging applications such as object detection in images involving hand-drawn illustrations or action detection in cluttered video data [24]. The final descriptor considered in this work is a generic one based on the popular *discrete cosine transform* [25] (DCT).

2 Method

Each of two major components of this method are thoroughly described in the ensuing subsections, along with pseudocode of the full method in Algorithm 1.

2.1 Modified GHT

In this segment, a variant of the GHT, known in this research as the *modified GHT*, is defined. Like the GHT, it has the ability to handle changes in illumination, partial occlusions and small deformations [3]. The new algorithm is used to compare a global *sketch* [26] of the face in a given, or query, image against sketches of a number of target faces in a database. Each sketch is made up of a set of *edges* [25]. In this work, edges are uncovered using a *Canny* [27] *edge operator* [28]. A number of thresholds that determine just what constitutes an edge are examined in the experiments of Sect. 3. Let $\boldsymbol{x}_i = (x_i, y_i)$ denote an individual edge in a sketch and let E be the set of all edges in a sketch, where $0 \leq i \leq n$, $|E| = n$, and x_i and y_i are the x and y coordinates, respectively, of an individual edge $\boldsymbol{x}_i \in E$ in a sketch. The angle or, more formally, *direction* [25], of an individual edge $\boldsymbol{x}_i$ is given by ϕ_i. Next, let $\boldsymbol{r}_i$ denote the vector between a given edge $\boldsymbol{x}_i$ and a *reference point* [3] $\boldsymbol{y} = (x_r, y_r)$ of a target image. This is the same reference point employed in the standard GHT and, in this particular implementation, is taken to be the center of mass of the set of edges E of a given target sketch. Just as with the conventional GHT, the vector $\boldsymbol{r}_i$, relating to a given edge point $\boldsymbol{x}_i$ of a target sketch, is added to an *R-table* [3]. An *R-table* is organized into a number of rows, or bins. An individual bin j contains the set $\{\boldsymbol{r}_j\}$ of individual vectors $\boldsymbol{r}_i$, each relating to an edge $\boldsymbol{x}_i$ at an angle $\phi_j = j\Delta\phi$ when rounded, for a selected step size $\Delta\phi$. Adding a descriptor D_i to a particular bin, where D_i, like $\boldsymbol{r}_i$ corresponds to a given edge $\boldsymbol{x}_i$ in a target sketch, one obtains a *modified R-table*. An example of a modified *R-table* that incorporates this additional descriptor information, specifically as individual sets $\{D_j\}$ of descriptors, is seen in Table 1. Image descriptors are examined in Sect. 2.2.

During the recognition process, in which a query sketch is checked against a target sketch, the descriptors D_i of the individual edges $\boldsymbol{x}_i$ of the query sketch are compared against those in the appropriate bin of the *R-table* of the target image. If a match is found, then the two corresponding edges in the query and target sketches are said to represent the same edge in a face. The details of this matching process are given in Sect. 2.2. If the descriptor of a given edge in a query sketch cannot be matched to any of those in the proper bin of the *R-table* of a target sketch, then that descriptor

Table 1 Modified *R-table*

j	ϕ_j	$\{\boldsymbol{r}_j\}$	$\{D_j\}$
0	0	$\{\boldsymbol{r}_i \mid \phi(\boldsymbol{x}_i) = 0\}$	$\{D_i \mid \phi(\boldsymbol{x}_i) = 0\}$
1	$\Delta\phi$	$\{\boldsymbol{r}_i \mid \phi(\boldsymbol{x}_i) = \Delta\phi\}$	$\{D_i \mid \phi(\boldsymbol{x}_i) = \Delta\phi\}$
2	$2\Delta\phi$	$\{\boldsymbol{r}_i \mid \phi(\boldsymbol{x}_i) = 2\Delta\phi\}$	$\{D_i \mid \phi(\boldsymbol{x}_i) = 2\Delta\phi\}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
j	$j\Delta\phi$	$\{\boldsymbol{r}_i \mid \phi(\boldsymbol{x}_i) = j\Delta\phi\}$	$\{D_i \mid \phi(\boldsymbol{x}_i) = j\Delta\phi\}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$

is discarded, and, consequently, the associated edge in the query sketch is removed from further consideration. As with the traditional GHT, when a match does occur, the individual entry $\boldsymbol{x}_i + \boldsymbol{r}_i^*$ of the Hough *accumulator* [3] array A is incremented, where $\boldsymbol{r}_i^* \in \{\boldsymbol{r}_j\}$ corresponds to the descriptor $D_i^* \in \{D_j\}$ of the appropriate bin j of the edge $\boldsymbol{x}_i$ in the query sketch. Examples of accumulator arrays are seen in Fig. 5. As with the conventional GHT, each individual entry of A is a count of the number of votes of the assumed location $\boldsymbol{x}_i + \boldsymbol{r}_i^* = \boldsymbol{y}^* = (x_r^*, y_r^*)$ of the reference point of the target sketch. As expected, the entry that receives the most votes is taken to be the reference point of the target sketch. Ultimately, the target image that receives the highest overall vote count, a value given by N, is selected as the best match to the face in the query image. The keen reader will observe that the specific pseudocode of Algorithm 1 includes an additional scaling factor when incrementing A. This is done so as to give more weight to those images that are a strong match to the current image.

2.2 Gradient Distance Descriptor

Image descriptors mathematically describe local sections of an image by way of attributes such as color, edges, luminosity, orientation, shape and texture. Moreover, certain descriptors are invariant to lighting variations, rotation, scaling, shearing, small deformations and translation, making them ideal for use in face recognition. While image descriptors typically require more memory and increase the overall computational complexity of the underlying algorithm, they are preferable to raw pixel intensities as they better represent the features of a face than do single pixels.

This research introduces a new descriptor, the GDD. It is based on the LARK descriptor, a technique that measures the similarity between a given pixel and its surrounding neighbors via image *gradients* [25] and *geodesic* [29] distance. The LARK descriptor has the ability to represent geometric shapes, even in noisy images or in the presence of distortions and backgrounds. To no surprise, it has been found to be useful in generic object detection [22]. A more detailed derivation of the LARK descriptor is given in Ref. [7]. The GDD goes one step further to consider the relative significance of individual pixels. In particular, pixels that are closer to a given pixel, ones that seemingly have more influence, are weighed more heavily than those that are further from the given pixel. Accordingly, the GDD is the weighted average of the horizontal and vertical image gradients G_x and G_y [25], denoted $\bar{G}_x$ and $\bar{G}_y$, respectively, of an edge $\boldsymbol{x}_i$, over the pixels in a patch surrounding that edge $\boldsymbol{x}_i$. Formally, for a patch of size $p \times p$ centered on a given edge $\boldsymbol{x}_i$, the GDD is equal to

$$\mathrm{GDD}(\boldsymbol{x}_i) = \begin{bmatrix} d_{1,1} & d_{1,2} & \cdots & d_{1,p} \\ d_{2,1} & d_{2,2} & \cdots & d_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ d_{p,1} & d_{p,2} & \cdots & d_{p,p} \end{bmatrix}, \tag{1}$$

where

$$d_{m,n} = \exp\left(-\left(\bar{G}_x \cdot dx_{m,n} + \bar{G}_y \cdot dy_{m,n}\right)^2\right), \tag{2}$$

for $1 \leq m, n \leq p$. So as to give more weight to those pixels that are nearest to the center of the descriptor, the average gradients $\bar{G}_x$ and $\bar{G}_y$ of each entry $d_{m,n}$ are scaled by the horizontal and vertical distances, $dx_{m,n}$ and $dy_{m,n}$, respectively, of a pixel $q_{m,n} \in \text{GDD}(\boldsymbol{x}_i)$ from the center of the descriptor. In particular,

$$dx_{m,n} = n - \left(\frac{p+1}{2}\right) \tag{3}$$

and

$$dy_{m,n} = m - \left(\frac{p+1}{2}\right). \tag{4}$$

The average gradients $\bar{G}_x$ and $\bar{G}_y$ are calculated using the MatLab® *circular averaging filter*, denoted as $K^{p\times p}$ in this paper. Formally,

$$\bar{G}_x = \frac{1}{p^2}\sum_{u=1}^{p}\sum_{v=1}^{p}\left(a_{u,v} \cdot G_x\right) \tag{5}$$

and

$$\bar{G}_y = \frac{1}{p^2}\sum_{u=1}^{p}\sum_{v=1}^{p}\left(a_{u,v} \cdot G_y\right), \tag{6}$$

for weights $a_{u,v} \in K^{p\times p}$.

The process of matching descriptors D_i, mentioned earlier in Sect. 2.1, is carried out using the robust MCS measure. The MCS measure was selected over the competing *correlation* [25] measure given its improved accuracy [30]. Further evidence of this improved performance is seen in the three plots of Fig. 1, in which a given image is compared, using these two measures, to individually brightened, contrast-adjusted and rotated versions of that image. In all three instances, the MCS measure outperforms the correlation measure, regardless of the particular choice of patch size. Note that the similarity of each measure, given on the vertical axes of the plots of Fig. 1, is a value in the interval [0, 1], where a lower value corresponds to a smaller degree of similarity and a larger value indicates a higher degree of similarity. Moving on, the degree of similarity between any two descriptors is given by δ. A match between two descriptors occurs when $\delta < \epsilon$, for a given threshold ϵ. In this research, the GDD takes the place of the generic descriptor D_i, which represents any descriptor. Since any descriptor may be used, the performance of the new method given in this research will inevitably improve as descriptors improve.

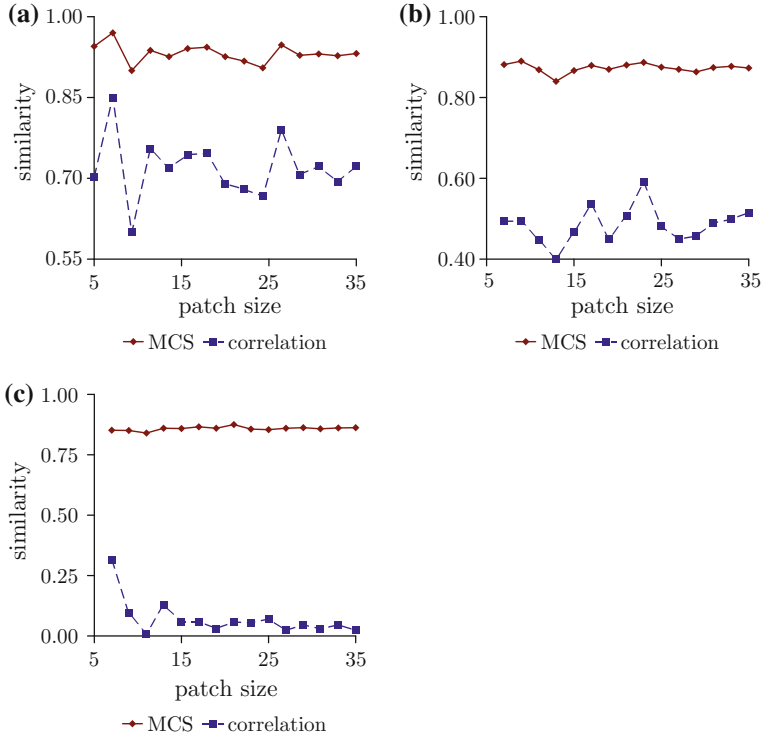


Fig. 1 Performance comparison of MCS measure versus correlation measure over three modified versions of a given image under varying patch sizes (original plots in *color*); **a** brightened image; **b** contrast-adjusted image; **c** rotated image (Color figure online)

3 Results and Discussion

The two-part algorithm of Sect. 2 is tested over the popular Yale face database. This database consists of 15 subjects, including both males and females, each in 11 different environments, resulting in a total of $15 \cdot 11 = 165$ images. Each of these 165 images is individually compared against the other 164 images in the database. A search is deemed to be "successful" if the current image is matched to one of the other ten images corresponding to the individual in the current image. The average processing time needed to compare two faces from this particular database is roughly 20 s. The overall rate of success, or recognition rate, is plotted on the vertical axes of each of the ensuing plots. Unless stated otherwise, in each of the following experiments, $\Delta\phi = 3°$, $\epsilon = 0.05$ and $C = 1 \times 10^6$, where C is part of the scaling factor mentioned briefly in Sect. 2.1.

In the first experiment, the modified GHT, in which each of the GDD, LARK, SSLD and DCT descriptors are substituted for the generic descriptor D_i, is tested. Four separate scenarios are considered. In the first scenario, the patch size of the four

$A \leftarrow 0$ {initialize accumulator array A to 0}
$R \leftarrow \emptyset$ {initialize R-table R to empty}

for all $\boldsymbol{x}_i \in E$ **do**
  $\phi_i \leftarrow$ direction of $\boldsymbol{x}_i$
  $\boldsymbol{r}_i \leftarrow \boldsymbol{x}_i - \boldsymbol{y}$
  $D_i \leftarrow$ descriptor of $\boldsymbol{x}_i$
  $R.\{\boldsymbol{r}_j\} \leftarrow R.\{\boldsymbol{r}_j\} \cup \{\boldsymbol{r}_i\} \mid \phi(\boldsymbol{x}_i) = \phi_j$ {add new entry $\boldsymbol{r}_i$ to $\{\boldsymbol{r}_j\}$ of R}
  $R.\{D_j\} \leftarrow R.\{D_j\} \cup \{D_i\} \mid \phi(\boldsymbol{x}_i) = \phi_j$ {add new entry D_i to $\{D_j\}$ of R}

  for all $D_i^* \in R.\{D_j\} \mid \phi(\boldsymbol{x}_i) = \phi_j$ **do**
    $\delta \leftarrow MCS\left(D_i, D_i^*\right)$
    if $\delta < \epsilon$ **then**
      $\boldsymbol{y}^* \leftarrow \boldsymbol{x}_i + \boldsymbol{r}_i^*$ {$\boldsymbol{r}_i^* \in R.\{\boldsymbol{r}_j\}$ corresponds to $D_i^* \in R.\{D_j\}$}
      $A(\boldsymbol{y}^*) \leftarrow A(\boldsymbol{y}^*) + \text{round}(C(\epsilon - \delta)) + 1$ {increment vote count}
    end if
  end for

end for

$N \leftarrow$ get_max_accumulator_array_vote_count(A)
$\hat{\boldsymbol{y}} \leftarrow$ get_max_accumulator_array_vote_count_point(A)
return $\{N, \hat{\boldsymbol{y}}\}$ {N is highest vote count, $\hat{\boldsymbol{y}}$ is best estimate of $\boldsymbol{y}$}

Algorithm 1: Modified GHT.

descriptors is varied. Exact sizes range from 7×7 to 35×35. The results are depicted in the plot of Fig. 2a. The recognition rate of each of the GDD, LARK and SSLD descriptors remains largely constant, regardless of the patch size. The performance of the DCT, on the other hand, improves as the patch size increases. In the second scenario, the results of which are displayed in Fig. 2b, the magnitude of the Canny edge threshold is varied from between 0.20 and 0.55. This threshold determines the number of edges, and therefore facial features, that are retained in a sketch. As the threshold is lowered, more details are preserved. In this second scenario, all four descriptors show varying degrees of performance as the threshold changes, with the GDD generally showing the best performance of the pack. The performance of the LARK descriptor appears to improve as the threshold increases. Conversely, the performance of the DCT descriptor falls as the threshold rises. Lastly, the SSLD appears to function best for a single threshold, namely 0.35, with poorer performance for both larger and smaller thresholds. The third scenario looks at the effects of raising the threshold ϵ from a minimum of 0.0001 to a maximum of 0.1000. This time, there are noticeable differences in the performance of each of the three competing descriptors, as one can see from Fig. 2c. Perhaps most exciting, the GDD shows consistent performance despite changes in the value of ϵ. The remaining three descriptors generally show lower performance for smaller thresholds. Should the threshold be lowered too much, however, the recognition rate will fall, regardless of the descriptor used. In fact, when the threshold ϵ is very small, it is often the case that $\delta \not< \epsilon$, meaning, of course, that the associated edge point is excluded from the voting process. With fewer edges taking part in the voting process, the overall recognition rate begins to

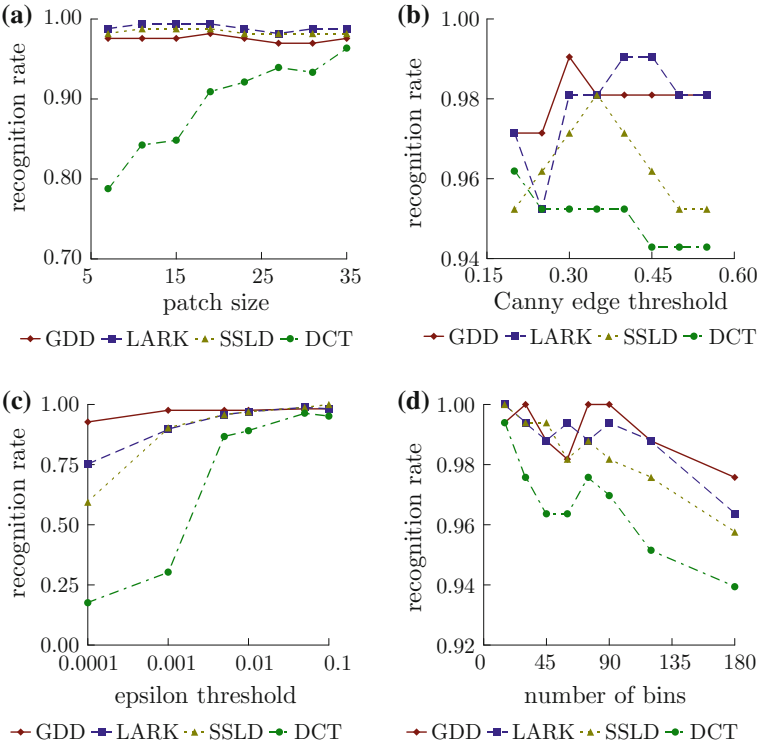


Fig. 2 Performance comparisons of modified GHT using GDD, LARK, SSLD and DCT descriptors over database under varying conditions (original plots in *color*); **a** varying patch size; **b** varying Canny edge threshold; **c** varying epsilon (ϵ) threshold (logarithmic scale); **d** varying number of bins (Color figure online)

fall. In the final scenario, the performance of the four descriptors is assessed as the number of bins changes. The precise number of bins considered varies between 15 and 180. The results of this final scenario are depicted in Fig. 2d. The GDD performs somewhat better than the others. In all cases, though, the recognition rate ultimately drops as the number of bins increases. The number of bins determines the maximum allowable difference between the angles of the individual edges in a given bin. Thus, when there are many bins, small differences in the individual angles ϕ_i of edges, typically the result of numerical error, cause these similarly-oriented edges to be placed into separate bins. Conversely, when there are fewer such bins the recognition rate is noticeably higher, as tiny variations in this angle become more or less negligible. If there are, for instance, 20 bins, then the maximum allowable difference is equal to $360°/20 = 18°$. If, however, there are many more bins, 90 perhaps, then this difference decreases to only $360°/90 = 4°$. Although having fewer bins has the advantage of making the method more robust to minor fluctuations in the directions of edges, it increases the computational complexity as there are more descriptors in each bin,

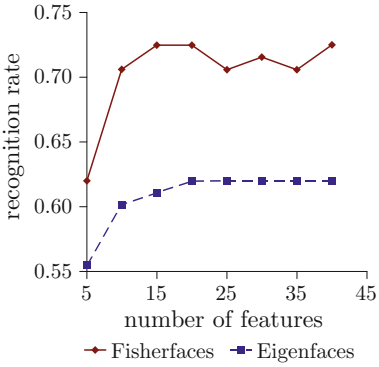


Fig. 3 Performance comparison of Eigenfaces and Fisherfaces over database (original plot in *color*)(Color figure Online)

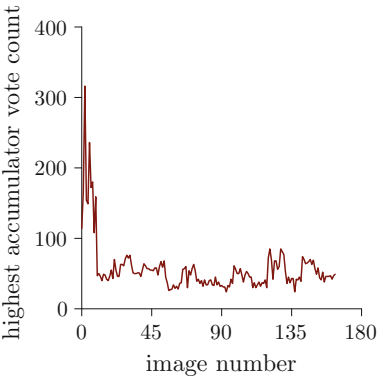


Fig. 4 Highest accumulator array vote counts over database (original plot in *color*)(Color figure Online)

which means that more comparisons between descriptors have to be performed.

The second experiment evaluates the performance of the classic Eigenfaces and Fisherfaces methods over this same database. This, therefore, allows one to compare the performance of these two traditional methods against the proposed technique based on the modified GHT. Training for both the Eigenfaces and Fisherfaces methods was carried out using 60 of the 165 images in the database, with the remaining 105 used for testing. The Fisherfaces procedure, as expected, decidedly outperforms the competing Eigenfaces method, as observed in Fig. 3.

According to the plot of Fig. 3, the Eigenfaces method does not achieve a recognition rate much beyond 0.60. The Fisherfaces approach, on the other hand, scores much higher, closer to 0.70. Looking back at the results of the preceding experiment, the modified GHT, in combination with the GDD, achieves recognition rates above 0.92, thereby significantly outperforming both of these classic approaches. For more information pertaining to the number of features term, given on the horizontal axes of the two plots of Fig. 3, see Refs. [14, 15].

Two additional experiments are conducted in order to better illustrate the underlying behavior of the modified GHT algorithm. The first, captured in the plot of Fig. 4, depicts the individually highest accumulator vote counts obtained when matching the first of the 165 images in the database against the remaining 164. This experiment

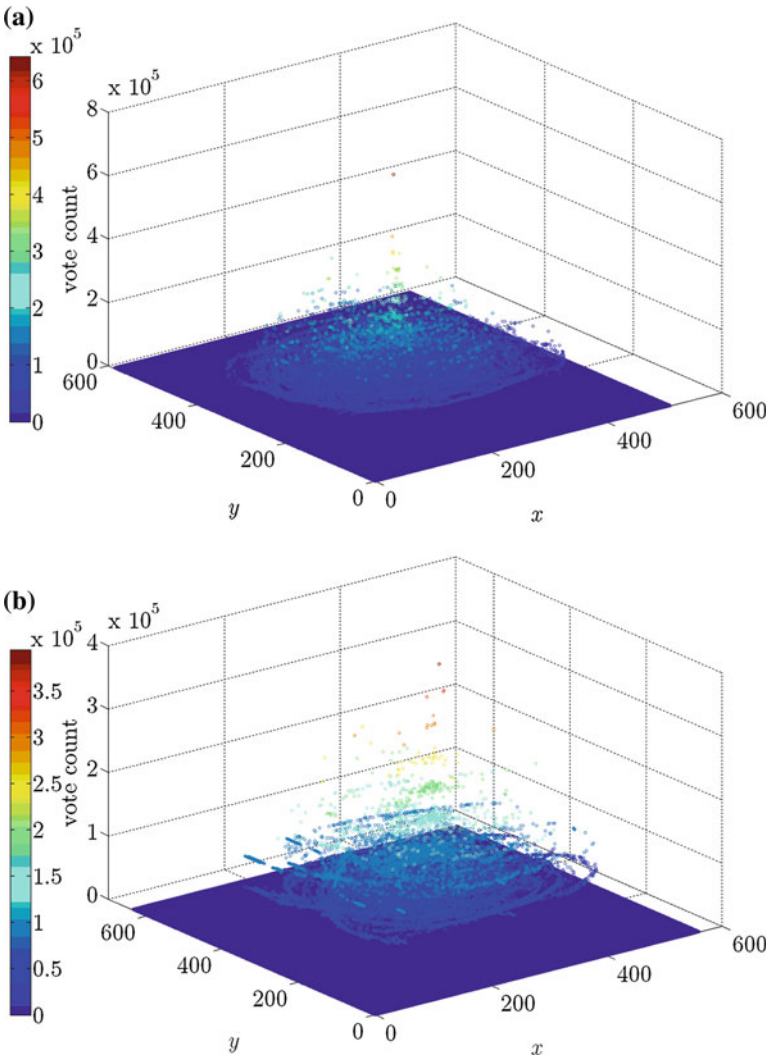


Fig. 5 Accumulator arrays (original plots in *color*); **a** same individual with and without glasses; **b** two different individuals (Color figure online)

was undertaken as part of an earlier effort and therefore makes use of the LARK descriptor rather than the newer GDD. In any case, one will notice that the highest vote counts are obtained over the first ten images, specifically those of the same individual in this first image of the database. Lower vote counts are obtained over the remaining images of the database, namely those corresponding to images of individuals other than the person in the first image.

Finally, as part of the last experiment, two separate accumulator arrays are visualized in Fig. 5. In both cases, brighter colors, in the vertical direction, represent higher vote counts, whereas darker shades, also along the vertical axis,

correspond to smaller counts. The x and y axes, together, represent the inferred position $\boldsymbol{y}^* = (x_r^*, y_r^*)$ of the reference point. The array shown in Fig. 5a is obtained by comparing two images of the same individual, one in which the person is wearing glasses and one in which the person is not. There are a few "significant" counts in the array of Fig. 5a, specifically those in the range of thereabouts 3×10^5 to 6×10^5. Conversely, the counts in the array of Fig. 5b, in which two different individuals are compared, are noticeably lower, with most lying between 1.5×10^5 and 3×10^5. This sizeable gap in counts can be used to distinguish between individuals, with a relatively high count suggesting that the faces in two different images are those of the same person, while a lower count implies that they are not of the same person.

4 Conclusions

This research addresses the long standing problem of face recognition using a two-part approach based on a variant of the GHT, along with a new image descriptor. One of the most significant advantages of the modified GHT is the fact that it does not require any training data. And, like the traditional GHT, it has the ability to handle changes in illumination, partial occlusions and small deformations. Better yet, this algorithm can be upgraded as new and more powerful descriptors become available.

A number of enhancements and extensions may explored at some point. First, it is imperative that the new method be tested over a variety of additional databases, such as the well-known *labeled faces in the wild* [31] database. Such databases bring new challenges, specifically ones relating to background removal, cropping and image alignment. And, perhaps as part of a larger study, the modified GHT could be compared with more than just Eigenfaces and Fisherfaces.

More specific enhancements include, for example, the introduction of a set of *attribute classifiers* [32] to reduce the number of target images in a database that need to be considered. This is especially important when it comes to much larger databases. These binary classifiers can be trained to differentiate individuals based on traits such as age, hair color, gender, race or other features. Thus, when incorporated into a face recognition system, they immediately allow certain target faces to be ruled out, greatly reducing the overall computational burden.

Ensuring the proper alignment of faces is another major concern. Some of the best alignment techniques are those based on the concept of *mutual information* [33]. This procedure is especially attractive given that it does not require complete information about the surface properties of a face. Instead, it relies only on the shape of a face. It is also robust to variations in illumination, a key concern in this field. Moreover, as documented in Ref. [34], this approach works well in domains in which edge or gradient methods experience difficulties and it is more robust than correlation. A rather efficient algorithm is given in Ref. [35]. This face alignment procedure is said to be robust not only to the effects of illumination, but also those of occlusion. This algorithm is thought to better capture underlying image structure than those approaches based on mere pixel intensities.

One might also wish to look at additional descriptors. One prospective contender in this area is the *local binary patterns* [36] (LBP) descriptor. This technology works by extracting features from a swath of different regions of an image and subsequently concatenating them to build a single descriptor. The SIFT descriptor of Ref. [37] offers another approach, this time one that is invariant to rotation and scaling and that can address complications stemming from changes in viewpoint, illumination and noise [38]. Another approach in this direction is presented in Ref. [39]. It, too, is engineered to deal with the effects of image deformations. An altogether distinct idea, given in Ref. [40], draws on the psychological and physiological characteristics of the *human visual system* [41] (HVS), most notably the *Weber-Fechner law* [41]. The result is a descriptor, identified as the *Weber local descriptor* [40] (WLD), that exploits the notion that humans perceive patterns according not only to changes in the intensity of a stimuli, but also the initial intensity of a stimuli. Finally, with different descriptors having their own unique advantages, one might also consider the possibility of combining multiple descriptors, with each encoding different elements of a face image.

Lastly, the task of identifying multiple faces in an image, a situation that frequently turns up in practice, could be addressed by way of the Hough forests method mentioned earlier and given in Ref. [13]. A further means of handling multiple faces is to employ a face detection scheme as part of a preprocessing stage. Later, only those faces actually detected would be considered by the modified GHT.

References

1. Li, S., Jain, A. (eds.): Handbook of Face Recognition, 2nd edn. Springer, New York (2011)
2. Liu, L., Özsu, M. (eds.): Encyclopedia of Database Systems. Springer, New York (2009)
3. Ballard, D.: Generalizing the Hough transform to detect arbitrary shapes. Pattern Recognit. **13**, 111–122 (1981)
4. Goshtasby, A.: Image Registration: Principles, Tools and Methods. Springer, New York (2012)
5. Yale Face Database. Retrieved July 4, 2012 from http://cvc.yale.edu/projects/yalefaces/yalefaces.html (1997)
6. Moise, M., Yang, X. D., Dosselmann, R.: Face recognition using modified generalized Hough transform and gradient distance descriptor. In: Proceedings of the 2nd International Conference Pattern Recognition Applications and Methods (2013)
7. Moise, M.: A new approach to face recognition based on generalized Hough transform and local image descriptors. Master's thesis. University of Regina, Regina, Canada (2012)
8. Li, M.-J., Dai, R.-W.: A personal handwritten Chinese character recognition algorithm based on the generalized Hough transform. In: Proceedings of the International Conference Document Analysis and Recognition, vol. 2, pp. 828–831 (1995)
9. Li, Q., Zhang, B.: Image matching under generalized Hough transform. In: Proceedings of the IADIS International Conference Applied Computing, pp. 45–50 (2005)
10. Anelli, M., Cinque, L., Sangineto, E.: Deformation tolerant generalized Hough transform for sketch-based image retrieval in complex scenes. Image Vis. Comput. **25**, 1802–1813 (2007)
11. Schubert, A.: Detection and tracking of facial features in real time using a synergistic approach of spatio-temporal models and generalized Hough transform techniques. In: Proceedings of the 4th IEEE International Conference Automatic Face and Gesture Recognition, pp. 116–121 (2000)

12. Gall, J., Lempitsky, V.: Class-specific Hough forests for object detection. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition (2009)
13. Barinova, O., Lempitsky, V., Kohli, P.: On detection of multiple object instances using Hough transforms. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, pp. 2233–2240 (2010)
14. Turk, M., Pentland, A.: Eigenfaces for recognition. J. Cognit. Neurosci. **3**, 71–86 (1991)
15. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces versus fisherfaces: recognition using class specific linear projection. IEEE Trans. Pattern Anal. Mach. Intell. **19**, 711–720 (1997)
16. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face recognition: a literature survey. ACM Comput. Surv. **35**, 399–458 (2003)
17. Kong, S., Heo, J., Abidi, B., Paik, J., Abidi, M.: Recent advances in visual and infrared face recognition: a review. Comput. Vis. Image Underst. **97**, 103–135 (2005)
18. Abate, A., Nappi, M., Riccio, D., Sabatino, G.: 2D and 3D face recognition: a survey. Pattern Recognit. Lett. **28**, 1885–1906 (2007)
19. Zhang, X., Gao, Y.: Face recognition across pose: a review. Pattern Recognit. **42**, 2876–2896 (2009)
20. Seo, H., Milanfar, P.: Face verification using the LARK representation. IEEE Trans. Inf. Forensics Secur. **6**, 1275–1286 (2011)
21. Duda, R., Hart, P., Stork, D.: Pattern Classification, 2nd edn. Wiley, New York (2001)
22. Seo, H., Milanfar, P.: Nonparametric detection and recognition of visual objects from a single example. In: Workshop on Defense Applications of Signal Processing (2009)
23. Seo, H.J., Milanfar, P.: Training-free, generic object detection using locally adaptive regression kernels. IEEE Trans. Pattern Anal. Mach. Intell. **32**, 1688–1704 (2010)
24. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: IEEE Conference Computer Vision Pattern Recognition, pp. 1–8 (2007)
25. Gonzalez, R., Woods, R.: Digital Image Processing, 2nd edn. Prentice Hall, New Jersey (2002)
26. Li, S. (ed.): Encyclopedia of Biometrics. Springer, New York (2009)
27. Canny, J.: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **8**, 679–698 (1986)
28. Burger, W., Burge, M.: Digital Image Processing: An Algorithmic Introduction Using Java. Springer, New York (2008)
29. Schneider, P., Eberly, D.: Geometric Tools for Computer Graphics. Morgan Kaufmann, San Francisco (2003)
30. Schneider, J., Borlund, P.: Matrix comparison, part 1: motivation and important issues for measuring the resemblance between proximity measures or ordination results. J. Am. Soc. Inf. Sci. Technol. **58**, 1586–1595 (2007)
31. Huang, G., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: a database for studying face recognition in unconstrained environments. Retrieved 14 Nov 2012 from http://viswww.cs.umass.edu/lfw/ (2007)
32. Kumar, N., Berg, A., Belhumeur, P., Nayar, S.: Attribute and simile classifiers for face verification. In: Proceedings of the IEEE International Conference Computer Vision, pp. 365–372 (2009)
33. Bose, R.: Information Theory, Coding and Cryptography, 2nd edn. Tata McGraw-Hill, New Delhi (2008)
34. Viola, P., Wells, W.M.: Alignment by maximization of mutual information. In: 5th International Conference Computer Vision, pp. 16–23 (1995)
35. Tzimiropoulos, G., Zafeiriou S., Pantic, M.: Robust and efficient parametric face alignment. In: Proceedings of the International Conference Computer Vision, pp. 1847–1854 (2011)
36. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. **24**, 971–987 (2002)
37. Bicego, M., Lagorio, A., Grosso, E., Tistarelli, M.: On the use of SIFT features for face authentication. In: Computer Vision and Pattern Recognition Workshop (2006)

38. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**, 91–110 (2004)
39. Cheng, H., Liu, Z., Zheng, N., Yang, J.: A deformable local image descriptor. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, pp. 1–8 (2008)
40. Chen, J., Shan, S., He, C., Zhao, G., Pietikäinen, M., Chen, X., Gao, W.: WLD: a robust local image descriptor. IEEE Trans. Pattern Anal. Mach. Intell. **32**, 1705–1720 (2010)
41. Winkler, S.: Digital Video Quality: Vision Models and Metrics. Wiley, New Jersey (2005)

Improved Boosting Performance by Explicit Handling of Ambiguous Positive Examples

Miroslav Kobetski and Josephine Sullivan

Abstract Visual classes naturally have ambiguous examples, that are different depending on feature and classifier and are hard to disambiguate from surrounding negatives without overfitting. Boosting in particular tends to overfit to such hard and ambiguous examples, due to its flexibility and typically aggressive loss functions. We propose a two-pass learning method for identifying ambiguous examples and relearning, either subjecting them to an exclusion function or using them in a later stage of an inverted cascade. We provide an experimental comparison of different boosting algorithms on the VOC2007 dataset, training them with and without our proposed extension. Using our exclusion extension improves the performance of almost all of the tested boosting algorithms, without adding any additional test-time cost. Our proposed inverted cascade adds some test-time cost but gives additional improvements in performance. Our results also suggest that outlier exclusion is complementary to positive jittering and hard negative mining.

Keywords Boosting · Image classification · Algorithm evaluation · Dataset pruning · VOC2007

1 Introduction

Recent efforts to improve image classification performance have focused on designing new discriminative features and machine learning methods. However, some of the performance gains of many well-established methods are due to dataset augmentation such as hard negative mining, positive mirroring and jittering [1–4]. These data-bootstrapping techniques aim at augmenting sparsely populated regions of the dataset to allow any learning method to describe the class distributions more

M. Kobetski (✉) · J. Sullivan
Computer Vision and Active Perception, KTH, 114 28 Stockholm, Sweden
e-mail: kobetski@kth.se

J. Sullivan
e-mail: sullivan@nada.kth.se

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_2

accurately, and they have become standard tools for achieving state-of-the-art performance for classification and detection tasks. In this paper we revisit the dataset augmentation idea, arguing and showing that pruning the positive training set by excluding hard-to-learn examples can improve performance for outlier-sensitive algorithms.

We focus on the boosting framework and propose a method to identify and exclude positive examples that a classifier is unable to learn—to make better use of the available training data rather than expanding it. We refer to the non-learnable examples as outliers and we wish to be clear that these examples are not label noise (such as has been studied in [5–7]), but rather examples that with a given feature and learner combination are ambiguous and too difficult to learn. We also propose an inverted cascade that allows inclusion of these hard examples at a later stage of the classification.

One of the main problems with most boosting methods is their sensitivity to outliers such as atypical examples and label noise [5, 8–10]. Some algorithms have tried to deal with this problem explicitly [6, 10–14], while others, such as LogitBoost [15] are less sensitive due to their softer loss function.

The boosting methods with aggressive loss functions give outliers high weight when fitting the weak learner, and therefore potentially work poorly in the presence of outliers. Softer loss functions as seen in the robust algorithms can on the other hand result in low weights for all examples far from the margin, regardless if they are noisy outliers or just data to which the current classifier has not yet been able to fit. This can be counter-productive in cases of hard inliers, which is illustrated in Fig. 2a. Another problem that soft loss functions are not able to solve, is that outliers are still able to affect the weak learners during the early stages of the training, which due to the greedy nature of boosting can only be undone later on by increasing the complexity of the final classifier. In this paper we provide an explicit analysis on how various boosting methods relate to examples via their weight functions and we argue that a distinct separation in the handling of inliers and outliers can help solve these problems that current robust boosting algorithms are facing.

Following this analysis we propose our two pass boosting algorithm extension, that explicitly handles learnable and non-learnable examples differently. We define outliers as examples that are too hard-to-learn for a given feature and weak learner set, and identify them based on their classification score after a first training round. A second round of training is performed, where the outliers are subjected to a much softer loss function and are therefore not allowed to interfere with the learning of the easier examples, in order to find a better optimum. This boosting algorithm extension consistently gives better test performance, with zero extra test-time costs at the expense of increased training time. Some examples of found inliers and outliers can be seen in Fig. 1. We also propose a method for reintroducing the hard examples without reducing the performance of the classifier. We call this an inverted cascade.

Fig. 1 Examples of outliers and inliers. The top rows of (a) and (b) show outliers while the bottom rows show inliers. We focus on how to detect the outliers and how their omission from training improves test results. The images are from the VOC2007 dataset. **a** Chair class, **b** Bottle class

1.1 Relation to Bootstrapping Methods

To further motivate our data-centric approach to learning, we illustrate the problems that different dataset augmentation techniques address. In regions where the positive training examples are dense and the negatives are existent but sparse, hard negative mining might improve the chances of finding the optimal decision boundary. In regions where positives are sparse and negatives existent, jittering and mirroring might have some effect, but the proper analogue to hard negative mining is practically much harder, since positive examples need to be labelled. At some scale this can be done by active learning [16], where labelling is done iteratively on selected examples.

Our approach tries to handle the regions where positives are sparse but additional hard positive mining is not possible, either due to limited resources or because all possible positive mining has already been done. We address this problem by restricting hard-to-learn positives from dominating the training with their increasingly high weights by excluding them from the training.

Our algorithm can be considered as dataset pruning and makes us face the high-level question of more data versus better data—something that has recently been addressed by [17]. It has been shown that in cases where huge labelled datasets are available, even simple learning methods perform very well [18–20]. We address the opposite case, where a huge accurately labelled data set cannot be obtained—a common scenario both in academic and industrial computer vision.

1.2 Contributions

We propose a two-pass boosting extension algorithm, suggested by a weight-centric theoretical analysis of how different boosting algorithms respond to outliers. We also demonstrate that it is important to distinguish between "hard-to-learn" examples and "non-learnable" outliers in vision as examples easily identified as positive by humans could be non-learnable given a feature and weak-learner set, and demonstrate that the different classes in VOC2007 dataset indeed have different fractions of hard-to-learn examples using HOG as base feature. We also propose the inverted cascade—that allows the hard positive examples to be re-introduced at a later stage. Finally we provide extensive experimental comparison of different boosting algorithms on real computer vision data and perform experiments using dataset augmentation techniques, showing that our method is complementary to jittering and hard negative mining.

2 Relation to Previous Work

As previously mentioned AdaBoost has been shown to be sensitive to noise [8, 9]. Other popular boosting algorithms such as LogitBoost or GentleBoost [15] have softer loss functions or optimization methods and can perform better in the presence of noise in the training data, but they have not been specifically designed to handle this problem. It has been argued that no convex-loss boosting algorithm is able to cope with random label noise [5]. This is however not the problem we want to address, as we focus on naturally occurring outliers and ambiguous examples, which is a significant and interesting problem in object detection today.

BrownBoost [11] and RobustBoost [10] are adaptive extensions of the Boost-By-Majority Algorithm [21] and have non-convex loss functions. Intuitively these algorithms "give up" on hard examples and this allows them to be less affected by erroneous examples.

Regularized LPBoost, SoftBoost and regularized AdaBoost [14, 22] regularize boosting to avoid overfitting to highly noisy data. These methods add the concept of soft margin to boosting by adding slack variables in a similar fashion to soft-margin SVMs, and this decreases the influence of outliers. Conceptually these methods bear some similarity to ours as the slack variables reduce the influence of examples on the wrong side of the margin, and they define an upper bound on the fraction ν of misclassified examples, which is comparable to the fraction of the dataset excluded in the second phase of training.

There is recent work on robust boosting where new semi-convex loss functions are derived based on probability elicitation [6, 7, 12]. These methods have shown potential for high-noise problems such as tracking, scene recognition and artificial label noise. But they have not been extensively compared to the common outlier-sensitive algorithms on low-noise problems, such as object classification, where the existing outliers are ambiguous or uncommon examples, rather than actual label errors.

In all the mentioned robust boosting algorithms the outliers are estimated and excluded on the fly and these outliers are therefore able to affect the training in the early rounds. Also, as can be seen in Fig. 2, these algorithms can treat uncommon non-outliers as conservatively as actual outliers, resulting in suboptimal decision boundaries.

Reducing overfitting by pruning the training set has been studied previously [23, 24] but improved results have mostly been seen in experiments where training sets include artificial label noise. Reference [23] is the only method that we have found where pruning improves performance on a "clean" dataset. They use an approach very similar to ours, detecting hard-to-learn examples, then removing those examples from training. The base algorithm to which [23] apply dataset pruning is AdaBoost, which we show is the most noise-sensitive boosting algorithm and not the one that should be used for image classification. We propose a similar but more direct approach that improves results for both robust and non-robust algorithms, while still using a reasonable number of weak learners.

It is important to note that vision data is typically very high-dimensional and boosting therefore also acts as feature selection—learning much fewer weak learners than available dimensions. This is one of the key differences to typical machine learning datasets, such as the ones used for validating the method in [23]. Our experiments on the VOC2007 dataset verify that exclusion of ambiguous examples is useful for the high-dimensional problems found in computer vision. We also compare a number of well-known boosting algorithms using typical vision data, something that we have not seen previously.

A different but related topic that deals with label ambiguity is Multiple Instance Learning (MIL). Viola et al. [25] suggest a boosting approach to the MIL problem, applying their solution to train an object detector with highly unaligned training data.

Our idea is also conceptually similar to a simplified version of self-paced learning [26]. We treat the hard and easy positives separately and do not let the hard examples dominate the easy ones in the search for the optimal decision boundary. This can seen

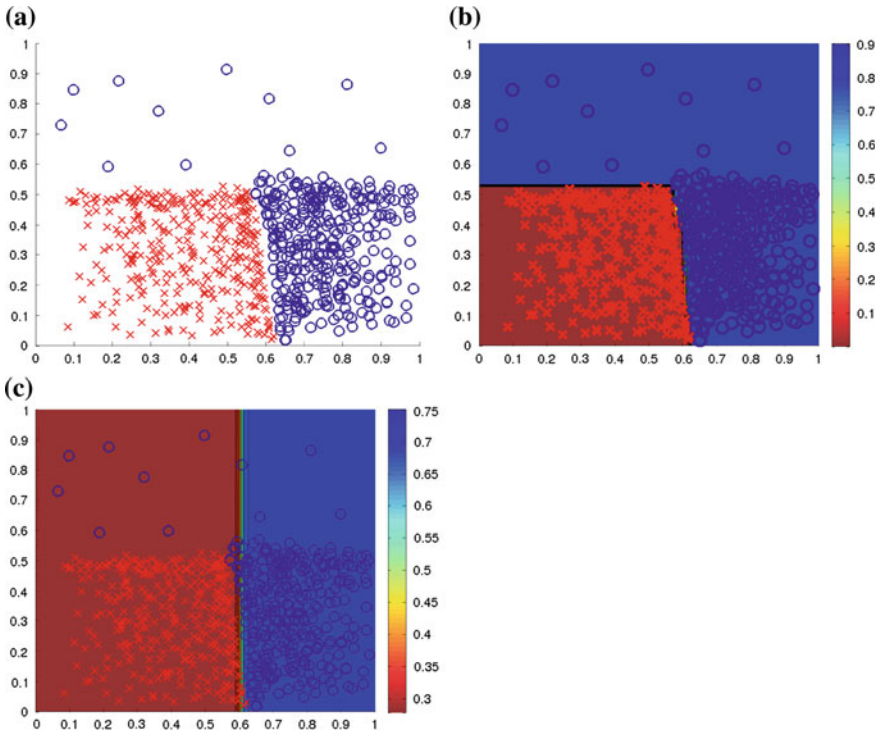


Fig. 2 Example with hard inliers. This toy problem shows how less dense, but learnable examples do not contribute to the decision boundary when learned using RobustBoost. The colour coding represents estimated probability $p(y = 1|x)$. (Best viewed in colour.) **a** Toy example without outliers, **b** Learnt AdaBoost classifier, **c** Learnt RobustBoost classifier

as a heavily quantized version of presenting the examples to the learning algorithm in the order of their difficulty.

3 Boosting Theory

Boosted strong classifiers have the form $H_m(x) = \sum_i^m \alpha_i h(x; \beta_i)$, where $h(x)$ is a weak learner, with multiplier α_i and parameters β_i. To learn such a classifier one wishes to minimize the average loss $\frac{1}{N} \sum_{j=1}^{N} L(H(x_j), y_j)$ over the N input data points (x_j, y_j) where each data label $y_j \in \{-1, 1\}$. Learning the classifier that minimizes the average loss by an exhaustive search is infeasible, so boosting algorithms do this in a greedy stepwise fashion. At each iteration the strong classifier is extended with the weak learner that minimizes the loss given the already learned strong classifier

$$\alpha^*, \beta^* = \underset{\alpha,\beta}{\operatorname{argmin}} \frac{1}{N} \sum_{j=1}^{N} L(H_m(x_j) + \alpha h(x_j; \beta), y_j). \tag{1}$$

Equation 1 is solved by weighting the importance of the input data by a weight function $w(x, y)$ when learning α and β. This $w(x_j, y_j)$ represents how poorly the current classifier $H_m(x_j)$ is able to classify example j.

Different boosting algorithms have different losses and optimizations procedures, but the key mechanism to their behaviour and handling of outliers is the weight function $w(x, y)$. For this reason we believe that analyzing the weight functions of different losses gives an insight to how different boosting algorithms behave in the presence of hard and ambiguous examples. So in order to compare a number of boosting algorithms in a consistent framework we re-derive $w(x, y)$ for each of the algorithms by following the GradientBoost approach [27, 28].

The GradientBoost approach views boosting as a gradient based optimization of the loss in function space. According to the GradientBoost framework a boosting algorithm can be constructed from any differentiable loss function, where each iteration is a combination of a least squares fitting of a weak regressor $h(x)$ to a target $w(x, y)$

$$\beta^* = \underset{\beta}{\operatorname{argmin}} \left(\sum_j \left(w(x_j, y_j) - h(x_j; \beta) \right)^2 \right), \tag{2}$$

and a line search $\alpha = \operatorname{argmin}_\alpha (L(H(x) + \alpha h(x; \beta)))$ to obtain α. The loss function is derived with respect to the current margin $v(x, y) = yH(x)$ to obtain the negative target function

$$w(x, y) = -\frac{\partial L(x, y)}{\partial v(x, y)}. \tag{3}$$

Equation 2 can then be interpreted as finding the weak learner that points in the direction of the steepest gradient of the loss, given the data.

3.1 Convex-Loss Boosting Algorithms

3.1.1 Exponential Loss Boosting

AdaBoost and GentleBoost [15, 29] are the most notable algorithms with the exponential loss

$$L_e(x, y) = \exp(-v(x, y)). \tag{4}$$

AdaBoost uses weak classifiers for $h(x)$ rather than regressors and directly solves for α, while GentleBoost employs Newton-step optimization for the expected loss. In the original algorithms $w(x, y)$ is exponential and comes in via the weighted fitting

of $h(x)$, but we obtain

$$w_e(x, y) = \exp(-v(x, y)), \tag{5}$$

from the GradientBoost approach to align all analyzed loss functions in the same framework. $w_e(x, y)$ has a slightly different meaning than the weight function of the original algorithms since it is the target of a non-weighted fit, rather than the weight of a weighted fit. However, its interpretation is the same—the importance function by which an example is weighted for the training of the weak learner $h(x)$. Also, it should be noted that we have omitted implementation-dependent normalization of the weight function.

3.1.2 Binomial Log-Likelihood Boosting

LogitBoost is a boosting algorithm that uses Newton stepping to minimize the expected value of the negative binomial log-likelihood

$$L_l(x, y) = \log(1 + \exp(-2v(x, y))). \tag{6}$$

This is potentially more resistant to outliers than AdaBoost or GentleBoost as the binomial log-likelihood is a much softer loss function than the exponential one [15].

Since the original LogitBoost optimizes this loss with a series of Newton steps, the actual importance of an example is distributed between a weight function for the weighted regression and a target for the regression—both varying with the margin of the example. We derive $w(x, y)$ by applying the GradientBoost approach to the binomial log-likelihood loss function to collect the example weight in one function

$$w_l(x, y) = \frac{1}{1 + \exp(v(x, y))}. \tag{7}$$

Figure 3a shows the different weight functions and suggests that LogitBoost should be affected less by examples far on the negative margin than the exponential-loss algorithms.

3.2 Robust Boosting Algorithms

3.2.1 RobustBoost

RobustBoost is specifically designed to handle outliers [10]. RobustBoost, a variation of BrownBoost, is based on the Boost-by-Majority algorithm and has a very soft and non-convex loss function

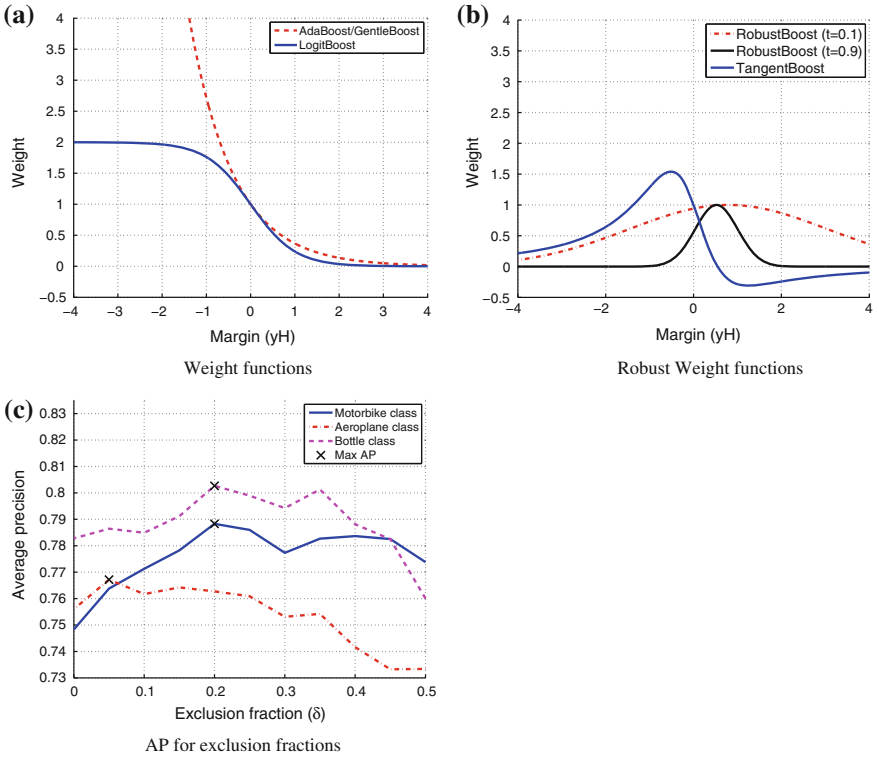


Fig. 3 **a, b** Weight functions with respect to the margin. This illustrates how much examples at different distances from the margin are able to affect the decision boundary for the different algorithms. **c** Performance for different exclusion fractions δ. Average precision on the test set, using the two-pass extension with LogitBoost for different exclusion fractions δ and different classes. This figure illustrates that different classes have different optimal exclusion fractions δ

$$L_r(x, y, t) = 1 - \text{erf}\left(\frac{v(x, y) - \mu(t)}{\sigma(t)}\right), \tag{8}$$

where $\text{erf}(\cdot)$ is the error function, $t \in [0, 1]$ is a time variable and $\mu(t)$ and $\sigma(t)$ are functions

$$\sigma^2(t) = (\sigma_f^2 + 1)\exp(2(1 - t)) - 1 \tag{9}$$

$$\mu(t) = (\theta - 2\rho)\exp(1 - t) + 2\rho, \tag{10}$$

with parameters θ, σ_f and ρ. Equation 8 is differentiated with respect to the margin to get the weight function

$$w_r(x, y, t) = \exp\left(-\frac{(v(x, y) - \mu(t))^2}{2\sigma(t)^2}\right). \tag{11}$$

Figure 3b shows Eq. 11 for some values of t. From these we can see the RobustBoost weight function changes over time. It is slightly more aggressive in the beginning and as $t \to 1$, it focuses less and less on examples far away from the target margin θ. One interpretation is that the algorithm focuses on all examples early in the training stage, and as the algorithm progresses it starts ignoring examples that it has not been able to push close to the target margin.

RobustBoost is self-terminating in that it finishes when $t \geq 1$. In our experiments we follow Freund's example and set $\sigma_f = 0.1$ to avoid numerical instability for t close to 1 and we obtain the parameters θ and ρ by cross-validation.

3.2.2 TangentBoost

TangentBoost was designed to have a positive bounded loss function for both positive and negative large margins, where the maximum loss for large positive margins is smaller than for large negative margins [12]. To satisfy these properties the method of probability elicitation [6] is followed to define TangentBoost to have a tangent link function

$$f(x) = \tan\left(p(x) - 0.5\right), \tag{12}$$

and a quadratic minimum conditional risk

$$C_L^*(x) = 4p(x)(1 - p(x)), \tag{13}$$

where $p(x) = \arctan(H(x)) + 0.5$. is the intermediate probability estimate. Combining the above equations results in the Tangent loss

$$L_t(x, y) = \left(2\arctan\left(v(x, y)\right) - 1\right)^2. \tag{14}$$

We immediately see that the theoretical derivation of TangentBoost and its implementation may have to differ as the probability estimates $p(x) \in [-\pi/2+0.5, \pi/2+0.5]$ are not proper, so that we only have proper probabilities $p(x) \in [0, 1]$ for $|H(x)| < 0.546$. This means that $|H(x)| > 0.546$ has to be handled according to some heuristic, which is not presented in the original paper [12]. In the original paper the Tangent loss is optimized through Gauss steps, which similarly to LogitBoost divides the importance of examples into two functions. So as with the other algorithms we re-derive $w_t(x, y)$ by using the GradientBoost method, and obtain

$$w_t(x, y) = -\frac{4\left(2\arctan\left(v(x, y)\right) - 1\right)}{1 + \left(v(x, y)\right)^2}. \tag{15}$$

As seen in Fig. 3b this weight function gives low weights for examples with large negative margin, but it also penalizes large positive margins by assigning negative weight to very confident examples. Since $w_t(x, y)$ is actually the regression target

this means that the weak learner tries to fit very correct examples to an incorrect label.

4 A Two-Pass Exclusion Extension

Our main point is that some fraction of the data, that is easy to learn with a given feature and weak-learner set, defines the core shape of the class—we call these examples inliers. Then there are examples that are ambiguous or uncommon so that they cannot be properly learned given the same representation. Trying to do so might lead to overfitting, create artefacts or force a poorer definition of the shape of the core of the class. We call these examples non-learnable or outliers and illustrate their effect on training in Fig. 4. It is important to note that there might be hard examples with large negative margin during some parts of training, but that eventually get learned without overfitting. We refer to these examples as hard inliers, and believe they are important for learning a well performing classifier.

Figure 4 illustrates that even if robust algorithms are better at coping with outliers, they are still negatively affected by them in two ways; The outliers still have an effect on the decision boundary learnt, even if their effect is reduced. Hard inliers are also subject to the robust losses, thus having less influence over the decision boundary than for non-robust losses, illustrated in Fig. 2.

We propose that outliers and inliers should be identified and handled separately so that the outliers are only allowed to influence the training when already close to the decision boundary and therefore can be considered as part of the core shape of the class. This can be achieved with a very soft loss function, such as the logistic loss or the Bayes consistent Savage loss [6]. We use the logistic loss, since the Savage loss gives more importance to slightly misclassified examples, rather than being symmetric around the margin.

Differentiating the logistic loss

$$L_s(x, y) = \frac{1}{1 + \exp\left(-\eta v(x, y)\right)}, \tag{16}$$

with respect to the margin results in the weight function

$$w_{excl}(x, y) = \eta\, \sigma(-\eta\, v(x, y))\, \sigma(\eta\, v(x, y)) \tag{17}$$

where $\sigma(\cdot)$ is the sigmoid function. This weight function can be made arbitrarily thin by increasing the η parameter. We call this function the *exclusion function*, as its purpose is to exclude outliers from training.

Since the inlier examples are considered learnable we want the difficult examples in the inlier set to have high weight, according to the original idea of boosting. For this reason all inliers should be subjected to a more aggressive loss such as the exponential loss or the binomial log-likelihood loss.

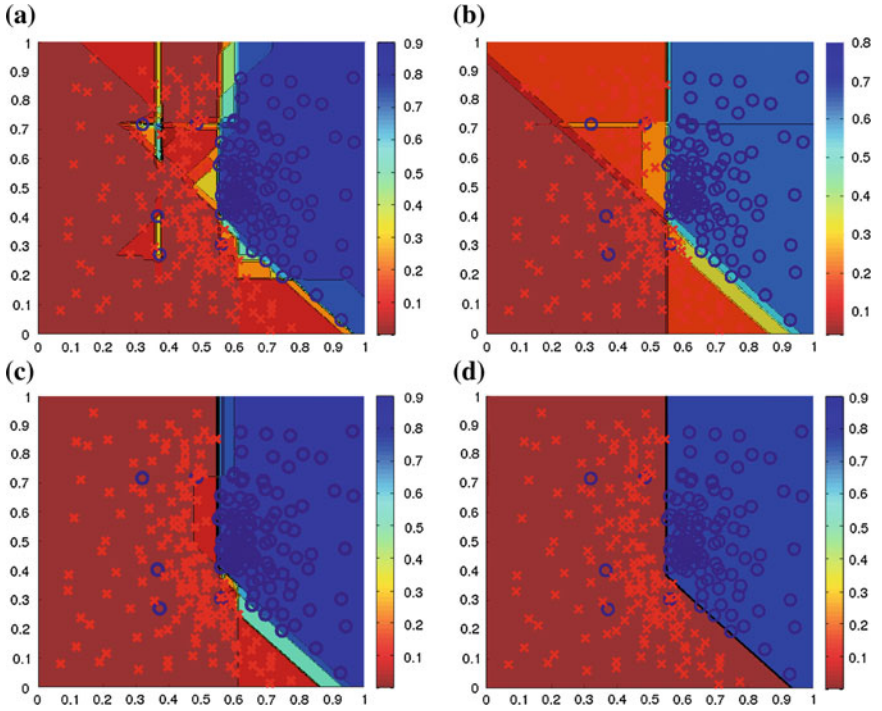


Fig. 4 Example with five outliers. The decision boundaries the different algorithms produce in the presence of a few outliers. The colour coding represents estimated probability $p(y = 1|x)$. We can see how AdaBoost overfits to the outliers, TangentBoost overfits slightly less and RobustBoost is able to handle the problem, even if it is less certain around the boundary. Applying the two-pass method to this problem results in a decision boundary that completely ignores the outliers. (Best viewed in colour.) **a** AdaBoost, **b** TangentBoost, **c** RobustBoost, **d** AdaBoost with two-pass method

The main challenge is to identify the outliers in a dataset. To do this we follow our definition of outliers as non-learnable and say that they are the examples with the lowest confidence after completed training. We therefore define the steady-state difficulty $d(x_j)$ of the examples as their negative margin $-v(x_j, y_j)$ after a fully completed training round, and normalize to get non-negative values.

$$d(x_j) = \begin{cases} \max(H(x)) - H(x_j) & \text{if } y_j = 1 \\ H(x_j) - \min(H(x)) & \text{if } y_j = -1, \end{cases} \tag{18}$$

where $H(x_j)$ is the classification score of example j. This is referred to as the first pass.

We order the positive examples according to their difficulty $d(x_j)$ and re-train the classifier, assigning a fraction δ of the most difficult examples to the outlier set and subjecting them to the logistic loss function. This second iteration of training

is what we call the second pass. Figure 1 shows some inlier and outlier examples for the bottle and chair classes. As we have mentioned, what will be considered an outlier depends on the features used. We use HOG in our experiments [2], so it is expected that the tilted bottles and the occluded ones are considered outliers, since HOG cannot capture such variation well.

As previously mentioned, our model for outlier exclusion has two parameters: δ and η, where δ controls how many examples will be considered as outliers and η controls how aggressively the outlier examples will be down-weighted. In our experiments we choose a large value for η—effectively ignoring outliers completely in the second round. The actual fraction of outliers is both class and feature dependent, so δ needs to be properly tuned. We tune δ by cross-validation, yet we have noticed that simple heuristics seem to work quite well too. Figure 3c shows how the performance is affected by δ for three different classes. We can clearly see that different classes have different optimal values for δ, which is related to the number of outliers in their datasets, given the used features and learners.

4.1 Inverted Cascade

The excluded examples are not necessarily label noise, but are defined as examples that increase the cross-validation error, they can still hold valuable information about the class. One way of making use of this information without reducing the performance of the exclusion-type classifier is to learn separate classifiers, and take the *max* of their classification scores as output. Since there is a difficulty ordering between the two classifiers, it is more natural to approach this problem as an inverted cascade. An inverted cascade is a cascade where each positive classification aborts the cascade instead of negative classifications as in standard cascades. Each negative classification continues to the next classifier. This way the learnable examples can be correctly classified by the first classifier while hard false negatives can be subjected to stricter classification criteria in a later cascade stage.

5 Experiments

We perform experiments on a large number of classes to reduce the influence of random performance fluctuations. For this reason, and due to the availability of a test set we select the VOC2007 dataset for our experiments. Positive examples have bounding-box annotations, so we crop and resize all positive examples to have the same patch size. Since our main focus is to investigate how our two-pass exclusion method improves learning, we want to minimize variance or tweaking in the areas not related to learning, and therefore choose a static non-deformable feature. We use the HOG descriptor [2] to describe the patches, since it has shown good performance

in the past, is popular within the vision community, and its best parameter settings have been well established.

To get the bounding boxes for the negative examples, we generate random positions, widths and heights. We make sure that box sizes and aspect ratios are restricted to values that are reasonable for the positive class. Image patches are then cropped from the negative training set according to the generated bounding boxes. The patches are also resized to have the same size as the positive patches, after which the HOG features are computed. We then apply our two-pass training procedure described in Sect. 4 to train a boosted stump classifier.

We train boosted stumps using four different boosting algorithms: AdaBoost, LogitBoost, TangentBoost and RobustBoost. AdaBoost and LogitBoost are chosen for their popularity and proliferation in the field. TangentBoost and RobustBoost are chosen as they explicitly handle outliers. We also train an SVM classifier to use as reference.

For LogitBoost, TangentBoost and AdaBoost there are no parameters to be set. For RobustBoost two parameters have to be tuned: the error goal ρ and the target margin θ. We tune them by holdout cross-validation on the training set. With TangentBoost we encountered an implementation issue, due to the possibility of negative weights and improper probability estimates. After input from the author of [12] we manually truncate the probability estimates to make them proper. Unfortunately this forces example weights to zero for margins $|v(x, y)| > 0.546$, which gives poor results as this quickly discards a large portion of the training set. To cope with this and to obtain reasonable performance we lower the learn rate of the algorithm. The linear SVM is trained with *liblinear* [30], with normalized features and using 5-fold cross-validation to tune the regularization term C.

Jittering the positive examples is a popular way of bootstrapping the positive dataset, but we believe that this can also generate examples that are not representative of that class. For this reason our two-pass approach should respond well to jittered datasets. We therefore redo the same experiments for the best performing classifier, augmenting the positive sets by randomly generating 2 positive examples per labelled positive example, with small random offsets in positions of the bounding box. We also mine for hard negatives to get a complete picture of how the outlier exclusion extension interacts with bootstrapping methods.

6 Results

A summary of our results is that all boosting algorithms except TangentBoost show consistent improvements for the experiments using our two-pass extension, which can be seen in Table 1. Before employing our two-pass extension LogitBoost performs best with 11 wins over the other algorithms. After the two-pass outlier exclusion LogitBoost dominates even more with 15 wins over other outlier-excluded algorithms and 13 wins over all other algorithms, including LogitBoost without

Table 1 Performance of our experiments

Class	Mean average precision for each algorithm																	
	Adaboost			LogitBoost			RobustBoost			TangentBoost			Linear SVM			LogitBoost inverted cascade		
	Use all	Exclude	Diff	Use all	Exclude	Diff	Use all	Exclude	Diff	Use all	Exclude	Diff	Use all	Exclude	Diff	Result	Improvement
Plane	**74.27**	73.87	−0.40	73.13	**74.09**	0.97	**72.49**	72.23	−0.27	**73.86**	72.88	−0.98	**74.04**	72.01	−2.03	79.03	5.90
Bike	87.59	**88.24**	0.66	87.85	**88.43**	0.58	86.86	**87.98**	1.12	**86.81**	86.80	−0.01	**85.12**	84.47	−0.66	88.57	0.72
Bird	46.69	**48.14**	1.45	48.08	**49.87**	1.79	46.66	**48.69**	2.03	46.97	**49.45**	2.48	**46.67**	46.32	−0.34	51.05	2.97
Boat	57.33	**58.96**	1.63	59.01	**60.81**	1.80	57.92	**58.31**	0.39	**58.61**	58.02	−0.59	**57.00**	56.94	−0.06	63.46	4.45
Bottle	74.57	**78.56**	3.99	76.02	**80.00**	3.98	72.74	**78.94**	6.20	76.79	**78.32**	1.53	76.22	**76.97**	0.75	80.23	4.21
Bus	**86.79**	86.18	−0.62	86.54	**86.96**	0.42	83.86	**86.82**	2.96	85.97	**86.26**	0.29	**82.07**	81.85	−0.22	88.63	2.09
Car	87.05	**87.73**	0.68	**88.32**	**88.32**	0.00	**88.26**	88.14	−0.11	**88.34**	88.30	−0.04	**86.22**	85.93	−0.29	89.45	1.13
Cat	**49.45**	49.18	−0.27	50.15	**52.44**	2.29	48.30	**52.21**	3.91	45.93	**50.90**	4.97	45.45	**45.48**	0.02	54.79	4.64
Chair	67.68	**69.36**	1.68	68.26	**69.93**	1.66	66.06	**68.91**	2.85	67.42	**68.09**	0.67	**64.08**	63.94	−0.14	71.69	3.43
Cow	81.90	**83.25**	1.35	81.99	**83.85**	1.87	81.83	**82.71**	0.87	**81.50**	80.63	−0.87	82.01	**82.44**	0.43	85.32	3.33
Table	40.89	**44.50**	3.61	43.68	**47.91**	4.24	39.89	**46.26**	6.37	**47.52**	36.34	−11.18	**28.35**	28.09	−0.26	50.95	7.27
Dog	47.83	**51.56**	3.73	51.27	**51.68**	0.41	47.66	**52.24**	4.58	**51.25**	50.21	−1.04	46.16	**46.94**	0.78	56.24	4.97
Horse	76.09	**76.10**	0.01	78.83	**78.84**	0.01	77.72	**78.28**	0.56	75.20	**76.20**	1.00	75.55	**75.77**	0.21	78.83	0.00
Motorbike	74.15	**77.15**	3.00	77.39	**78.33**	0.94	76.75	**79.20**	2.45	75.16	**75.46**	0.30	**73.31**	72.56	−0.76	79.31	1.92

(continued)

Table 1 (Continued)

Class	Mean average precision for each algorithm																
	Adaboost			LogitBoost			RobustBoost			TangentBoost			Linear SVM			LogitBoost inverted cascade	
	Use all	Exclude	Diff	Use all	Exclude	Diff	Use all	Exclude	Diff	Use all	Exclude	Diff	Use all	Exclude	Diff	Result	Improvement
Person	58.56	**63.64**	5.08	65.16	**67.03**	1.87	60.17	**65.09**	4.92	66.04	**67.02**	0.98	60.86	**61.84**	0.98	67.89	2.73
Plant	**58.08**	57.60	−0.48	60.46	**60.51**	0.04	56.87	**59.62**	2.75	**59.28**	58.23	−1.05	58.42	**58.90**	0.48	60.10	−0.36
Sheep	80.60	**84.41**	3.81	81.59	**83.11**	1.53	**80.78**	80.11	−0.66	**84.38**	82.01	−2.37	80.07	**80.17**	0.10	84.49	2.90
Sofa	61.83	**66.28**	4.44	62.35	**66.66**	4.31	63.92	**67.81**	3.89	56.73	**63.06**	6.33	**62.33**	62.18	−0.15	67.50	5.15
Train	73.18	**76.91**	3.74	73.98	**76.74**	2.75	73.13	**77.26**	4.13	73.87	**74.74**	0.87	71.82	**72.10**	0.28	78.74	4.76
Tv	**92.11**	**92.11**	0.00	**92.57**	**92.57**	0.00	**91.97**	**91.97**	0.00	**91.69**	**91.69**	0.00	**91.28**	**91.28**	0.00	92.65	0.08
Mean	68.83	**70.69**	1.85	70.33	**71.90**	1.57	68.69	**71.14**	2.45	69.67	**69.73**	0.06	**67.35**	67.31	−0.04	73.45	3.11
WWA	4	15	–	0	18	–	3	16	–	9	10	–	10	9	–	–	–
WBA	2	1	–	11	15	–	1	4	–	5	0	–	1	0	–	–	–

Average Precisions of different classifiers, when applied to the VOC2007 test set. The gray box on each row indicates the best performing classifier for the object class. Boldface numbers indicate best within-algorithm-performance for excluding outliers or not. Gray cells indicate total best performance for a given class—not including the inverted cascade. *Wins within algorithm* (*WWA*) summarizes how often a learning method is improved by our extension, and *Wins between algorithms* (*WBA*) summarizes how often an algorithm outperforms the others when having the same strategy for handling outliers. Note that these results cannot be directly compared to results from the original VOC2007 challenge since we are performing image patch classification, using the annotated bounding boxes to obtain positive object positions. The inverted cascade achieves the highest performance and the last column shows its improvement over the base algorithm

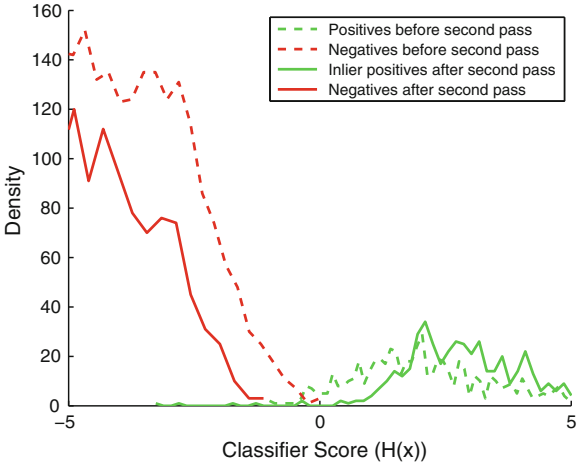


Fig. 5 Margin before and after outlier exclusion. The margin for the learnable examples becomes wider when using the outlier exclusion extension

outlier exclusion. Figure 5 shows an example of how the margin becomes much wider when using the extension.

The inverted cascade with LogitBoost performs the best overall with the most significant improvements, both over the base algorithm but also over the outlier-excluded one.

6.1 Comparison of Boosting Algorithms

Table 1 also includes a comparison of the performance between algorithms, showing performance differences with and without the outlier exclusion. Among the convex-loss algorithms LogitBoost performs better than AdaBoost. The difference in performance shrinks when our two-pass method is applied, which suggests that naturally occurring outliers in real-world vision data affects the performance of boosting algorithms and that those better able to cope with such outliers have a greater potential for good performance.

Even so, the robust algorithms perform worse than LogitBoost. One reason for this could be that the robust algorithms make no distinction between outliers and hard inliers, as previously discussed. Our two-pass algorithm only treats "non-learnable" examples differently, not penalizing unkommon learnable examples for being difficult in the early stages of learning.

Although RobustBoost has inherent robustness, it is improved the most by our extension. One explanation is its variable target error rate ρ, which after the exclusion of outliers obtains a lower value through cross-validation. RobustBoost with a small

value ρ is more similar to a non-robust algorithm, and should not suffer as much from the hard-inlier-problem demonstrated in Fig. 2c.

The SVM classifier is provided as a reference and sanity check, and we see that the boosting algorithms give superior results even though only decision stumps are used.

The higher performance of the boosted classifier is likely due to that combination of decision stumps can produce more complex decision boundaries than the hyperplane of a linear SVM. It is not surprising that the linear SVM is not improved by the outlier exclusion as it has a relatively soft hinge loss, tuned soft margins, and lacks the iterative reweighting of examples and greedy strategy, that our argumentation is based on. SVMs suffer from outliers too [17], but our method is not optimal for SVMs, since some of the excluded examples could be important support vectors.

6.2 Bootstrapping Methods in Relation to Outlier Exclusion

We can see in Table 2 that jittering has a positive effect on classifier performance and that our outlier exclusion method improves that performance even more. This shows that our two-pass outlier exclusion is complementary to hard negative mining and positive jittering and could be considered as a viable data augmentation technique when using boosting algorithms.

7 Discussion and Future Work

We show that all boosting methods perform better when handling outliers separately during training. As RobustBoost and TangentBoost do not reach the performance of LogitBoost they might be too aggressive in reducing the importance of hard inliers and not aggressive enough for outliers. We must remember that the problem posed by the VOC2007 dataset does not include label noise, but does definitely have hard and ambiguous examples that might interfere with learning the optimal decision boundary. RobustBoost and other robust boosting algorithms have previously shown good results on artificially flipped labels, but we believe that a more common problem in object classification is naturally occurring ambiguous examples and we have therefore not focused on artificial experiments where labels are changed at random.

We notice that the improved performance from excluding hard and ambiguous examples is correlated with the severity of the loss function of the method. AdaBoost, with its exponential loss function, shows large improvement while LogitBoost has less average gain and TangentBoost gains almost nothing from the exclusion of outliers. More surprising is that RobustBoost is improved the most, in spite of its soft loss function. One explanation is that there is an additional mechanism at work in improving the performance of RobustBoost. RobustBoost is self-terminating, stopping when it has reached its target error. When training on a dataset with a

Table 2 Outlier exclusion with other dataset augmentation techniques

	Mean average precision for each object class																			
Method	Plane	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Motor-bike	Person	Plant	Sheep	Sofa	Train	Tv	Mean
Default	73.13	**87.85**	48.08	59.01	76.02	86.54	88.32	50.15	68.26	81.99	43.68	51.27	78.83	77.39	65.16	60.46	81.59	62.35	73.98	92.57	70.33
J	75.64	87.33	49.34	57.59	76.68	88.69	88.29	49.95	68.28	83.63	48.25	51.87	79.78	77.83	66.17	**63.59**	82.86	64.51	**78.08**	92.55	71.55
J + OE	77.18	87.56	50.30	60.32	79.54	**89.73**	89.62	50.64	68.75	84.73	50.39	**55.09**	79.33	**79.68**	**67.73**	62.83	83.71	**67.65**	75.79	92.55	72.66
J + HN	76.31	87.35	**50.85**	**61.10**	80.08	89.41	89.62	51.23	68.93	84.88	49.35	53.08	80.90	79.67	65.78	61.98	81.39	64.07	76.18	**92.98**	72.26
J + HN + OE	**77.73**	87.46	48.60	59.76	**80.67**	89.37	**90.07**	**53.76**	**70.11**	**86.55**	**51.24**	53.99	**80.92**	77.78	67.54	62.39	**85.60**	66.83	77.38	92.98	**73.04**

Positive jittering (J) and hard negative mining (HN) improves performance even more in combination with outlier exclusion (OE)

smaller fraction non-learnable examples, a lower target error is better suited and the cross-validation process will ensure that a lower target error is selected. This will result in later termination and an overall more aggressive loss function.

We have seen that pruning of the hard-to-learn examples in a dataset without label noise can lead to improved performance, contrary to the "more data is better" philosophy. This point has also been examined by [17], who demonstrate that removing perfectly fine examples can improve overall classification performance in the case where model flexibility is insufficient to adapt to all training data.

Our belief is that our method removes bad data, but also that it reduces the importance of examples that make the learning more difficult, in this way allowing the boosting algorithms to find better local minima. We still believe that more data is better if properly handled, so this first approach of selective example exclusion should be extended in the future, and might potentially combine well with positive example mining, especially in cases where the quality of the positives cannot be guaranteed.

8 Conclusions

We provide an analysis of several boosting algorithms and their sensitivity to outlier data. Following this analysis we propose a two-pass training extension that can be applied to boosting algorithms to improve their tolerance to naturally occurring outliers. We show experimentally that excluding the hardest positives from training by subjecting them to an exclusive weight function is beneficial for classification performance. Introducing an inverted cascade to re-include the hard positives we improve the performance even more. We also show that this effect is complementary to jittering and hard negative mining, which are common bootstrapping techniques.

The main strength of our approach is that classification performance can be improved without any extra test-time cost, only at the expense of training-time cost. If test time can be increased slightly the inverted cascade allows additional performance. We believe that handling the normal and hard examples separately might allow bootstrapping of training sets with less accurate training data.

We also present results on the VOC2007 dataset, comparing the performance of different boosting algorithms on real world vision data, concluding that LogitBoost performs the best and that some of this difference in performance can be due to its ability to better cope with naturally occurring outlier examples.

Acknowledgments This work has been funded by the Swedish Foundation for Strategic Research (SSF); within the project VINST

References

1. Felzenszwalb, P., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. PAMI (2010)
2. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
3. Laptev, I.: Improving object detection with boosted histograms. IVC (2009)
4. Kumar, M.P., Zisserman, A., Torr, P.H.S.: Efficient discriminative learning of parts-based models. In: ICCV (2009)
5. Long, P.M., Servedio, R.A.: Random classification noise defeats all convex potential boosters. In: ICML (2008)
6. Masnadi-shirazi, H., Vasconcelos, N.: On the design of loss functions for classification: theory, robustness to outliers, and savageboost. In: NIPS (2008)
7. Leistner, C., Saffari, A., Roth, P.M., Bischof, H.: On robustness of on-line boosting—a competitive study. In: ICCV Workshops (2009)
8. Bauer, E., Kohavi, R.: An empirical comparison of voting classification algorithms: bagging, boosting, and variants. MLJ (1999)
9. Dietterich, T.: An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. MLJ (2000)
10. Freund, Y., Science, C.: A more robust boosting algorithm. arXiv:0905.2138 (2009)
11. Freund, Y.: An adaptive version of the boost by majority algorithm. In: COLT (1999)
12. Masnadi-Shirazi, H., Mahadevan, V., Vasconcelos, N.: On the design of robust classifiers for computer vision. In: CVPR (2010)
13. Grove, A., Schuurmans, D.: Boosting in the limit: maximizing the margin of learned ensembles. In: AAAI (1998)
14. Warmuth, M., Glocer, K., Rätsch, G.: Boosting algorithms for maximizing the soft margin. In: NIPS (2008)
15. Friedman, J., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. AOS (2000)
16. Vijayanarasimhan, S.: Large-scale live active learning: training object detectors with crawled data and crowds. In: CVPR (2011)
17. Zhu, X., Vondrick, C., Ramanan, D., Fowlkes, C.C.: Do we need more training data or better models for object detection? In: BMVC (2012)
18. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth images. In: CVPR (2011)
19. Torralba, A., Fergus, R., Freeman, W.T.: 80 million tiny images: a large data set for nonparametric object and scene recognition. PAMI (2008)
20. Hays, J., Efros, A.: Scene completion using millions of photographs. TOG (2007)
21. Freund, Y.: Boosting a weak learning algorithm by majority. IANDC (1995)
22. Rätsch, G., Onoda, T., Müller, K.: Soft margins for AdaBoost. MLJ (2001)
23. Vezhnevets, A., Barinova, O.: Avoiding boosting overfitting by removing confusing samples. In: ECML (2007)
24. Angelova, A., Abu-Mostafam, Y., Perona, P.: Pruning training sets for learning of object categories. In: CVPR (2005)
25. Viola, P., Platt, J.: Multiple instance boosting for object detection. In: NIPS (2006)
26. Kumar, M.P., Packer, B.: Self-paced learning for latent variable models. In: NIPS (2010)
27. Friedman, J.: Greedy function approximation: a gradient machine. AOS (2001)
28. Mason, L., Baxter, J., Bartlett, P., Frean, M.: Boosting algorithms as gradient descent in function space. In: NIPS (1999)
29. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. In: COLT (1995)
30. Fan, R., Chang, K., Hsieh, C., Wang, X., Lin, C.: LIBLINEAR: a library for large linear classification. JMLR (2008)

Discriminative Dimensionality Reduction for the Visualization of Classifiers

Andrej Gisbrecht, Alexander Schulz and Barbara Hammer

Abstract Modern nonlinear dimensionality reduction offers powerful techniques to directly inspect high dimensional data in the plane. Since the task of data projection is generally ill-posed and information loss cannot be avoided while projecting, the quality and meaningfulness of the outcome is not clear. In this contribution, we argue that discriminative dimensionality reduction, i.e. the concept to enhance the dimensionality reduction technique by supervised label information, offers a principled way to shape the outcome of a dimensionality reduction technique. We demonstrate the capacity of this approach for benchmark data sets. In addition, based on discriminative dimensionality reduction, we propose a pipeline how to visualize the function of general nonlinear classifiers in the plane. We demonstrate this approach by providing a generic visualization of the function of support vector machine classifiers.

Keywords Dimensionality reduction · Fisher information metric · Classifier visualization · Evaluation.

1 Introduction

The digitalization of almost all areas of daily life causes a dramatic increase of electronically available data sets. Due to improved sensor technology and dedicated storage formats, data sets are increasing with respect to both, size and complexity. This fact poses new challenges for data analysis: automated machine learning techniques such as clustering, classification, or visualization constitute standard approaches to automatically extract relevant information from the data. In recent time, however, methods have to deal with very large data sets such that many algorithms rely on

A. Gisbrecht (✉) · A. Schulz · B. Hammer
University of Bielefeld - CITEC Centre of Excellence, Bielefeld, Germany
e-mail: agisbrec@techfak.uni-bielefeld.de

A. Schulz
e-mail: aschulz@techfak.uni-bielefeld.de

B. Hammer
e-mail: bhammer@techfak.uni-bielefeld.de

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_3

sampling or approximation techniques to maintain feasibility [2, 29]. In addition, an exact objective is often not clear a priori; rather, the user specifies her interests and demands interactively when applying data mining techniques and inspecting the results [17, 36]. This places the human into the loop, causing the need for intuitive interfaces to the machine learning scenarios [28, 32, 33].

The visual system constitutes one of our most advanced senses. Humans possess astonishing cognitive capabilities as concerns e.g. visual grouping or recognition of artifacts, according to the Gestalt principles of visual perception [14]. Relying on these principles and the efficiency of human visual perception, visualization plays an essential part in the context of the presentation of data and results to humans, and in the enabling of efficient interactive machine learning systems. This causes the need for reliable, fast and online visualization techniques of data and machine learning results when training on the given data.

Dimensionality reduction refers to the specific task to map high dimensional data points into low dimensions such that data can directly be displayed on the screen while as much information as possible is preserved [19]. Classical techniques such as principle component analysis (PCA) offer a linear projection only, thus their flexibility is limited. Nevertheless, they are widely used today due to their excellent generalization ability and scalability. In recent years, a large variety of nonlinear alternatives has been proposed, formalizing the ill-posed objective of dimensionality reduction while preserving as much information as possible via different mathematical objectives. Maximum variance unfolding, non-parametric embedding, Isomap, locally linear embedding (LLE), stochastic neighbor embedding (SNE), and similar, constitute just a few popular examples, see e.g. the overviews [5, 19, 31]. Still, many practitioners rely on simpler linear techniques such as PCA for practical applications e.g. in the biomedical context [3]. The reasons for this fact are twofold. On the one hand, nonlinear dimensionality reduction techniques are often computationally costly, and they provide a mapping of the given data points only, requiring additional effort for out-of-sample extensions. On the other hand, the results of nonlinear dimensionality reduction are often not easily interpretable by humans. Even a universally accepted formal quantitative evaluation of the quality of a projection is not yet available, first steps into the direction of principled evaluation measures for dimensionality reduction having just recently been proposed [20].

In this contribution, we address the problem of shaping the ill-posed problem of dimensionality reduction by means of an integration of often easily accessible auxiliary information: a class labeling of the given data, more formally, discriminative dimensionality reduction. In this setting, the goal is to visualize those aspects of the data which are of particular relevance for the given labeling, as specified by the practitioner. Hence this auxiliary information allows to explicitly specify which information is regarded as relevant to the user: the aspects relevant for the given labeling; versus which aspects can be neglected while projecting the data: all aspects which do not change the shape of the data with respect to the given labeling.

A few approaches have been proposed in this context: classical Fisher's linear discriminant analysis (LDA) projects data such that within class distances are minimized while between class distances are maximized, still relying on a linear mapping. The

objective of partial least squares regression (PLS) is to maximize the covariance of the projected data and the given auxiliary information. It is also suited if data dimensionality is larger than the number of data points. Informed projections [8] extend PCA to minimize the sum squared error and the mean value of given classes, this way achieving a compromise of dimensionality reduction and clustering. Reference [24] extends a dimensionality reduction mapping which is based on deep autoencoders by auxiliary function information learned in parallel. In [11], the metric is adapted according to auxiliary class information prior to projection to yield a global linear matrix transform. Further, interesting extensions of multidimensional scaling to incorporate class information have recently been proposed [37]. Modern techniques extend these settings to general nonlinear projections of data. One way is offered by kernelization such as kernel LDA [1, 21, 23]. The approaches introduced in [15, 22] can both be understood as extensions of SNE. Multiple relational embedding (MRE) incorporates several dissimilarity structures in the data space induced by labeling, for example, into one latent space representation. Colored MVU incorporates auxiliary information into MVU by substituting the raw data by the combination of the data and the covariance matrix induced by the given auxiliary information.

Another principled way to extend dimensionality reducing data visualization to auxiliary information is offered by an adaptation of the underlying metric. The principle of learning metrics has been introduced in [18, 25]: the Riemanian metric on the given data manifold is substituted by an adapted form which measures the information of the data for the given classification task [18, 25, 27, 34]. A slightly different approach is taken in [9], relying on an ad hoc adaptation of the metric. Metric adaptation based on the classification margin and subsequent visualization has been proposed in [6], for example. Alternative approaches to incorporate auxiliary information modify the cost function of dimensionality reducing data visualization. In this contribution, we will rely on the principle of learning metrics based on estimations of the Fisher information, and we will integrate this approach into a kernel extension of t-distributed stochastic neighbor embedding, yielding excellent results.

What are the differences of a supervised visualization as compared to a direct classification of the data, i.e. a simple projection of the data points to their corresponding class labels? We will argue that auxiliary information in the form of class labeling can play a crucial role when addressing dimensionality reduction: it offers a natural way to shape the inherently ill-posed problem of dimensionality reduction by explicitly specifying which aspects of the data are relevant and, in consequence, which aspects should be emphasized—those aspects of the data which are relevant for the given auxiliary class labeling. In addition, the integration of auxiliary information can help to solve the problem of the computational complexity of dimensionality reduction. In this contribution, we will show that discriminative dimensionality reduction can be used to infer a mapping of points based on a small subsample of data only, thus reducing the complexity by an order of magnitude. We will use this technique in a general framework which allows us to visualize not only a given labeled data set, rather full classification models can be displayed this way, as we will demonstrate for SVM.

Albeit classification constitutes one of the standard tasks in data analysis, generic techniques to visualize classifiers are often rather restricted. At present, the major way to display the result of a classifier and to judge its suitability is by means of the classification accuracy. This quantitative evaluation, however, does not offer any intuitive interpretation of the result at all. It is not clear how to inspect the complexity of the class boundary, the presence of outliers, the question whether classes overlap etc. based on the classification accuracy only. Visualization could offer an intuitive access to these questions, by directly displaying the data points and classification boundary on the plane.

Still, visualization is used in only a few places when inspecting a classifier: in a low dimensional space, a direct visualization of the data points and classification boundaries in 2D or 3D can be done. For high dimensional data, which constitutes the standard case, a direct visualization of the classifier is not possible and dimensionality reduction techniques are required. One line of research addresses visualization techniques to accompany the accuracy by an intuitive interface to set certain parameters of the classification procedure, such as e.g. ROC curves to set the desired specificity, or more general interfaces to optimize parameters connected to the accuracy [13]. Surprisingly, there exists relatively little work to visualize the underlying classifier itself for high dimensional settings. For the popular support vector machine, for examples, only some specific approaches have been proposed: one possibility is to let the user decide an appropriate linear projection dimension by means of tour methods [7]. As an alternative, some techniques rely on the distance of the data points to the class boundary and present this information using e.g. nomograms [16] or by using linear projection techniques on top of this distance [26]. A few nonlinear techniques exist such as SVMV [35], which visualizes the given data by means of a self-organizing map and displays the class boundaries by means of sampling. Further, very interesting nonlinear dimensionality reduction, albeit not for the primary aim of classifier visualization, has been introduced in [4]. These techniques offer first steps to visually inspect an SVM solution such that the user can judge e.g. remaining error regions, the modes of the given classes, outliers, or the smoothness of the separation boundary based on a visual impression.

However, so far, these techniques are often only linear, they require additional parameters, and they provide combinations of a very specific classifier such as SVM and a specific visualization technique. Discriminative dimensionality reduction constitutes an important general technique based on which a given classifier can be visualized. Here, we propose a principled alternative based on discriminative t-SNE with the Fisher metric, which allows to visualize any given classifier provided the method does not only output the class label, but some security of the classification such as e.g. the distance to the decision boundary or a possibly nonlinear, monotonic transformation thereof.

Now we will first introduce the Fisher metric as a general way to include auxiliary class labels into a non-linear dimensionality reduction technique. Thereby, we consider only one prototypical dimensionality reduction technique and emphasize the role of discriminative visualization rather than a comparison of the underlying dimensionality reduction technique. Hence we restrict to t-distributed stochastic neighbor

embedding (t-SNE), which constitutes one of the most successful nonlinear dimensionality reduction techniques used today [31]. All techniques could also be based on alternatives such as LLE or Isomap. We show the difference of the result from a direct classification in the context of discriminative t-SNE. Afterwards, we demonstrate the effect of the Fisher metric in nonlinear dimensionality reduction mapping, namely kernel t-distributed stochastic neighbor embedding. Finally, we display the suitability of discriminative dimensionality reduction to visualize classifiers in a generic way.

2 Supervised Visualization Based on the Fisher Information

Given a set of data points $\mathbf{x}_i$ in some high-dimensional data space X, t-SNE finds projections $\mathbf{y}_i$ for these points in the two dimensional plane $Y = \mathbb{R}^2$ such that the probabilities of data pairs in the original space and the projection space are preserved as much as possible. Probabilities in the original space are defined as $p_{ij} = (p_{(i|j)} + p_{(j|i)})/(2N)$ where N is the number of data points and

$$p_{j|i} = \frac{\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|\mathbf{x}_i - \mathbf{x}_k\|^2/2\sigma_i^2)}.$$

depends on the pairwise distance; the neighborhood range σ_i is automatically determined such that the effective number of neighbors coincides with a parameter, the perplexity. In the projection space, probabilities are defined as

$$q_{ij} = \frac{(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2)^{-1}}{\sum_{k \neq l}(1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2)^{-1}}.$$

Using student-t instead of Gaussians helps to avoid the crowding problem because of the involved long tails [31]. The goal of t-SNE is to find projectionts $\mathbf{y}_i$ such that the difference between p_{ij} and q_{ij} becomes small as measured by the Kullback-Leibler divergence. Usually, a gradient based optimization technique is used to minimize these costs. Considerable speed-up of the technique can be obtained based on approximations and tailored geometric data structures to handle distances in the low dimensional projections, see [38].

The goal of dimensionality reduction is inherently ill-posed: since there does not exist a loss-free representation of data in two-dimensions, information loss is inevitable while projecting. It depends on the actual setting which type of information is relevant for the application. A dimensionality reduction technique specifies which type of information is regarded as relevant by specifying a mathematical objective which is optimized while mapping. A formal mathematical cost function or training prescription, however, is hardly accessible to a user, and it cannot easily be influenced in terms of simple accessible parameters. This fact is one of the reasons that simple

but easily interpretable dimensionality reduction techniques such as PCA are often preferred in comparison to more powerful but not as easily interpretable ones.

To get around this problem, it has been proposed in [18, 25, 34] to enhance data by auxiliary information specified by the user. Here, we restrict to the particularly simple setting that auxiliary label information is given. Formally, we assume that every data point $\mathbf{x}_i$ is equipped with a class label c_i which are instances of a finite number of possible classes C. The task is to find projections $\mathbf{y}_i$ such that the aspects of $\mathbf{x}_i$ which are relevant for c_i are displayed. A given labeling induces a probability $p(c|\mathbf{x})$ given a data point $\mathbf{x}$. Albeit, in practice, such a probability is not available explicitly, it can be estimated based on the given data and labels using e.g. Parzen window estimates or any other suitable non-parametric estimator.

One can define a Riemannian manifold which is based on the information of $\mathbf{x}_i$ for the class labels in a canonic way. The tangent space at a point $\mathbf{x}_i$ of the data manifold is equipped with the quadratic form

$$d_{\mathbf{x}_i}(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{J}(\mathbf{x}_i)\mathbf{y}$$

where $\mathbf{J}(\mathbf{x})$ denotes the Fisher information matrix

$$\mathbf{J}(\mathbf{x}) = E_{p(c|\mathbf{x})}\left\{\left(\frac{\partial}{\partial \mathbf{x}} \log p(c|\mathbf{x})\right)\left(\frac{\partial}{\partial \mathbf{x}} \log p(c|\mathbf{x})\right)^T\right\}.$$

Note that this quadratic form scales dimensions locally according to its relevance for the class probabilities. Data dimensions are emphasized if and only if the label information changes rapidly along the considered axes, while dimensions along which the class labeling is constant do not contribute to the tensor.

Per definition, $\mathbf{J}(\mathbf{x})$ is symmetric and positive semidefinite such that it can serve as metric tensor. In addition, for typically smooth probability estimators $p(c|\mathbf{x})$, the parametrization is smooth such that $J(\mathbf{x})$ is C^∞, i.e. it can serve as tensor for a C^∞ manifold.

A Riemannian metric is induced from this tensor by taking minimum path integrals

$$d(\mathbf{x}, \mathbf{y}) = \inf_p \int_0^1 \sqrt{d_{p(t)}(p', p')}dt$$

for any two points $\mathbf{x}$ and $\mathbf{y}$ on the manifold. Here, the path $p : [0, 1] \to X$ ranges over all smooth curves with $p(0) = \mathbf{x}$ to $p(1) = \mathbf{y}$ in X. We refer to this metric as the Fisher metric in the following.

It offers a straightforward way to embed auxiliary information into t-SNE or any other dimensionality reduction technique which relies on distances: Instead of the standard Euclidean distance in X, we can use the Fisher distance to compare points in the high dimensional data space. This procedure transforms the high dimensional

space in such a way that those dimensions are locally emphasized, which influence the class labeling.

Note that the resulting distance measure is different from a simple classification of data, since it preserves important information such as e.g. the number of modes of a class, outliers, or regions with overlapping classes. This fact is demonstrated in a simple example in Fig. 1. Three classes which consist of two clusters each are generated in two dimensions. Thereby, the classes of two modes overlap (see arrow). We measure pairwise distances of these data using the Fisher metric, and display the resulting transformed space using classical multidimensional scaling. As can be seen, the following effects occur:

- the distance of data within a single mode belonging to one class becomes smaller by scaling dimensions which are unimportant for a given labeling at a smaller scale. Thus, data points in one clearly separated mode have the tendency to be mapped on top of each other, and these cluster structures become more apparent.
- the number of modes of the classes is preserved, emphasizing the overall structure of the class distribution in space—unlike a simple mapping of data to class labels which would map all modes of one class on top of each other.
- overlapping classes are displayed as such (see arrow) and directions which cause this conflict are preserved since they have an influence on the class labeling. In contrast, a direct mapping of such data to their class labels tries to resolve such conflicts in the data.

In practical applications, two approximations are necessary when computing the Fisher metric:

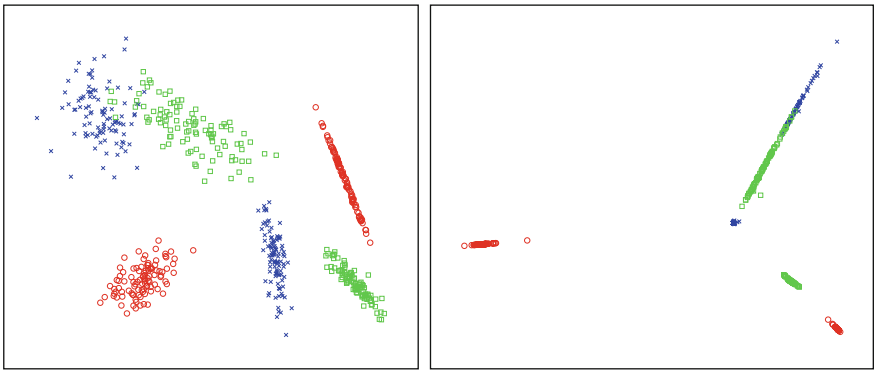


Fig. 1 A simple example which demonstrates important properties of the Fisher Riemannian tensor: multi-modality as well as class overlaps are preserved. The original data are displayed at the *left*, a plot of the data equipped with the Fisher metric displayed using metric multidimensional scaling is shown on the *right*, the *arrows point* to regions of overlap of the classes, which are preserved by the metric

2.1 Computation of the Class Probabilities

The conditional probabilities $p(c|\mathbf{x})$ are usually not available in closed form. However, they can always be estimated from the data using the Parzen nonparametric estimator

$$\hat{p}(c|\mathbf{x}) = \frac{\sum_i \delta_{c=c_i} \exp(-\|\mathbf{x} - \mathbf{x}_i\|^2/2\sigma^2)}{\sum_j \exp(-\|\mathbf{x} - \mathbf{x}_j\|^2/2\sigma^2)}.$$

The Fisher information matrix becomes

$$\mathbf{J}(\mathbf{x}) = \frac{1}{\sigma^4} E_{\hat{p}(c|\mathbf{x})} \left\{ \mathbf{b}(\mathbf{x}, c)\mathbf{b}(\mathbf{x}, c)^T \right\}$$

where

$$\mathbf{b}(\mathbf{x}, c) = E_{\xi(i|\mathbf{x},c)}\{\mathbf{x}_i\} - E_{\xi(i|\mathbf{x})}\{\mathbf{x}_i\}$$

$$\xi(i|\mathbf{x}, c) = \frac{\delta_{c,c_i} \exp(-\|\mathbf{x} - \mathbf{x}_i\|^2/2\sigma^2)}{\sum_j \delta_{c,c_j} \exp(-\|\mathbf{x} - \mathbf{x}_j\|^2/2\sigma^2)}$$

$$\xi(i|\mathbf{x}) = \frac{\exp(-\|\mathbf{x} - \mathbf{x}_i\|^2/2\sigma^2)}{\sum_j \exp(-\|\mathbf{x} - \mathbf{x}_j\|^2/2\sigma^2)}$$

E denotes the empirical expectation, i.e. weighted sums with weights depicted in the subscripts. If large data sets or out-of-sample extensions are dealt with, a subset of the data only is usually sufficient for the estimation of $\mathbf{J}(\mathbf{x})$.

2.2 Approximation of Minimum Path Integrals

Because of a general shape of the Riemannian tensor, path integrals and their minimization is usually impossible in closed form. Note that it is not necessary to compute exact values in particular for far apart points if distances are used for dimensionality reduction techniques; rather, the order of magnitude should be appropriate. There exist different efficient ways to approximate the path integrals based on the Fisher matrix as discussed in [25]. One possibility is offered by T-approximations: T equidistant points on the line from $\mathbf{x}_i$ to $\mathbf{x}_j$ are sampled, and the Riemannian distance on the manifold is approximated by

$$d_T(\mathbf{x}_i, \mathbf{x}_j) = \sum_{t=1}^{T} d_1 \left(\mathbf{x}_i + \frac{t-1}{T}(\mathbf{x}_j - \mathbf{x}_i), \mathbf{x}_i + \frac{t}{T}(\mathbf{x}_j - \mathbf{x}_i) \right)$$

where $d_1(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T J(\mathbf{x}_i)(\mathbf{x}_i - \mathbf{x}_j)$ is the standard distance as evaluated in the tangent space of $\mathbf{x}_i$. Locally, this approximation gives good results such that a faithful dimensionality reduction of data can be based thereon [25].

We integrate the Fisher information into a parametric kernel mapping, and in the visualization of classifiers, to demonstrate its benefits.

3 Training a Discriminative Visualization Mapping

Similar to many other nonlinear projection techniques, t-SNE has the drawback that it does not provide an explicit embedding mapping. First extensions towards parametric maps have been proposed in [5, 30]. The approach [5] has the drawback that it relies on locally linear maps, i.e. it has limited local flexibility. Reference [30] on the other hand uses deep autoencoders, i.e. very flexible maps, which require a large training set for good generalization. A compromise has been proposed in [10] which is based on kernel mappings. An explicit functional form is defined by

$$\mathbf{x} \mapsto \mathbf{y}(\mathbf{x}) = \sum_j \alpha_j \cdot \frac{k(\mathbf{x}, \mathbf{x}_j)}{\sum_l k(\mathbf{x}, \mathbf{x}_l)}.$$

The parameters $\alpha_j \in Y$ are points in the projection space. The centre points $\mathbf{x}_j$ are taken from a fixed sample of data points used to train the mapping. k is the Gaussian kernel.

Training takes place in two steps: First, an exemplary set of points $\mathbf{x}_i$ and projections $\mathbf{y}_i$ obtained by standard t-SNE (or any other dimensionality reduction technique). Afterwards, the parameters α_j can analytically be determined as the least squares solutions of these projections: the matrix $\mathbf{A}$ of parameters α_j is given by

$$\mathbf{A} = \mathbf{Y} \cdot \mathbf{K}^{-1}$$

where $\mathbf{K}$ is the normalized matrix with entries $k(\mathbf{x}_i, \mathbf{x}_j)/\sum_j k(\mathbf{x}_i, \mathbf{x}_j)$. $\mathbf{Y}$ denotes the matrix of projections $\mathbf{y}_i$, and $\mathbf{K}^{-1}$ refers to the pseudo-inverse.

This technology, referred to as kernel t-SNE, has the benefit that training can be done on a small subset of data only, extending the mapping to the full data set by means of the explicit mapping prescription. Thus, besides an explicit parametric form, a considerable speed up can be obtained.

We demonstrate the effect of incorporating the Fisher metric into the projection mapping using three examples. In all cases, we substitute the dimensionality reduction technique used to obtain the training set for kernel t-SNE by Fisher t-SNE. For the estimation of the Fisher information, 1 % of the data are used. For training the kernel mapping, 10 % of the data are used. Results are reported for the following three data sets:

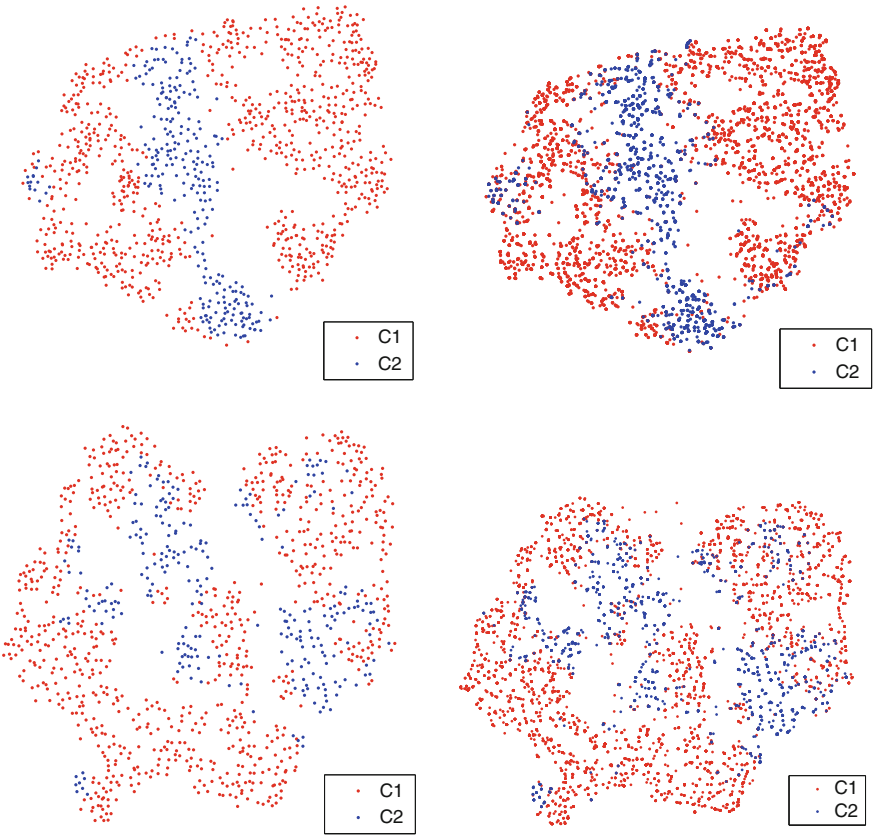


Fig. 2 Visualization of the ball data set using kernel t-SNE for the training set (*left*) and out of sample extension (*right*) with the Fisher metric (*top*) versus the standard Euclidean metric (*bottom*)

- The **ball** data consists of randomly generated points on the sphere where class labeling is given by two classes, induced by a band around the zero meridian.
- The **usps** data set consists of 11.000 points with dimensions representing handwritten digits from 0 to 9 [12].
- The **mnist** data set contains 60.000 points with 484 dimensions also representing handwritten digits.[1]

A visualization of the resulting projections for the training set and out of sample extension obtained with t-SNE and Fisher t-SNE is displayed in Fig. 2 for the ball data set, in Fig. 3 for the usps data set, and in Fig. 4 for the mnist data set. In all three settings, the generalization of the kernel t-SNE parametric mapping to novel data is excellent. As can be seen, the Fisher information accounts for clearer class boundaries and a clearer formation of cluster structures.

[1] http://yann.lecun.com/exdb/mnist/index.html.

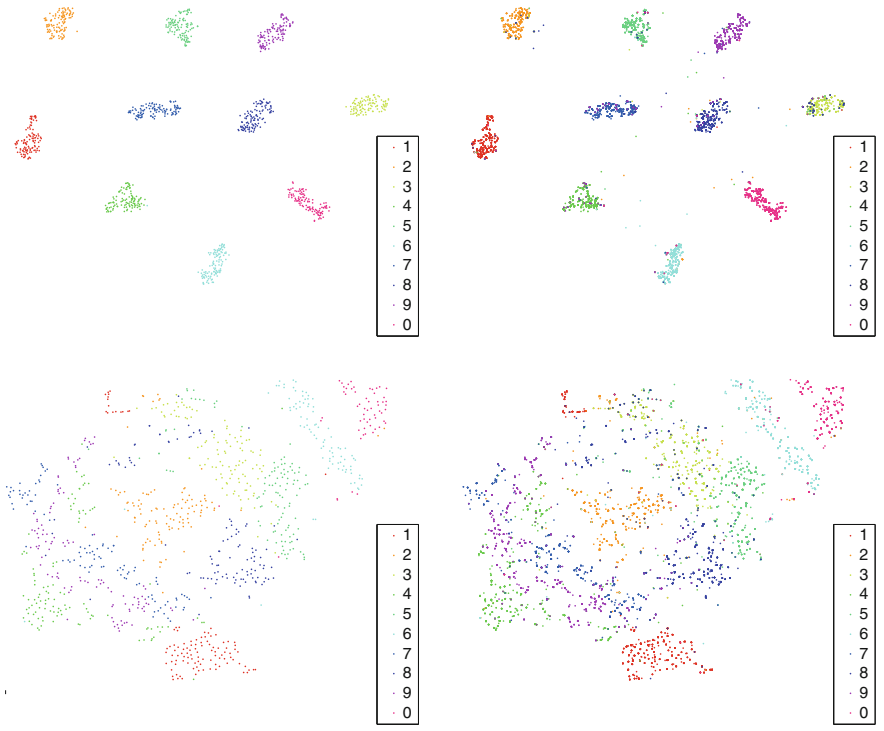


Fig. 3 Visualization of the usps data set using kernel t-SNE for the training set (*left*) and out of sample extension (*right*) with the Fisher metric (*top*) versus the standard Euclidean metric (*bottom*)

For the ball data, the class around the zero meridian stays more connected if the Fisher information is used as compared to a spread of the class for the standard Euclidean metric. Similarly, the classes observed for usps and mnist are clearer separated if the Fisher information is used. This finding can be accompanied by a numerical evaluation taking the k-nearest neighbor classification error of the embedded data. The results are displayed in Table 1. The classification accuracy is larger if Fisher information is included.

4 Visualization of Classifiers

We demonstrate the suitability of supervised dimensionality reduction to visualize general classifiers. For this purpose, we assume a classification mapping $f : X \to \{1, \ldots, C\}$ is present, which can be given by a support vector machine, for example. This mapping has been trained using some points $\mathbf{x}_i$ and their label c_i. In addition, we assume that the label prediction $f(\mathbf{x}_i)$ of a point $\mathbf{x}_i$ can be accompanied by a real value

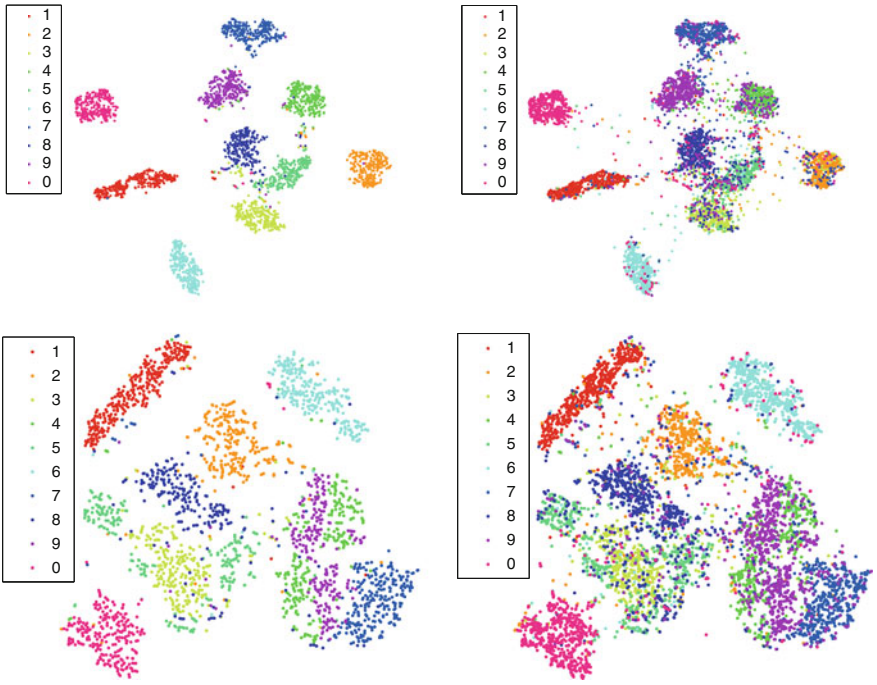


Fig. 4 Visualization of the mnist data set using kernel t-SNE for the training set (*left*) and out of sample extension (*right*) with the Fisher metric (*top*) versus the standard Euclidean metric (*bottom*)

Table 1 Classification accuracy of a k-NN classifier for the projected data sets

	k-tsne	Fisher k-t-SNE
Ball: test	0.9127	0.9153
train	0.9121	0.9764
mnist: test	0.8419	0.8813
train	0.9271	0.9608
usps: test	0.8496	0.8769
train	0.9273	0.9991

$r(\mathbf{x}_i) \in \mathbb{R}$ which indicates the (signed) strength of class-membership association. This can be induced by the class probability or ta monotonic transformation thereof. for example. Now the task is to map the points $\mathbf{x}_i$ and the classification boundary induced by f to 2D.

A very simple approach would consist in the following procedure: we sample the original space X and project these data $\mathbf{x}$ using a standard dimensionality reduction technique. Since smooth values $r(\mathbf{x})$ are present, isobars corresponding to the classifier can then be displayed in the plane. This naive approach encounters two problems: (i) sampling the original data space X is infeasible due to a usually high dimensionality and (ii) projecting exhaustive samples from high dimensions to 2D necessarily encounters loss of possibly relevant information.

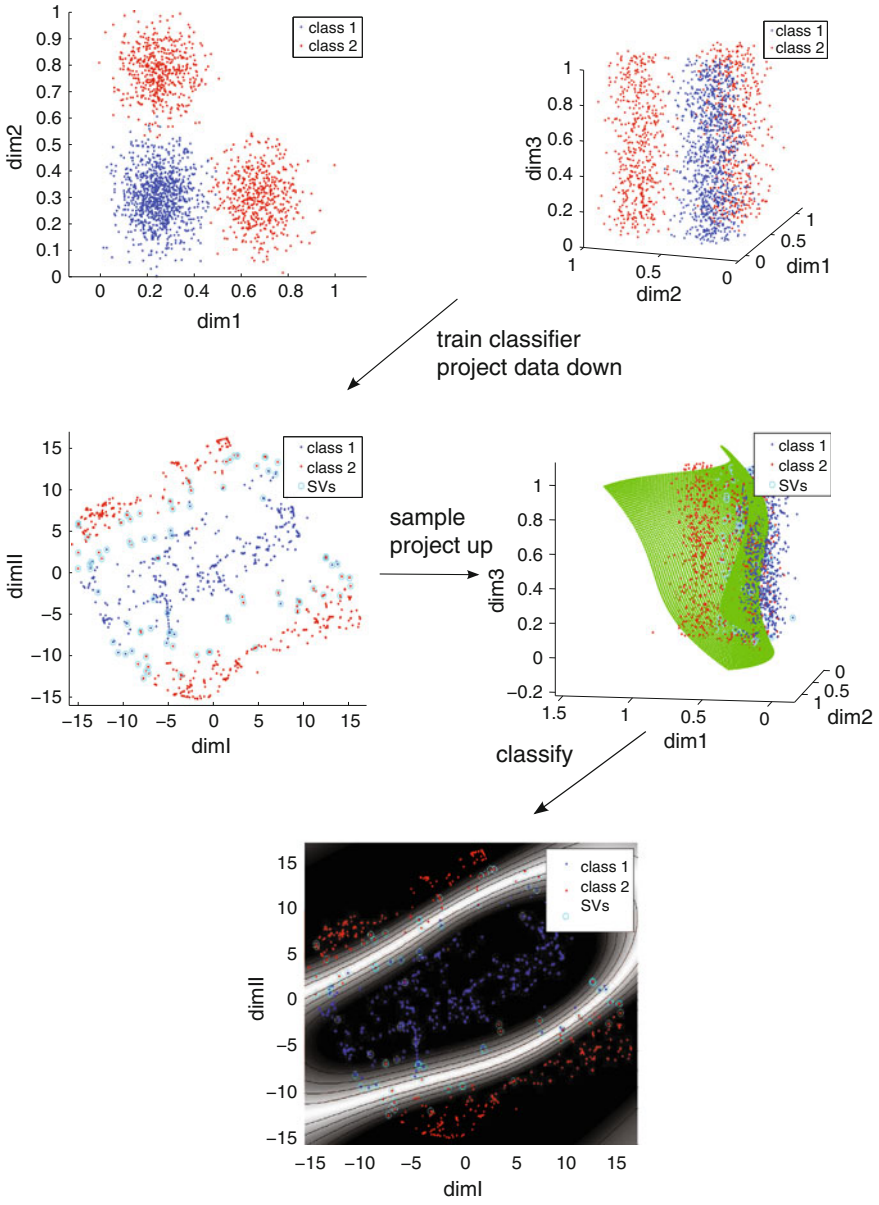


Fig. 5 Principled procedure how to visualize a given data set and a trained classifier. The example displays a SVM trained in 3D

These two problems can be avoided if label information is taken into account already at the dimensionality reduction step, and the procedure is slightly altered. We propose the following procedure as displayed in Fig. 5:

- Project the data $\mathbf{x}_i$ using a nonlinear discriminative visualization technique leading to points $p(\mathbf{x}_i) \in Y = \mathbb{R}^2$.
- Sample the projection space Y leading to points $\mathbf{z}'_i$. Determine points $\mathbf{z}_i$ in the data space X which are projected to these points $p(\mathbf{z}_i) \approx \mathbf{z}'_i$.
- Visualize the training points $\mathbf{x}_i$ together with the contours induced by the sampled function $(\mathbf{z}'_i, r(\mathbf{z}_i))$.

This procedure avoids the problems of the naive approach: a discriminative dimensionality reduction technique focusses on the aspects which are particularly relevant for the class labels and thus emphasizes the important characteristics of the classification function. On the other hand, sampling takes place in the projection space only, which is low dimensional. One open issue remains: we need to determine points $\mathbf{z}_i \in X$ which correspond to the projections $\mathbf{z}'_i \in Y$. Here, we take an approach similar to kernel t-SNE; we define a mapping

$$p^{-1} : Y \to X, \mathbf{y} \mapsto \sum_i \alpha_i \cdot \frac{k_i(\mathbf{y}_i, \mathbf{y})}{\sum_i k_i(\mathbf{y}_i, \mathbf{y})} = \mathbf{A} \cdot [\mathbf{K}]_i$$

of the projection space to the original space which is trained based on the given samples $\mathbf{x}_i$, its projections $\mathbf{y}_i$, and its labels c_i. As before, k is the Gaussian kernel, $\mathbf{K}$ the kernel matrix applied to the points $\mathbf{y}_i$ which are projections of $\mathbf{x}_i$ and $[\mathbf{K}]_i$ the ith column. $\mathbf{A}$ is the matrix of parameters α_i. These parameters α_i are determined by means of a numeric optimization of the error:

$$\lambda_1 \cdot \|\mathbf{X} - \mathbf{A} \cdot \mathbf{K}\|^2 + \lambda_2 \cdot \|r(\mathbf{X}) - r(\mathbf{A} \cdot \mathbf{K})\|^2$$

Thereby, $\mathbf{X}$ denotes the points $\mathbf{x}_i$ used to train the discriminative mapping. $r(\cdot)$ denotes real values associated to the classification f indicating the strength of the class-membership association. λ_1 and λ_2 are positive weights which balance the two objectives: a correct inverse mapping of the data $\mathbf{x}_i$ and its projections $\mathbf{y}_i$ on the one side and a correct match of the induced classifications as measured by r on the other side.

An example application is displayed in Fig. 6 for the ball data set, Fig. 7 for the usps data set, and Fig. 8 for mnist. For all settings, an SVM with Gaussian kernel is trained on a subset of the data which is not used to train the subsequent kernel t-SNE or Fisher kernel t-SNE, respectively. Thereby, we use two different t-SNE mappings to obtain the training set for the inverse mapping p^{-1}: kernel t-SNE and Fisher kernel t-SNE, respectively. The weights of the cost function has been chosen as $\lambda_1 = 0.1$ and $\lambda_2 = 10{,}000$, respectively.

Obviously, the visualization based on Fisher kernel t-SNE displays much clearer class boundaries as compared to a visualization which does not take the class labeling into account. Similarly, the classification of the Fisher-induced visualization better coincides with the classification of the observed SVM. This visual impression is

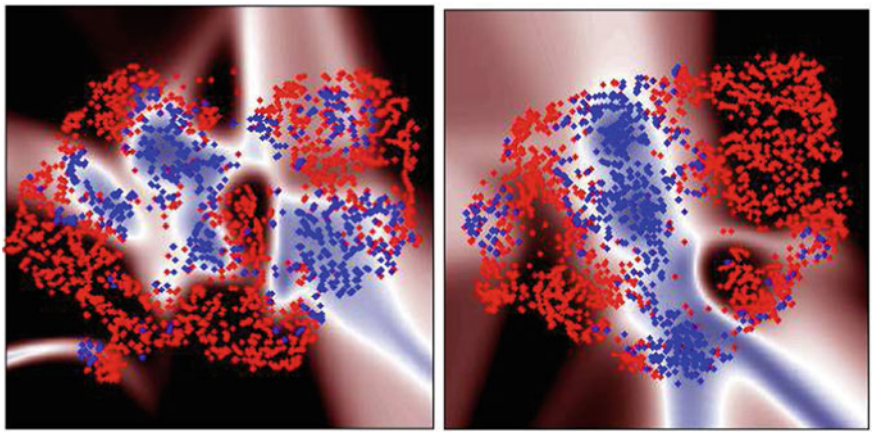


Fig. 6 Visualization of an SVM classifier trained on the ball data set by means of kernel t-SNE (*left*) and Fisher kernel t-SNE (*right*). The accuracy of the SVM is 95 %. The SVM classification coincides for 92 % of the data for the Fisher projection, and only for 87 % for the standard projection

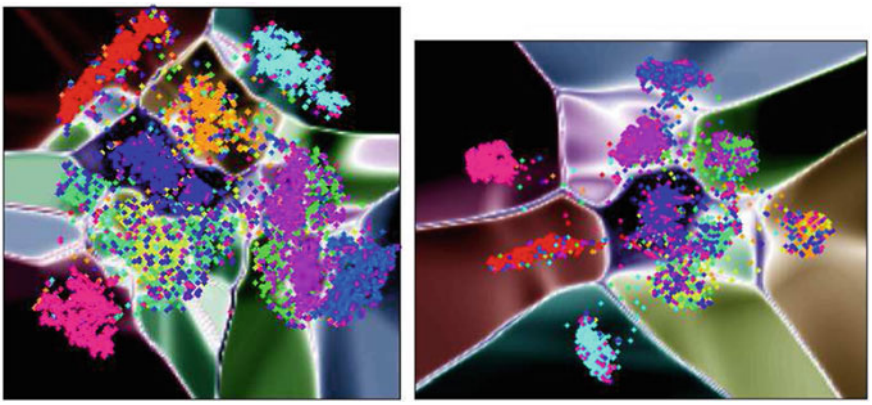


Fig. 7 Visualization of an SVM classifier trained on the mnist data set by means of kernel t-SNE (*left*) and Fisher kernel t-SNE (*right*). The accuracy of the SVM is 99 % on the training set and 97 % on the test set. The SVM classification coincides with 90 % of the data for the Fisher projection, and only 87 % for the standard projection

mirrored by a quantitative comparison of the projections, as detailed in the figure captions: including Fisher information always leads to an improved coincidence of the mapping accuracy and the original SVM, i.e. a projection which is more trustworthy as regards the overall accuracy.

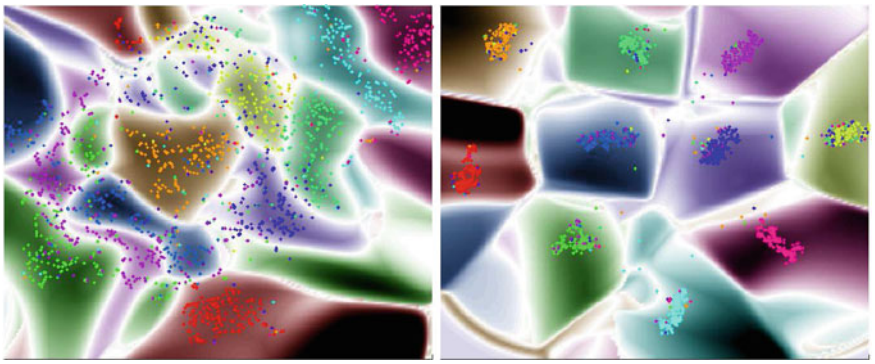


Fig. 8 Visualization of an SVM classifier with 99 % accuracy on the training set and 97 % accuracy on the test set for the usps data, the visualization is trained by means of kernel t-SNE (*left*) and Fisher kernel t-SNE (*right*). The accuracy of the SVM is 95 %. The SVM classification coincides with 92 % of the data for the Fisher projection, and only 85 % for the standard projection

5 Conclusions

We have introduced discriminative visualization by means of the Fisher information into kernel t-SNE mapping, and we have demonstrated the potential of this technique in the context of the visualization of classifiers. At present, we have restricted to SVM as one of the most popular classifiers, leaving the test of the technique for alternative classifiers as future work.

Acknowledgments Funding by DFG under grants number HA 2719/7-1, HA 2719/6-2 and by the CITEC centre of excellence are gratefully acknowledged.

References

1. Baudat, G., Anouar, F.: Generalized discriminant analysis using a kernel approach. Neural Comput. **12**, 2385–2404 (2000)
2. Bekkerman, R., Bilenko, M., Langford, J. (eds.): Scaling Up Machine Learning: Parallel and Distributed Approaches. Cambridge University Press, Cambridge (2011)
3. Biehl, M., Hammer, B., Merényi, E., Sperduti, A., Villmann, T. (eds.): Learning in the context of very high dimensional data (Dagstuhl Seminar 11341), vol. 1 (2011)
4. Braun, M.L., Buhmann, J.M., Müller, K.-R.: On relevant dimensions in kernel feature spaces. J. Mach. Learn. Res. **9**, 1875–1908 (2008)
5. Bunte, K., Biehl, M., Hammer, B.: A general framework for dimensionality reducing data visualization mapping. Neural Comput. **24**(3), 771–804 (2012)
6. Bunte, K., Schneider, P., Hammer, B., Schleif, F.-M., Villmann, T., Biehl, M.: Limited rank matrix learning, discriminative dimension reduction and visualization. Neural Netw. **26**, 159–173 (2012)

7. Caragea, D., Cook, D., Wickham, H., Honavar, V.: Visual methods for examining svm classifiers. In: Simoff, S.J., Böhlen, M.H., Mazeika, A. (eds.) Visual Data Mining. Lecture Notes in Computer Science, vol. 4404, pp. 136–153. Springer, Berlin (2008)
8. Cohn, D.: Informed projections. In: Becker, S., Thrun, S., Obermayer, K. (eds.) NIPS, pp. 849–856. MIT Press, Cambridge (2003)
9. Geng, X., Zhan, D.-C., Zhou, Z.-H.: Supervised nonlinear dimensionality reduction for visualization and classification. IEEE Trans. Syst. Man Cybern, Part B **35**(6), 1098–1107 (2005)
10. Gisbrecht, A., Mokbel, B., Hammer, B.: Linear basis-function t-sne for fast nonlinear dimensionality reduction. In: IJCNN (2013)
11. Goldberger, J., Roweis, S., Hinton, G., Salakhutdinov, R.: Neighbourhood components analysis. In: Advances in Neural Information Processing Systems 17, pp. 513–520. MIT Press, Cambridge (2004)
12. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer Series in Statistics. Springer, New York (2001)
13. Hernandez-Orallo, J., Flach, P., Ferri, C.: Brier curves: a new cost-based visualisation of classifier performance. In: Proceedings of International Conference on Machine Learning (2011)
14. Humphrey, G.: The psychology of the gestalt. J. Educ. Psychol. **15**(7), 401–412 (1924)
15. Iwata, T., Saito, K., Ueda, N., Stromsten, S., Griffiths, T.L., Tenenbaum, J.B.: Parametric embedding for class visualization. Neural Comput. **19**(9), 2536–2556 (2007)
16. Jakulin, A., Možina, M., Demšar, J., Bratko, I., Zupan, B.: Nomograms for visualizing support vector machines. In: KDD, pp. 108–117. ACM, New York (2005)
17. Kaski, S., Peltonen, J.: Dimensionality reduction for data visualization [applications corner]. IEEE Signal Process. Mag. **28**(2), 100–104 (2011)
18. Kaski, S., Sinkkonen, J., Peltonen, J.: Bankruptcy analysis with self-organizing maps in learning metrics. IEEE Trans. Neural Netw. **12**, 936–947 (2001)
19. Lee, J.A., Verleysen, M.: Nonlinear Dimensionality Reduction. Springer, New York (2007)
20. Lee, J.A., Verleysen, M.: Scale-independent quality criteria for dimensionality reduction. Pattern Recognit. Lett. **31**, 2248–2257 (2010)
21. Ma, B., Qu, H., Wong, H.: Kernel clustering-based discriminant analysis. Pattern Recognit. **40**(1), 324–327 (2007)
22. Memisevic, R., Hinton, G.: Multiple relational embedding. In: Saul, L.K., Weiss, Y., Bottou, L. (eds.) Advances in Neural Information Processing Systems 17, pp. 913–920. MIT Press, Cambridge (2005)
23. Mika, S., Rätsch, G., Weston, J., Schölkopf, B., Müller, K.-R.: Fisher discriminant analysis with kernels. In: Proceedings of IEEE, pp. 41–48 (1999)
24. Min, M.R., van der Maaten, L., Yuan, Z., Bonner, A.J., Zhang, Z.: Deep supervised t-distributed embedding. In: ICML, pp. 791–798 (2010)
25. Peltonen, J., Klami, A., Kaski, S.: Improved learning of riemannian metrics for exploratory analysis. Neural Netw. **17**, 1087–1100 (2004)
26. Poulet, F.: Visual svm. In: ICEIS. vol. 2, pp. 309–314 (2005)
27. Ruiz, H., Jarman, I.H., Martín, J.D., Lisboa, P.J.G.: The role of fisher information in primary data space for neighbourhood mapping. In: ESANN (2011)
28. Rüping, S.: Learning Interpretable Models. Ph.D. thesis, Dortmund University (2006)
29. Tsang, I.W., Kwok, J.T., ming Cheung, P., Cristianini, N.: Core vector machines: fast svm training on very large data sets. J. Mach. Learn. Res. **6**, 363–392 (2005)
30. van der Maaten, L.: Learning a parametric embedding by preserving local structure. J. Mach. Learn. Res. Proc. Track **5**, 384–391 (2009)
31. van der Maaten, L., Hinton, G.: Visualizing high-dimensional data using t-sne. J. Mach. Learn. Res. **9**, 2579–2605 (2008)
32. Vellido, A., Martín, J.D., Rossi, F., Lisboa, P.J.G.: Seeing is believing: the importance of visualization in real-world machine learning applications. In: ESANN (2011)
33. Vellido, A., Martin-Guerroro, J., Lisboa, P.: Making machine learning models interpretable. In: ESANN'12 (2012)

34. Venna, J., Peltonen, J., Nybo, K., Aidos, H., Kaski, S.: Information retrieval perspective to nonlinear dimensionality reduction for data visualization. J. Mach. Learn. Res. **11**, 451–490 (2010)
35. Wang, X., Wu, S., Wang, X., Li, Q.: Svmv—a novel algorithm for the visualization of svm classification results. In: Wang, J., Yi, Z., Zurada, J., Lu, B.-L., Yin, H. (eds.) Advances in Neural Networks—ISNN 2006. Lecture Notes in Computer Science, vol. 3971, pp. 968–973. Springer, Berlin (2006)
36. Ward, M., Grinstein, G., Keim, D.A.: Interactive Data Visualization: Foundations, Techniques, and Application. A.K. Peters Ltd, Natick (2010)
37. Witten, D.M., Tibshirani, R.: Supervised multidimensional scaling for visualization, classification, and bipartite ranking. Comput. Stat. Data Anal. **55**(1), 789–801 (2011)
38. Yang, Z., Peltonen, J., Kaksi, S.: Scalable optimization of neighbor embedding for visualization, In: ICML (2013)

Online Unsupervised Neural-Gas Learning Method for Infinite Data Streams

Mohamed-Rafik Bouguelia, Yolande Belaïd and Abdel Belaïd

Abstract We propose an unsupervised online learning method based on the "growing neural gas" algorithm (GNG), for a data-stream configuration where each incoming data is visited only once and used to incrementally update the learned model as soon as it is available. The method maintains a model as a dynamically evolving graph topology of data-representatives that we call neurons. Unlike usual incremental learning methods, it avoids the sensitivity to initialization parameters by using an adaptive parameter-free distance threshold to produce new neurons. Moreover, the proposed method performs a merging process which uses a distance-based probabilistic criterion to eventually merge neurons. This allows the algorithm to preserve a good computational efficiency over infinite time. Experiments on different real datasets, show that the proposed method is competitive with existing algorithms of the same family, while being independent of sensitive parameters and being able to maintain fewer neurons, which makes it convenient for learning from infinite data-streams.

Keywords Incremental learning · Unsupervised neural learning · Online learning · Data streams.

1 Introduction

Recently, research focused on designing efficient algorithms for learning from continuously arriving streams of data, in an incremental way, where each data can be visited only once and processed dynamically as soon as it is available. Particularly, unsupervised incremental neural learning methods take into account relations of

M.-R. Bouguelia (✉) · Y. Belaïd · A. Belaïd
Université de Lorraine - LORIA, UMR 7503, F-54506 Vandoeuvre-les-Nancy, France
e-mail: mohamed.bouguelia@loria.fr

Y. Belaïd
e-mail: yolande.belaid@loria.fr

A. Belaïd
e-mail: abdel.belaid@loria.fr

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_4

neighbourhood between representatives, and show a good clustering performance. Among these methods, GNG algorithm [1] has attracted considerable attention. It allows dynamic creation and removal of neurons (representatives) and edges between them during learning by maintaining a graph topology using a competitive Hebbian Learning strategy [2]. Each edge has an associated age which is used in order to remove old edges and keeps the topology dynamically updated. After adapting the graph topology using a fixed number of data-points from the input space (i.e. a time period), a new neuron is inserted between the two neighbouring neurons that cumulated the most important error. Unlike usual clustering methods (e.g. Kmeans), it does not require initial conditions such as a predefined number of representatives and their initialization. This represents an important feature in the context of data streams where we have no prior knowledge about the whole dataset. However, in GNG, the creation of a new neuron is made periodically, and a major disadvantage concerns the choice of this period. For this purpose, some adaptations that relaxes this periodical evolution have been proposed. The main incremental variants are IGNG [3], I2GNG [4] and SOINN [5]. Unfortunately, the fact that these methods depend on some sensitive parameters that must be specified prior to the learning, reduces the importance of their incremental nature. Moreover, large classes are unnecessarily modelled by many neurons representing many small fragments, and leading to a significant drop of computational efficiency over time.

In this paper we propose a GNG based incremental learning algorithm (AING) where the decision of producing a new neuron from a new coming data-point is based on an adaptive parameter-free distance threshold. The algorithm overcomes the shortcoming of excessive number of neurons by condensing them based on a probabilistic criterion, and building a new topology with a fewer number of neurons, thus preserving time and memory resources. The algorithm depends only on a parameter generated by the system requirements (e.g. allowed memory budget), and unlike the other algorithms, no parameter related to a specific characteristics dataset needs to be specified. Indeed, it can be really difficult for a user to estimate all the parameters that are required by a learning algorithm. According to [6], "A parameter-free algorithm would limit our ability to impose our prejudices, expectations, and presumptions on the problem at hand, and would let the data itself speak to us". An algorithm which uses as few parameters as possible without requiring prior knowledge is strongly preferred, especially when the whole dataset is not available beforehand (i.e. a data-stream configuration).

This paper is organized as follows. In Sect. 2, we describe a brief review of some incremental learning methods (mainly the GNG based ones), and analyse their problems. Then the algorithm we propose is presented in Sect. 3. In Sect. 4, we expose our experimental evaluation on synthetic and real datasets. In Sect. 5, we give the conclusion and present some perspectives of this work.

2 Related Work

Before describing some incremental methods and discussing their related problems, we firstly give some notations to be used in the rest of this paper: x refers to a data-point, y to a neuron, X_y is the set of data-points that are already assigned to neuron y, V_y is the set of current neurons that are neighbours of y (neurons linked to y by an edge), w_y is the reference vector of neuron y, and $n_y = |X_y|$ is the number of data-points currently assigned to y.

The basic idea of the Incremental Growing Neural Gas algorithm (IGNG) [3] is that the decision of whether a new coming data-point x is close enough to its nearest neurons is made according to a fixed distance threshold value T (Fig. 1a). Nevertheless, the main drawback of this approach is that the threshold T is globally the same for all neurons and must be provided as a parameter prior to the learning. There is no way to know beforehand which value is convenient for T, especially in a configuration where the whole dataset is not available.

I2GNG [4] is an improved version of IGNG where each neuron y has its own local threshold value (Fig. 1b) which is continuously adapted during learning. If there is currently no data-point assigned to a neuron y, then its associated threshold is a default value T which is an input parameter given manually as in IGNG; otherwise, the threshold is defined as $\bar{d} + \alpha\sigma$, where $\bar{d}$ is the mean distance of y to its currently assigned data-points, σ is the corresponding standard deviation, and α a parameter. Choosing "good" values for parameters T and α is important since the evolution of the threshold will strongly depends on them. For instance, if they are set at a relatively small value (depending on the dataset) then many unnecessary neurons are created. On the other hand, if their values are relatively big, then some data-points may wrongly be assigned to some close clusters. This clearly makes systems

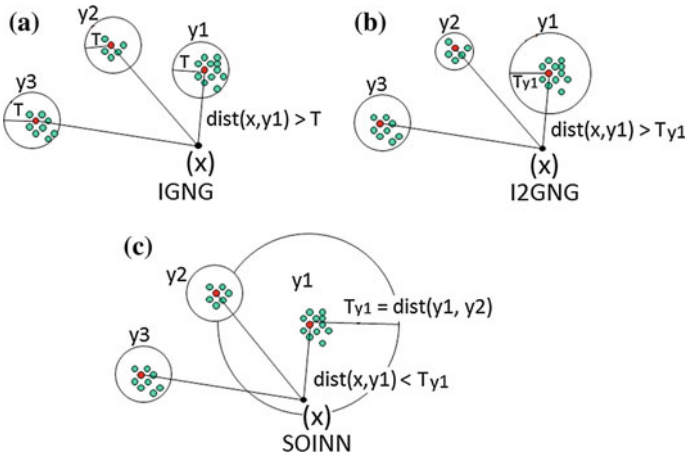


Fig. 1 Threshold based methods

using such an algorithm dependent on an expert user and gives less emphasis to its incremental nature.

In the Self-Organizing Incremental Neural Network (SOINN) [5], the threshold of a given neuron y is defined as the maximum distance of neuron y to its current neighbours if they exist, otherwise it is the distance of y to its nearest neuron among the existing ones (Fig. 1c). SOINN's threshold is often more sensitive to the creation order of neurons (induced by the arrival order of data-points), especially in first steps. Furthermore, SOINN deletes isolated neurons and neurons having only one neighbour when the number of input data-points is a multiple of a parameter λ (a period).

Many other parameter-driven methods have been designed especially for data stream clustering, among this methods we can cite: Stream [7], CluStream [8] and Density-Based clustering for data stream [9].

There are several variants of Kmeans that are said "incremental". The one proposed in [10] is based on a creation cost of cluster centers; the higher it is, the fewer is the number of created clusters. The cost is eventually incremented and the cluster centers are re-evaluated. However, the algorithm assumes that the size of the processed dataset is known and finite.

3 Proposed Algorithm (AING)

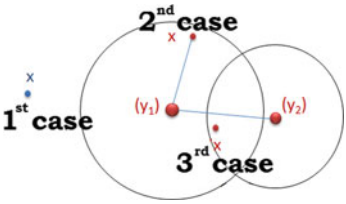
In this section, we propose a scalable unsupervised incremental learning algorithm that is independent of sensitive parameters, and dynamically creates neurons and edges between them as data come. It is called "AING" for Adaptive Incremental Neural Gas.

3.1 General Behaviour

The general schema of AING can be expressed according to the following three cases. Let y_1 and y_2 respectively be the nearest and the second nearest neurons from a new data-point x, such that $\text{dist}(y_1, x) < \text{dist}(y_2, x)$:

1. if x is *far enough* from y_1: a new neuron y_{new} is created at x (see Fig. 2, 1st case).
2. if x is *close enough* to y_1 but *far enough* from y_2: a new neuron y_{new} is created at x, and linked to y_1 by a new edge (see Fig. 2, 2nd case).
3. if x is *close enough* to y_1 and *close enough* to y_2 (see Fig. 2, 3rd case):
   - move y_1 and its neighbouring neurons towards x, i.e. modify their reference vectors to be less distant from x
   - increase the age of y_1's edges

Fig. 2 AING general cases



- link y_1 to y_2 by a new edge (reset its age to 0 if it already exists)
- activate the neighbouring neurons of y_1
- delete the old edges if any.

An age in this context is simply a value associated to each existing edge. Each time a data-point x is assigned to the winning neuron y_1 (the 3rd case), the age of edges emanating from this neuron is increased. Each time a data-point x is close enough to neurons y_1 and y_2, the age of the edge linking this two neurons is reset to 0. If the age of an edge continues to increase without being reset, it will reaches a maximum age value and the edge will be considered "old" and thus removed.

A data-point x is considered *far* (respectively *close*) enough from a neuron y, if the distance between x and y is higher (respectively smaller) than a threshold T_y. The following subsection shows how this threshold is defined.

3.2 AING Distance Threshold

Since the input data distribution is unknown, we define a parameter-free adaptive threshold T_y which is local to each neuron. The idea is to make the threshold T_y of a neuron y, dependent on the distances to data in its neighbourhood. The neighbourhood of y consists of data-points previously assigned to y (for which y is the nearest neuron), and data-points assigned to the neighbouring neurons of y (neurons that are linked to y by an edge).

According to formula 1, the threshold T_y of a neuron y is defined as the sum of distances from y to its data-points, plus the sum of weighted distances from y to its neighbouring neurons,[1] averaged on the total number of the considered distances. In the case where the neuron y has no data-points that were already assigned to it (X_y is empty) and has no neighbour (V_y is empty), then we consider the threshold T_y as the half distance from y to its nearest neuron.

$$T_y = \begin{cases} \dfrac{\sum\limits_{e \in X_y} \mathrm{dist}(y,e) + \sum\limits_{e \in V_y} |X_e| \times \mathrm{dist}(y,e)}{|X_y| + \sum\limits_{e \in V_y} |X_e|} & \text{if } X_y \neq \emptyset \vee V_y \neq \emptyset \\ \dfrac{\mathrm{dist}(y,\tilde{y})}{2},\ \tilde{y} = \underset{\tilde{y} \neq y}{\mathrm{argmin}}\ \mathrm{dist}(y,\tilde{y}) & \text{otherwise} \end{cases} \tag{1}$$

[1] The distance is weighted by the number of data-points associated to the neighbouring neuron.

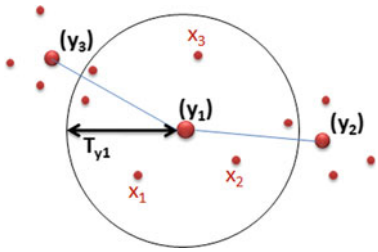


Fig. 3 AING threshold definition

Note that we do not need to save data-points that are already seen in order to compute this threshold. It is incrementally computed each time a new data-point comes, by updating some information associated to each neuron (e.g. number of data-points associated to a neuron, the sum of their distances to this neuron, etc.). If we consider the example of Fig. 3, there are 3 data-points assigned to y_1 (namely x_1, x_2 and x_3), and two neurons that are neighbours of y_1 (namely y_2 with 4 assigned data-points, and y_3 with 5 data-points). In this case, the threshold associated to the neuron y_1 is computed as

$$T_{y_1} = \frac{\text{dist}(y_1, x_1) + \text{dist}(y_1, x_2) + \text{dist}(y_1, x_3) + 4\,\text{dist}(y_1, y_2) + 5\,\text{dist}(y_1, y_3)}{3 + 4 + 5}$$

As we can see, the proposed threshold is independent of parameters and evolves dynamically according to the data and the topology of neurons.

3.3 AING Merging Process

Since data is processed online, it is usually common that algorithms for data stream clustering generate many cluster representatives. However, this may significantly compromise the computational efficiency over time. Instead of introducing parameters in the threshold computation to control the number of created neurons, AING can eventually reduce the number of neurons through the merging process. Indeed, when the number of current neurons reaches an upper bound (*up_bound*), some close neurons can be merged.

The merging process globally follows the same scheme as previously, but instead of relying on a hard rule based on a threshold, it uses a more relaxed rule based on a probabilistic criterion. Saying that "a neuron y is *far enough* from its nearest neuron $\tilde{y}$" is expressed as the probability that y will not be assigned to $\tilde{y}$, according to the formula $P_{y,\tilde{y}} = \frac{|X_y| \times \text{dist}(y,\tilde{y})}{\kappa}$. This probability is proportional to the distance between the two neurons ($\text{dist}(y, \tilde{y})$) and to the number of data-points assigned to y ($|X_y|$), that is, the more y is large and far from $\tilde{y}$, the more likely it is to remain not merged. The probability is in contrast inversely proportional to a variable κ, which means that by incrementing κ, any given neuron y will have more chance to be

merged with its nearest neuron. Let $\bar{d}$ be the mean distance of all existing neurons to the center-of-mass of the observed data-points. κ is incremented by $\kappa = \kappa + \bar{d}$ each time the neurons need to be more condensed, i.e. until the merging process takes effect and the number of neurons becomes less than the specified limit up_bound. Note that $P_{y,\tilde{y}}$ as specified may be higher than 1 when κ is not yet sufficiently big; a better formulation would be $P_{y,\tilde{y}} = \min(\frac{|X_y| \times \text{dist}(y,\tilde{y})}{\kappa}, 1)$, to guarantee it to be always a true probability.

The merging process is optional. Indeed, up_bound can be set to $+\infty$ if desired. Alternatively, the merging process can be triggered at any time chosen by the user, or by choosing the parameter up_bound according to some system requirements such as the memory budget that we want to allocate for the learning task, or the maximum latency time tolerated by the system due to a high number of neurons.

Finally, the code is explicitly presented in Algorithms 1 and 2, which provide an overall insight on the AING method. They both follow the same scheme described in Sect. 3.1. Algorithm 1 starts from scratch and incrementally processes each data-point from the stream using the adaptive distance threshold described in Sect. 3.2. When the number of current neurons reaches a limit, Algorithm 2 is called and some neurons are grouped together using the probabilistic criterion described in Sect. 3.3. We just need to point out two additional details appearing in our algorithms:

- If a data-point x is close enough to its two nearest neurons y_1 and y_2, it is assigned to y_1 and the reference vector of this later and its neighbours are updated (i.e. they move towards x) by a learning rate: ϵ_b for y_1 and ϵ_n for its neighbours (lines 15–17 of Algorithm 1). Generally, a too big learning rate implies instability of neurons, while a too small learning rate implies that neurons do not learn enough from their assigned data. Typical values are $0 < \epsilon_b \ll 1$ and $0 < \epsilon_n \ll \epsilon_b$. In AING, $\epsilon_b = \frac{1}{|X_{y_1}|}$ is slowly decreasing proportionally to the number of data-points associated to y_1, i.e. the more y_1 learns, the more it becomes stable, and ϵ_n is simply heuristically set to 100 times smaller than the actual value of ϵ_b (i.e. $\epsilon_n \ll \epsilon_b$).
- Each time a data-point is assigned to a winning neuron y_1, the age of edges emanating from this neuron is increased (line 14 of Algorithm 1). Let n_{max} the maximum number of data-points assigned to a neuron. A given edge is then considered "old" and thus removed (line 19 of Algorithm 1) if its age becomes higher than n_{max}. Note that this is not an externally-set parameter, it is the current maximum number of data-points assigned to a neuron among the existing ones.

Algorithm 1: AING Algorithm (up_bound).

init graph G with the two first coming data-points
$\kappa = 0$
while some data-points remain unread **do**
    get next data-point x, update $\bar{d}$ accordingly
    let y_1, y_2 the two nearest neurons from x in G
    get T_{y_1} and T_{y_2} according to formula 1
    if $dist(x, w_{y_1}) > T_{y_1}$ **then**
        $G \leftarrow G \cup \{y_{new}/w_{y_{new}} = x\}$
    else
        if $dist(x, w_{y_2}) > T_{y_2}$ **then**
            $G \leftarrow G \cup \{y_{new}/w_{y_{new}} = x\}$
            connect y_{new} to y_1 by an edge of age 0
        else
            increase the age of edges emanating from y_1
            let $\epsilon_b = \frac{1}{|X_{y_1}|}, \epsilon_n = \frac{1}{100 \times |X_{y_1}|}$
            $w_{y_1} += \epsilon_b \times (x - w_{y_1})$
            $w_{y_n} += \epsilon_n \times (x - w_{y_n}), \forall y_n \in V_{y_1}$
            connect y_1 to y_2 by an edge of age 0
            remove old edges from G if any
        end if
    end if
    while number of neurons in $G >$ up_bound **do**
        $\kappa = \kappa + \bar{d}$
        $G \leftarrow$ Merging(κ, G)
    end while
end while

4 Experimental Evaluation

4.1 Experiments on Synthetic Data

In order to test AING's behaviour, we perform an experiment on artificial 2D data of 5 classes (Fig. 4a) composed of a Gaussian cloud, a uniform distribution following different shapes, and some uniformly distributed random noise. Figure 4b, c show the topology of neurons obtained without using the merging process ($up\_bound = +\infty$), whereas for Fig. 4d, e, the merging process was also considered. However, for Fig. 4b, d, the data were given to AING class by class in order to test the incremental behaviour of AING. The results show that AING perfectly learns the topology of data and confirms that it has good memory properties.
On the other hand, for Fig. 4c, e the arrival order of data was random. The results show that AING performs well, even if the arrival order of data is random.

Algorithm 2: Merging (κ, G).

init $\tilde{G}$ with two neurons chosen randomly from G
for all $y \in G$ **do**
  let $\tilde{y}_1$, $\tilde{y}_2$ the two nearest neurons from y in $\tilde{G}$
  let $d_1 = dist(w_y, w_{\tilde{y}_1}), d_2 = dist(w_y, w_{\tilde{y}_2})$
  if $\underset{uniform}{\text{random}}([0, 1]) < \min(\frac{n_y \times d_1}{\kappa}, 1)$ **then**
    $\tilde{G} \leftarrow \tilde{G} \cup \{\tilde{y}_{new} / w_{\tilde{y}_{new}} = w_y\}$
  else
    if $\underset{uniform}{\text{random}}([0, 1]) < \min(\frac{n_y \times d_2}{\kappa}, 1)$ **then**
      $\tilde{G} \leftarrow \tilde{G} \cup \{\tilde{y}_{new} / w_{\tilde{y}_{new}} = w_y\}$
      connect $\tilde{y}_{new}$ to $\tilde{y}_1$ by an edge of age 0
    else
      increase age's edges emanating from $\tilde{y}_1$
      Let $\epsilon_b = \frac{1}{|X_{\tilde{y}_1}|}, \epsilon_n = \frac{1}{100 \times |X_{\tilde{y}_1}|}$
      $w_{\tilde{y}_1} += \epsilon_b \times (w_y - w_{\tilde{y}_1})$
      $w_{\tilde{y}_n} += \epsilon_n \times (w_y - w_{\tilde{y}_n}), \forall \tilde{y}_n \in V_{\tilde{y}_1}$
      connect $\tilde{y}_1$ to $\tilde{y}_2$ by an edge of age 0
      remove old edges from $\tilde{G}$ if any
    end if
  end if
end for
return $\tilde{G}$

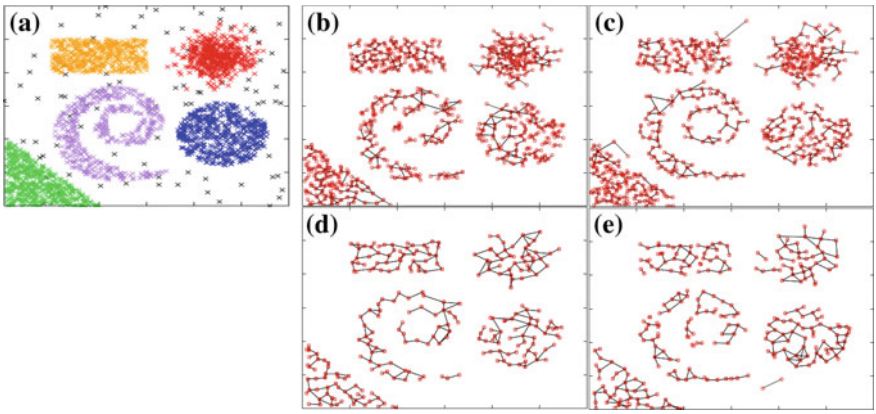


Fig. 4 The built topology of activated neurons, with and without the merging process **a** 2D dataset. **b**, **c** AING without merging. **d**, **e** AING with merging. **c**, **e** Random arrival order of data. **b**, **d** Class by class arrival order of data

4.2 Experiments on Real Datasets

We consider in our experimental evaluation, AING with and without the merging process,[2] some main incremental neural clustering algorithms, and an accurate incremental Kmeans [10] as a reference in comparing the results.

We consider a total of six datasets of different size and dimensions. Three standard public handwritten digit datasets (i.e. Pendigit and Optdigit from the UCI repository [11], and Mnist dataset [12]), and three different datasets of documents represented as bag of words, taken from a real administrative documents processing chain:

- Pendigit: 7,494 data for learning, 3,498 data for testing, 17 dimensions, 10 classes.
- Optdigit: 3,823 data for learning, 1,797 for testing, 65 dimensions, 10 classes.
- Mnist: 60,000 data for learning, 10,000 for testing, 784 dimensions, 10 classes.
- 1st documentary dataset: 1,554 data for learning, 777 for testing, 272 dimensions, 143 classes.
- 2nd documentary dataset. 2,630 data for learning, 1,315 for testing, 278 dimensions, 24 classes.
- 3rd documentary dataset. 3,564 data for learning, 1,780 for testing, 293 dimensions, 25 classes.

In addition to the number of produced representatives and the number of required parameters, we consider as evaluation measures the recognition rate (R) and the v-measure (V) [13]. Basically, v-measure is an entropy-based measure which expresses the compromise between homogeneity and completeness of the produced clusters and gives an idea about the ability to generalize to future data. Indeed, according to [13], it is important that clusters contain only data-points which are members of a single class (perfect homogeneity), but it is also important that all the data-points that are members of a given class are elements of the same cluster (perfect completeness).

For each algorithm, we repeat many experiments by slightly varying the parameter values needed by each of them. We finally keep the parameter values matching the best clustering results according to the considered evaluation measures.

The results obtained on the 3 first datasets are shown in Table 1, where AING1 (respectively AING2) refers to AING without (respectively with) the merging process. From Table 1, we see that concerning the 1st dataset, Kmeans achieves a better v-measure, and maintains fewer representatives, but does not reach a recognition rate which is comparable to the other algorithms. Although AING1 (without the merging process) is independent of external parameters, it realises almost the same recognition rate and v-measure as SOINN and I2GNG. AING2 (with the merging process) produces fewer neurons and the recognition rate as well as the v-measure are improved further. Concerning the 2nd dataset (Optdigit), AING1 realises the greatest performances. With AING2, the number of neurons is considerably reduced and a better compromise between homogeneity and completeness is achieved. The recognition rate is a little worse than the AING1, but still very close to the highest rate obtained by the other algorithms. Concerning the Mnist dataset, AING2 achieved the best performances.

[2] We will refer to AING without the merging process by AING1, and to AING with the merging process by AING2.

Table 1 Validation on public standard datasets (R = Recognition rate, V = V-Measure, Params = Number of parameters)

Method	Neurons	R (%)	V (%)	Params
Pendigit dataset				
AING1	1943	97.427	52.538	**0**
AING2	1403	**97.827**	53.624	1
Kmeans	**1172**	97.055	**54.907**	3
SOINN	1496	97.341	52.222	3
I2GNG	2215	97.541	52.445	4
Optdigit dataset				
AING1	1371	**97.718**	54.991	**0**
AING2	**825**	97.440	**55.852**	1
Kmeans	1396	97.495	52.899	3
SOINN	1182	96.82	53.152	3
I2GNG	1595	97.161	53.555	4
Mnist dataset				
AING1	3606	94.06	45.258	**0**
AING2	**2027**	**94.21**	**46.959**	1
Kmeans	2829	94.04	45.352	3
SOINN	2354	93.95	44.293	3
I2GNG	5525	94.10	43.391	4

Table 2 shows the results obtained on the documentary datasets. Roughly, we can make the same conclusions as with the previous datasets. AING1 performs well, although it does not require other pre-defined parameters. However, when using the merging process (AING2) on these datasets, the obtained results are of lower quality than those obtained with AING1. This is due to the fact that these documentary datasets are not very large and that the obtained neurons are not sufficient to represent well all the different classes. Indeed, the documentary datasets contains much more classes than the 3 first handwritten digits datasets. The merging process is thus more convenient when dealing with large datasets.

Figure 5 shows how the recognition rate changes with changing values of the upper bound parameter (up_bound) for some datasets. Due to the reason cited previously, for the documentary datasets, the results are better as the value of the parameter up_bound is higher (which implies more neurons). However, if we take as an example the Pendigit dataset, we can observe that for all values greater than or equal to 600 (i.e. most reasonable values that up_bound can take), the recognition rate is in [97, 98] (i.e. around the same value). Note that for two experiments with a fixed value of up_bound, the result may slightly be different since the merging process is probabilistic. Furthermore, the maximum number of neurons that can be generated for this example is 1943, thus, for values of up_bound in $[1943, +\infty[$, the merging process does not take place and AING2 performs exactly like AING1 (i.e. for AING on the Pendigit dataset $\forall$ up_bound $\in [1943, +\infty[$: R = 97.4271 %).

Table 2 Validation on datasets of administrative documents (R = Recognition rate, V = V-Measure, Params = Number of parameters)

Method	Neurons	R (%)	V (%)	Params
1st documentary dataset				
AING1	1030	**91.505**	**87.751**	**0**
AING2	1012	89.446	87.461	1
Kmeans	**1013**	90.862	86.565	3
SOINN	1045	88.545	87.375	3
I2GNG	1367	91.119	86.273	4
2nd documentary dataset				
AING1	1215	98.251	**57.173**	**0**
AING2	**1011**	97.490	**59.356**	1
Kmeans	1720	98.098	53.966	3
SOINN	1650	97.338	55.124	3
I2GNG	1846	**98.403**	54.782	4
3rd documentary dataset				
AING1	2279	**91.685**	60.922	**0**
AING2	**1897**	89.269	**62.367**	1
Kmeans	2027	91.179	60.192	3
SOINN	2437	88.707	61.048	3
I2GNG	2618	90.393	60.954	4

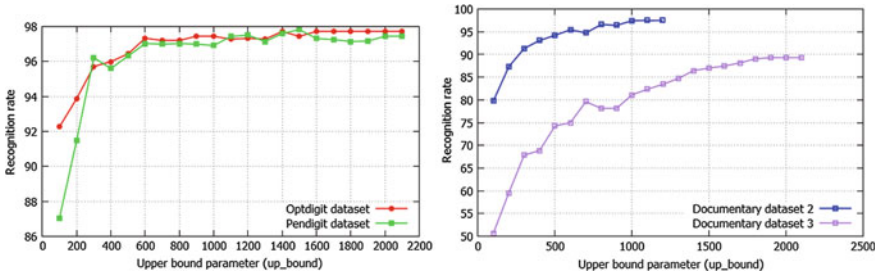


Fig. 5 The recognition rate achieved by AING according to the parameter up_bound for some datasets

Furthermore, the time required to incrementally integrate one data-point is strongly related to the current number of neurons (representatives) because the search for the nearest neurons from a new data-point is the most consuming operation. Figure 6 shows that AING is more convenient for a long-life learning task since it maintains a better processing time than the other algorithms over long periods of time learning, thanks to the merging process. The overall running time for the Mnist dataset (i.e. required for all the 60,000 data-points) is 1.83 h for AING, 2.57 h for SOINN and 4.49 h for I2GNG.

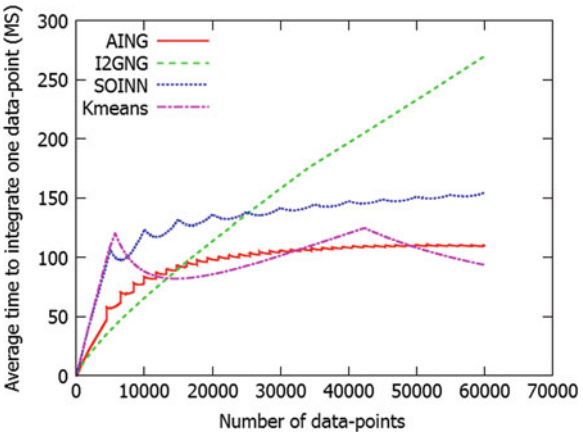


Fig. 6 The average time (in millisecond) required to incrementally integrate one data-point (for the Mnist dataset)

5 Conclusions and Future Work

This paper presents an online unsupervised learning method which incrementally processes data from the data stream, without being sensitive to initialization parameters. It initially decides whether a new data-point should produce a new cluster representative by means of a parameter-free adaptive threshold associated to each existing representative, and evolving dynamically according to the data and the topology of neurons. Some representatives may eventually be assigned to others by means of a distance-based probabilistic criterion each time their number exceed a specified limit; thus, maintaining a better clusters completeness, and preserving time and memory resources.

Nonetheless, further work still needs to be done. One of our directions for future work is to provide some theoretical worst-case bounds on memory and time requirement, and allow the algorithm to automatically determine an appropriate upper bound for the number of representatives; this will allow AING to perform a long-life learning. Then, we want to integrate the algorithm in a case-based reasoning system for document analysis, whose case-base will be continuously maintained by the AING algorithm.

Another direction is the extension of this work to semi-supervised and active learning. Indeed, AING is unsupervised and can not be directly applied to any classification task. The work in [14] extends AING in order to be suitable for a text document classification task, by allowing the algorithm to learn from both labelled and unlabelled documents and to actively query (from a human annotator) the class-labels of only documents that are most informative for learning, thus saving annotation time and effort.

References

1. Fritzke, B.: A growing neural gas network learns topologies. In: Neural Information Processing Systems, pp. 625–632 (1995)
2. Martinetz, T.: Competitive Hebbian learning rule forms perfectly topology preserving maps. In: Proceedings of the International Conference on Artificial Neural Networks, pp. 427–434. Amsterdam, Netherlands (1993)
3. Prudent, Y., Ennaji, A.: An incremental growing neural gas learns topologies. In: International Joint Conference on Neural Networks, pp. 1211–1216 (2005)
4. Hamza, H., Belaid, Y., Belaid, A., Chaudhuri, B.: Incremental classification of invoice documents. In: Proceedings of International Conference on Pattern Recognition, pp. 1–4 (2008)
5. Shen, F., Ogura, T., Hasegawa, O.: An enhanced self-organizing incremental neural network for online unsupervised learning. Neural Netw. **20**(8), 893–903 (2007)
6. Keogh, E., Lonardi, S., Ratanamahatana, C.A.: Towards parameter-free data mining. In: Proceedings of the 10th International Conference on Knowledge Discovery and Data Mining, pp. 206–215 (2004)
7. O'Callaghan, L., Meyerson, A., Motwani, R., Mishra, N., Guha, S.: Streaming-data algorithms for high-quality clustering. In: Proceedings of International Conference on Data Engineering, pp. 685–696. San Francisco (2002)
8. Aggarwal, C.C., Han, J., Wang, J., Yu, P.S.: A framework for clustering evolving data streams. In: Proceedings of the 29th International Conference on Very Large Data Bases, pp. 81–92. Berlin, Germany (2003)
9. Chen, Y., Tu, L.: Density-based clustering for real-time stream data. In: Proceedings of International Conference on Knowledge Discovery and Data Mining, pp. 133–142 (2007)
10. Shindler, M., Wong, A., Meyerson, A.: Fast and accurate k-means for large datasets. In: Neural Information Processing Systems, pp. 2375–2383 (2011)
11. Frank, A., Asuncion, A.: The UCI machine learning repository. http://archive.ics.uci.edu/ml/. (2010)
12. Yann, L., Corinna, C.: MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/. (2010)
13. Rosenberg, A., Hirschberg, J.: V-measure: a conditional entropy-based external cluster evaluation measure. In: Neural Information Processing Systems, pp. 410–420 (2007)
14. Bouguelia, M-R., Belaid, Y., Belaid, A.: A stream-based semi-supervised active learning approach for document classification. In: International Conference on Document Analysis and Recognition (2013)

The Path Kernel: A Novel Kernel for Sequential Data

Andrea Baisero, Florian T. Pokorny, Danica Kragic and Carl Henrik Ek

Abstract We define a novel kernel function for finite sequences of arbitrary length which we call the path kernel. We evaluate this kernel in a classification scenario using synthetic data sequences and show that our kernel can outperform state of the art sequential similarity measures. Furthermore, we find that, in our experiments, a clustering of data based on the path kernel results in much improved interpretability of such clusters compared to alternative approaches such as dynamic time warping or the global alignment kernel.

Keywords Kernels · Sequences

1 Introduction

Machine learning methods have had an enormous impact on a large range of fields such as computer vision, robotics and computational biology. These methods have allowed researchers to exploit evidence from data to learn models in a principled manner. One of the most important developments has been that of kernel methods

This work was supported by the EU project TOMSY (IST-FP7-270436) as well as the Swedish Foundation for Strategic Research.
An earlier version of this work was published under the title "The Path Kernel" at the International Conference on Pattern Recognition Applications and Methods, 2013.

A. Baisero (✉) · F.T. Pokorny · D. Kragic · C.H. Ek
Computer Vision and Active Perception Laboratory, Centre for Autonomous Systems,
KTH Royal Institute of Technology, Stockholm, Sweden
e-mail: baisero@kth.se

F.T. Pokorny
e-mail: fpokorny@kth.se

D. Kragic
e-mail: danik@kth.se

C.H. Ek
e-mail: chek@kth.se

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_5

[5] which embed the input data in a potentially high-dimensional vector space with the intention of achieving improved robustness of classification and regression techniques. The main benefit of kernel methods is that, rather than defining an explicit feature space that has the desired properties, the embedding is characterised implicitly through the choice of a kernel function which models the inner product in an induced space. This creates a very natural paradigm for recovering the desired characteristics of a representation. Kernel functions based on a stationary distances (usually an $\mathcal{L}_p - norm$) have been particularly successful in this context [3]. However, for many application domains, the data does not naturally lend itself to a finite dimensional vectorial representation. Symbolic sequences and graphs, for example, pose a problem for such kernels.

For non-vectorial data, the techniques used for learning and inference are generally much less developed. A desirable approach is hence to first place the data in a vector space where the whole range of powerful machine learning algorithms can be applied. Simple approaches such as the Bag-of-Words model, which creates a vectorial representation based on occurrence counts of specific representative "words", have had a big impact on computer vision [18]. These methods incorporate the fact that a distance in the observed space of image features does not necessarily reflect a similarity between the observed scenes. Another approach, where strings are transformed into a vectorial representation before a kernel method is applied, has been the development of string kernels [14, 17]. Such kernels open up a whole range of powerful techniques for non-vectorial data and they have been been applied successfully to robotics [15], computer vision [13] and biology [12]. Other related works are based on convolution kernels [11]. Using such kernels, a vectorial representation that respects the structure of a graph can be recovered. Another approach to define an inner product between sequences is to search for a space where similarity is reflected by "how well" sequences align [6, 7, 19].

In this paper, we present a new kernel for representing sequences of symbols which extends and further develops the concept of sequence alignment. Our kernel is based on a *ground space* which encodes similarities between the symbols in a sequence. We show that our kernel is a significant improvement compared to the state of the art both in terms of computational complexity and in terms of its ability to represent the data.

2 Kernels and Sequences

Before we proceed with describing previous work for creating kernel induced feature spaces for sequences, we will clarify our notation and our notion of kernels. When discussing kernels in the context of machine learning, we have to distinguish between several uses of the word kernel. In this paper, a kernel denotes any symmetric function $k : X \times X \to \mathbb{R}$, where X is a non-empty set [10]. A positive semi-definite (psd) kernel is a kernel $k : X \times X \to \mathbb{R}$ such that $\sum_{i,j}^{n} c_i c_j k(x_i, x_j) \geqslant 0$ for any $\{x_1, \ldots, x_n\} \subset X$, $n \in \mathbb{N}$ and $c_1, \ldots c_n \in \mathbb{R}$. If the previous inequality is strict

when $c_i \neq 0$ for at least one $i \in \{1, \ldots, n\}$, the kernel is called positive definite (pd). Further specialisations, such as negative definite (nd) kernels, exist and are of independent interest.

While there are strong theoretical results on the existence of embeddings corresponding to psd kernels [2, p. 22], non-psd kernel functions can still be useful in applications. Examples of kernels that are known to be neither pd nor psd but which are still successfully used in classification include [9]. On another note, there are also kernels which are conjectured to be psd, and which have been shown to be psd in experiments, but for which there currently is no proof for the corresponding positiveness [1].

In this work, we consider finite sequences of symbols belonging to an alphabet set Σ, i.e. $s = (s_1, s_2, \ldots, s_{|s|})$ denotes such a sequence, with $s_i \in \Sigma$, and where $|s| \in \mathbb{N}_0$ denotes the length of the sequence. We denote by $s_{a:b}$, with $1 \leqslant a < b \leqslant |s|$, the subsequence $s_{a:b} = (s_a, \ldots, s_b)$. When the indices a or b are omitted, they implicitly refer to 1 or $|s|$ respectively. The *inverse* of a sequence s is defined by $\mathrm{inv}(s)_i = s_{|s|-(i-1)}$.

In this work, we assume that we are given a psd kernel function $k_\Sigma : \Sigma \times \Sigma \to \mathbb{R}$ describing the similarity between elements of the alphabet Σ and will refer to k_Σ as the ground kernel. Given k_Σ, we can now define the *path matrix*.

Definition 1 (*Path Matrix*) Given two finite sequences s, t with elements in an alphabet set Σ and a kernel $k_\Sigma : \Sigma \times \Sigma \to \mathbb{R}$, we define the *path matrix* $G(s, t) \in \mathbb{R}^{|s| \times |t|}$ by $[G(s, t)]_{ij} = k_\Sigma(s_i, t_j)$.

We denote $\delta_{00} \stackrel{\text{def}}{=} (0, 0)$, $\delta_{10} \stackrel{\text{def}}{=} (1, 0)$, $\delta_{01} \stackrel{\text{def}}{=} (0, 1)$, $\delta_{11} \stackrel{\text{def}}{=} (1, 1)$ and $S \stackrel{\text{def}}{=} \{\delta_{10}, \delta_{01}, \delta_{11}\}$. S is called the set of admissible steps. A sequence of admissible steps starting from (1, 1) defines the notion of a path:

Definition 2 (*Path*) A path over a $m \times n$ path-matrix G is a map $\gamma : \{1, \ldots, |\gamma|\} \to \mathbb{N} \times \mathbb{N}$ such that

$$\gamma(1) = (1, 1), \tag{1}$$

$$\gamma(i + 1) = \gamma(i) + \delta_i, \quad \text{for } 1 \leqslant i < |\gamma|, \text{ with } \delta_i \in S, \tag{2}$$

$$\gamma(|\gamma|) = (m, n). \tag{3}$$

$|\gamma|$ and δ_i denote the path's *length* and ith *step* respectively. Furthermore, we adopt the notation $\gamma(i) = (\gamma_X(i), \gamma_Y(i))$. A path determines *stretches*, or alignments, on the input sequences according to $s_{\gamma_X} = (s_{\gamma_X(1)}, \ldots, s_{\gamma_X(|\gamma|)})$ and $t_{\gamma_Y} = (t_{\gamma_Y(1)}, \ldots, t_{\gamma_Y(|\gamma|)})$.

We denote the set of all paths on a $m \times n$ matrix as $\Gamma(m, n)$. Its cardinality is equal to the Delannoy number $Del(m, n)$.

2.1 Sequence Similarity Measures

A popular similarity measure between time-series is Dynamic Time Warping (DTW) [8, 16], which determines the distance between two sequences s and t as the minimal score obtained by all paths, i.e.

$$d_{\mathrm{DTW}}(s,t) = \min_{\gamma \in \Gamma} D_{s,t}(\gamma), \tag{4}$$

where $D_{s,t}$ represents the score of a path γ defined by

$$D_{s,t}(\gamma) = \sum_{i=1}^{|\gamma|} \varphi(s_{\gamma_X(i)}, t_{\gamma_Y(i)}), \tag{5}$$

where φ is some given similarity measure. However, DTW lacks a geometrical interpretation in the sense that it does not necessarily respect the triangle inequality [7]. Furthermore, this similarity measure is not likely to be robust as it only uses information from the minimal cost alignment.

Taking the above into consideration, Cuturi et al. suggest a kernel referred to as the *Global Alignment Kernel* [7]. Instead of considering the minimum over all paths, the Global Alignment Kernel combines all possible path scores. The kernel makes use of an exponentiated soft-minimum of all scores, generating a more robust result which reflects the contents of all possible paths:

$$k_{\mathrm{GA}}(s,t) = \sum_{\gamma \in \Gamma} e^{-D_{s,t}(\gamma)}. \tag{6}$$

By taking the ground kernel to be $k_\Sigma(\alpha,\beta) = e^{-\varphi(\alpha,\beta)}$, k_{GA} can be described using the path matrix as

$$k_{\mathrm{GA}}(s,t) = \sum_{\gamma \in \Gamma} \prod_{i=1}^{|\gamma|} G(s,t)_{\gamma(i)}. \tag{7}$$

The leading principle in this approach is hence a combination of kernels on the level of symbols over all paths along $G(s,t)$. Cuturi shows that incorporating all the elements of $G(s,t)$ into the final results can vastly improve classification compared to using only the minimal cost path. Furthermore, k_{GA} is proven to be psd under the condition that both k_Σ and $\frac{k_\Sigma}{1+k_\Sigma}$ are psd [7], giving foundation to its geometrical interpretation.

However, Cuturi's kernel makes use of products between ground kernel evaluations along a path. This implies that the score for a complete path will be very small if $\varphi(s_i, t_j)$ is sufficiently large, which leads to the problem of diagonally dominant kernel matrices [6, 8] from which the global alignment kernel suffers. The issue is particularly troubling when occurring at positions near the top-left or bottom-right

corners of the path matrix because it will affect many of the paths. Furthermore, paths contribute with equal weight to the value of the kernel. To reduce this effect, it is suggested in [7] to rescale the kernel values and use its logarithm instead. We argue that paths which travel closest to the main diagonal of the path matrix should be considered as more important than others, since they minimise the distortion imposed on the input sequences, i.e. s_{γ_X} and t_{γ_Y} are then most similar to s and t. To rectify this and to include a preference towards diagonal paths, a generalisation called the *Triangular Global Alignment Kernel* was developed, which considers only a subset of the paths [6]. This generalisation imposes a crude preference for paths which do not drift far away from the main diagonal.

In this paper, we develop a different approach by introducing a weighting of the paths in Γ based on the number of diagonal and off-diagonal steps taken. We manipulate the weights to encode a preference towards consecutive diagonal steps while at the same time accumulating information about all possible paths. Furthermore, by replacing the accumulation of symbol kernel responses along the path using a summation rather than a product, the kernel's evaluation reflects more gracefully the structure of the sequences and avoids abrupt changes in value.

3 The Path Kernel

In this section we will describe our proposed kernel which we will refer to as the *path kernel*. Figure 1 illustrates the contents of a path matrix in a simplified example, showing the emergence of diagonal patterns when the two sequences are in good correspondence.

Table 1 shows the resulting alignments associated with the paths shown in Fig. 1. We argue that the values, the length and the location of these diagonals positively reflect the relation between the inputs and should thus be considered in the formu-

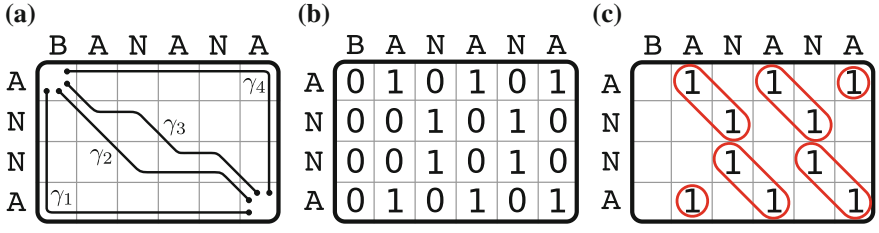


Fig. 1 Illustration of the concept of paths and the contents of $G(s, t)$ for s = "ANNA" and t = "BANANA". On the *left*, we illustrate a small number of paths which traverse G. The path kernel makes use of these, together with all the other paths, to collect data from the matrix and to extract a final score. In the *center*, we display the contents of $G(s, t)$, assuming $k_\Sigma(\alpha, \beta) = \delta_{\alpha\beta}$, i.e. Kroenecker's delta function. On the *right*, we highlight the corresponding diagonals whose number, length and position are related to the similarity between subsequences of s and t

Table 1 Stretches associated to the paths in Fig. 1a with the underlined substrings denoting a repeated symbol

Stretches			
γ_1	A N N A A A A A A	γ_2	A N N N N A
	B B B B A N A N A		B A N A N A
γ_3	A N N N N A	γ_4	A A A A A A N N A
	B A N A N A		B A N A N A A A A

Note that, even though γ_2 and γ_3 produce the same stretches, they traverse the matrix differently and should thus be considered separately

lation of a good kernel. High values imply a good match on the ground kernel level, while their length encodes the extent of the match. On the other hand, the position relative to the main diagonal reflects how much the input sequences had to be "stretched" in order for the match to be encountered. We wish to have a feature space where a smaller stretch implies a better correspondence between the sequences.

Let us now define a new kernel that incorporates different weightings depending on the steps used to travel along a path.

Definition 3 (*Path Kernel*) For any sequences s, t, we define

$$k_{\text{PATH}}(s,t) \overset{\text{def}}{=} \begin{cases} k_\Sigma(s_1,t_1) & \\ \quad + C_{\text{HV}}\, k_{\text{PATH}}(s_{2:},t) & \\ \quad + C_{\text{HV}}\, k_{\text{PATH}}(s,t_{2:}) & |\,s\,| \geqslant 1 \,|\, t\,| \geqslant 1, \\ \quad + C_{\text{D}}\, k_{\text{PATH}}(s_{2:},t_{2:}) & \\ 0 & \text{otherwise}, \end{cases} \tag{8}$$

where C_{HV} and C_{D} represent weights assigned to vertical or horizontal steps and diagonal steps respectively. By enforcing the constraints $C_{\text{HV}} > 0$ and $C_{\text{D}} > C_{\text{HV}}$, we aim to increase the relative importance of paths with many diagonal steps.

The symmetry of the kernel is easily verifiable. On the other hand, the positive semi-definiteness of the kernel is not immediately obvious from the definition.[1]

3.1 Efficient Computation

Kernel methods often require the computation of a kernel function on a large dataset, where the number of kernel evaluations will grow quadratically with the number of data-points. It is hence essential that the kernel evaluations themselves are efficiently computable.

Not only can the path kernel be evaluated using a Dynamic Programming algorithm which avoids the expensive recursion in (8) and which achieves a computa-

[1] In an extension of this work, which is currently under review, we provide a proof of the positive semi-definiteness of our kernel.

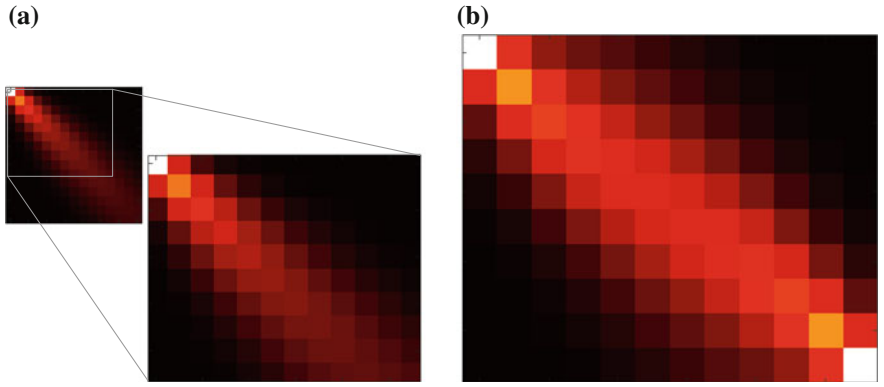


Fig. 2 On the *left*, a precomputed 15 × 15 weight matrix with $C_{HV} = 0.3$ and $C_D = 0.34$ is used to select a 10 × 12 weight matrix which can then be used to evaluate $k_{\mathrm{PATH}}(s,t)$ for input sizes $|s| = 10$ and $|t| = 12$. On the *right*, the inversion invariant $\tilde{\omega}_{\mathrm{PATH}}$ corresponding to $\tilde{k}_{\mathrm{PATH}}$ for the same input sizes is displayed

tional complexity comparable with DTW and k_{GA}, but it can also be computed very efficiently using the following alternative formulation:

$$k_{\mathrm{PATH}}(s,t) = \sum_{ij} G(s,t)_{ij}\ \omega_{\mathrm{PATH}ij}, \tag{9}$$

$$[\omega_{\mathrm{PATH}}]_{ij} = \sum_{d=0}^{\min(i,j)-1} C_{\mathrm{HV}}^{i+j-2-2d} C_{\mathrm{D}}^{d}(d, i-1-d, j-1-d)!. \tag{10}$$

The usefulness of (9) comes from the fact that the contents of the weight matrix ω_{PATH} do not really depend on s, t, and thus ω_{PATH} can in fact be pre-computed up to an adequate size[2] (Fig. 2). After this, the evaluation of the kernel for input of sizes m and n is achieved by simply selecting the sub-matrix ranging from $(1,1)$ to (m,n); the remaining matrix element-wise multiplication can then be efficiently carried out. By taking advantage of this, one can evaluate the kernel at speeds depending only on the speed of the evaluation of G and the speed of a simple matrix multiplication (with the initial overhead consisting of the pre-computation of ω_{PATH}). The weight matrix can also be computed through an efficient and very simple Dynamic Programming algorithm similar to the one which can be used to evaluate the kernel itself.

We call a kernel satisfying $k(s,t) = k(\mathrm{inv}(s), \mathrm{inv}(t))$ inversion invariant. If a kernel k does not naturally have this property, it can be enforced by replacing k with

$$\tilde{k}(s,t) = \frac{1}{2}\left[k(s,t) + k(\mathrm{inv}(s), \mathrm{inv}(t))\right]. \tag{11}$$

[2] For any specific dataset, that would be the length of the longest sequence.

The path kernel is not originally inversion invariant, but invariance can be enforced without the need for a double computation of the kernel for each evaluation. This is done by modifying the selected sub-matrix of ω_{PATH} as follows: for any two inputs with lengths m and n, we replace the weight matrix ω_{PATH} by

$$[\tilde{\omega}_{\text{PATH}}]_{ij} = \frac{1}{2}\left[\,\omega_{\text{PATH}ij} + \omega_{\text{PATH}m-i+1,n-j+1}\,\right], \tag{12}$$

and then proceed using this weight matrix.

3.2 Ground Kernel Choice

The path kernel is based on a ground kernel which, apart from being a psd kernel function, is not constrained in any other way. However, we show in this paragraph that an arbitrarily k_Σ may lead to undesirable results.

Assume an alphabet and a ground kernel such that $\alpha, \beta \in \Sigma$, $k_\Sigma(\alpha, \alpha) = 1$, $k_\Sigma(\beta, \beta) = 1$ and $k_\Sigma(\alpha, \beta) = -1$. Given the input sequences $s = (\alpha, \beta, \ldots, \alpha, \beta)$ and $t = (\beta, \alpha, \ldots, \beta, \alpha)$, one may be inclined to say that s and t are very similar because each can be obtained from the other by cyclically shifting the symbols by one position. However, the contents of $G(s, t)$ show a collection of ones and negative ones organised in a chessboard-like disposition. This obviously leads to heavy fluctuations during the computation of $k_{\text{PATH}}(s, t)$ and to potentially very small values. Furthermore, the issue is present even in the computation of $k_{\text{PATH}}(s, s)$ and $k_{\text{PATH}}(t, t)$ which is not desirable under any circumstance. This problem is however easily rectifiable by considering only ground kernels that yield non-negative results on elements of Σ.

4 Experiments

In this section, we present the results of experiments performed with the proposed kernel. In particular, we perform two separate quantitative experiments that, in addition to our qualitative results, shed some light on the behaviour of the proposed method in comparison to previous work. We will compare our approach to k_{GA} as well as the non-psd kernel obtained by using the negative exponential of the DTW distance,

$$k_{\text{DTW}}(s, t) = e^{-d_{\text{DTW}}(s,t)}. \tag{13}$$

In the first experiment, we generate eight different classes of uni-variate sequences. Each class consists of a periodic waveform namely $\pm$*sine*, $\pm$*cosine*, $\pm$*sawtooth* and $\pm$*square*. From these, we generate noisy versions by performing three different operations:

1. The length of the sequence is generated by sampling from a normal distribution $\mathcal{N}(100, \sigma_l^2)$, rounding the result and rejecting non-positive lengths.
2. We obtain an *input* sequence as $|\, s \,|$ equidistant numbers spanning 2 periods of the wave; we then add to each element an *input* noise which follows a normal distribution $\mathcal{N}(0, \sigma_i^2)$.
3. We feed the noisy input sequence to the generating waveform and get an *output* sequence, to which we add *output* noise which follows a normal distribution $\mathcal{N}(0, \sigma_o^2)$.

Figure 3 shows the sequences for the parameter setting $\sigma_{\{l,i,o\}} = 5$. This corresponds to the setting which we will use to present our main results.

The path kernel has two different sets of parameters: the ground kernel and the weights associated with steps in the path matrix. In our experiments, we use a simple zero mean Gaussian kernel with standard deviation 0.1 as ground kernel. The step weights C_{HV} and C_{D} are set to 0.3 and 0.34 respectively. We use the same setting throughout the experiments. The behaviour of the kernel will change with the value of these parameters. A complete analysis of this is however beyond the scope of this paper. Here, we focus on the general characteristics of our kernel which summarises all possible paths using step weights satisfying $C_{\mathrm{HV}} < C_{\mathrm{D}}$—implying a preference for diagonal paths.

In order to get an understanding of the geometric configuration of the data that our kernel matrices corresponds to, we project the data onto the two first principal directions as determined by each of these kernels. The result can be seen in Fig. 4. It is important to note that, as the DTW kernel has negative eigenvalues, it does not imply a geometrically valid configuration of datapoints in a feature space.

From Fig. 4, we get a qualitative understanding of how the induced feature spaces looks like. However, a representation is simply the means to an end and to be able to make a valuable assessment of its useability, we need to use it to achieve a task. We do so through two different experimental setups: The first is meant to test the discriminative capabilities of the representation; the second evaluates how well the representation is suited for generalisation.

In order to test the discriminability of the feature space generated by the path kernel, we perform a classification experiment using the same data as explained above, and where the task is to predict the generating class of a waveform. We feed the kernel matrix into an SVM classifier [4], use a 2-fold cross-validation, and report the average over 50 runs. Due to the negative eigenvalues, the classification fails for the DTW kernel. For this reason, we only present results for the remaining two kernels. In Fig. 5a, the results for the classification with increasing noise levels are shown. For moderate noise-levels (up to $\sigma_{\{l,i,o\}} = 5$), the global alignment and the path kernel are comparable in performance, while—at a higher noise level—the performance of the global alignment kernel rapidly deteriorates and at $\sigma_{\{l,i,o\}} = 9$ its performance is about chance, while the path kernel still achieves a classification rate of over 80 %.

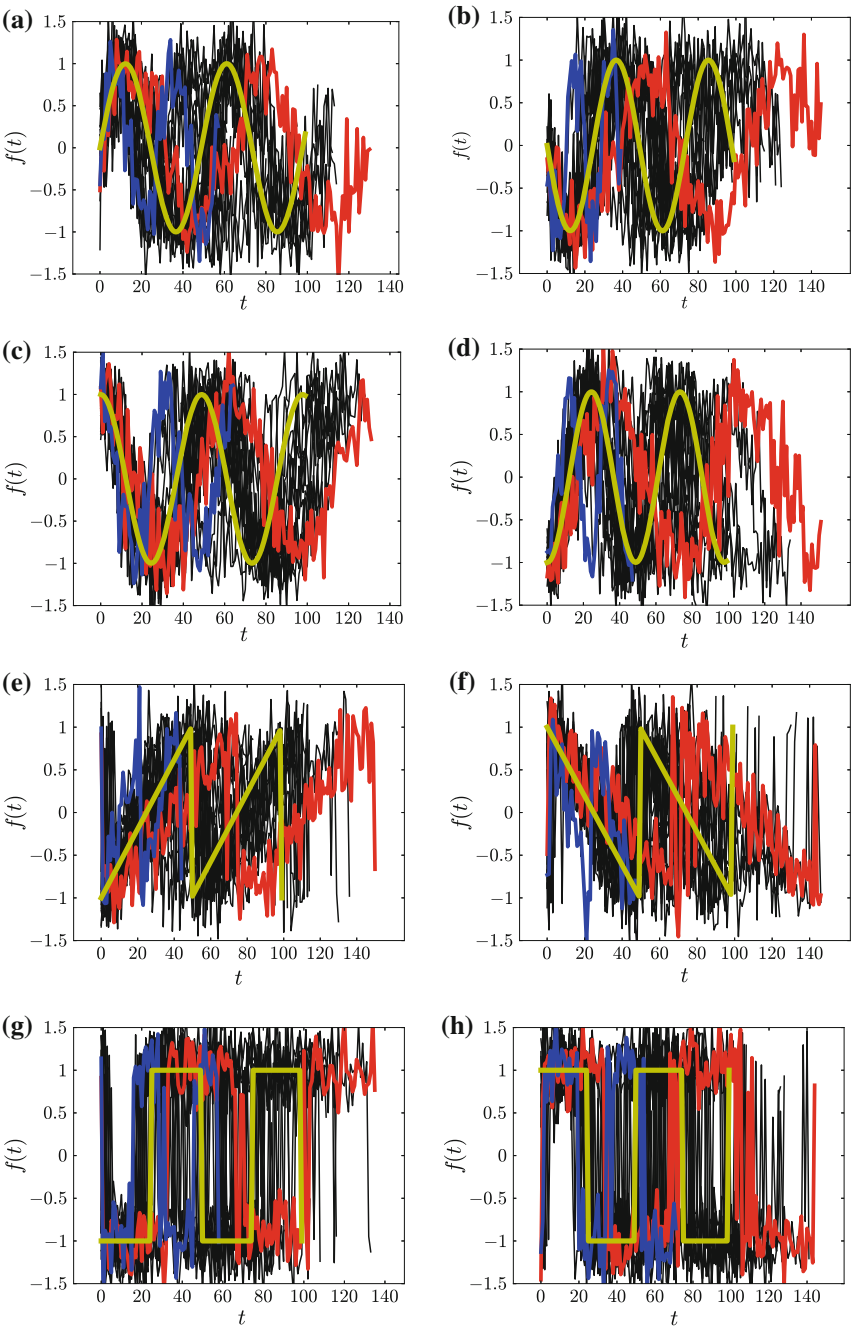


Fig. 3 The figure above shows the eight different waveforms used for the classification for a noise level corresponding to $\sigma_{\{l,i,o\}} = 5$. The *golden curve* depicts the base waveform without noise while the *blue* and *red curves* show the shortest and the longest noisy example respectively. The *black curves* display the remaining examples in the database

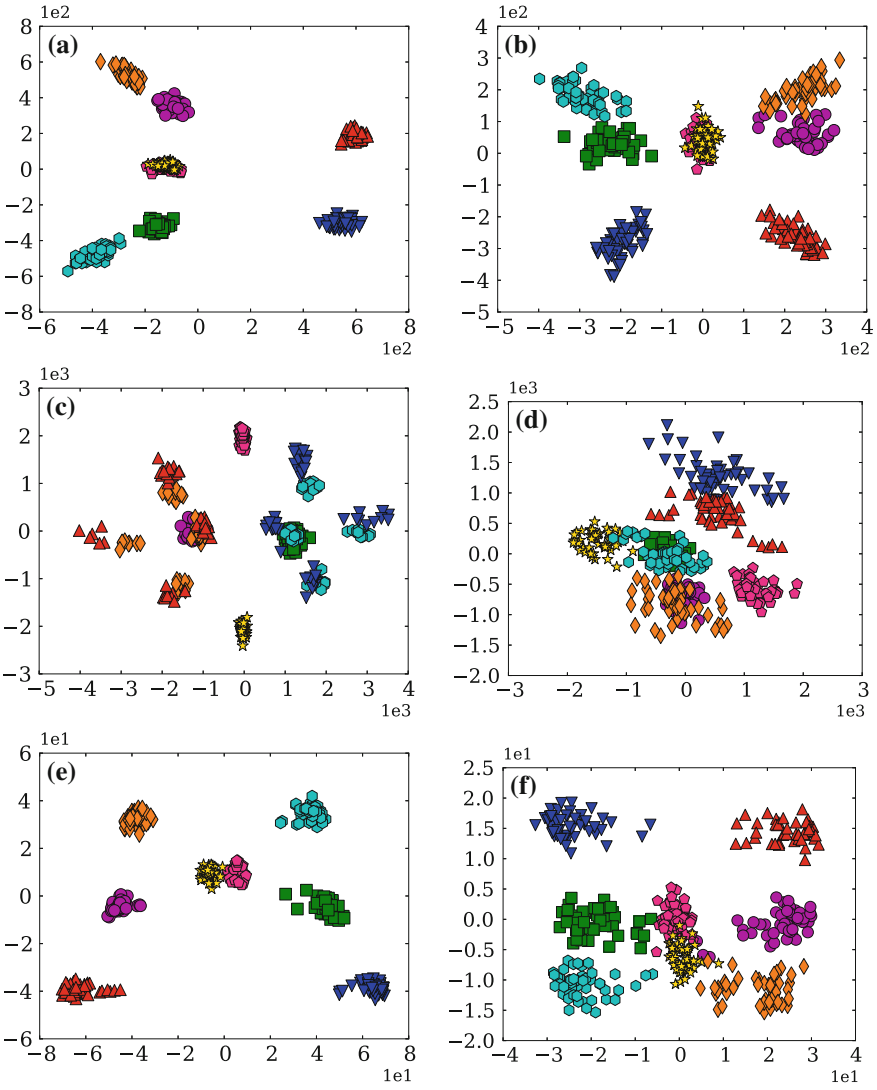


Fig. 4 The above figure displays the two dimensional principal subspace for the DTW Kernel (*top*), Global Alignment Kernel (*middle*) and the Path Kernel (*bottom*). The left column represents data with generation noise $\sigma_{\{l,i,o\}} = 2$, while, in the *right column*, the noise is increased to $\sigma_{\{l,i,o\}} = 5$, corresponding to the waveforms in Fig. 3. The different waveforms are displayed as follows: sine and-sine as a *magenta circle* and a *green square*, cosine and-cosine as a *pink pentagon* and a *yellow star*, sawtooth and-sawtooth as a *light-blue hexagon* and an *orange diamond* and *square* and-*square* as a *blue* and a *red triangle* respectively

The classification experiment shows that the path kernel significantly outperformed the global alignment kernel when noise in the sequence became significant. Looking at the feature space, depicted in Fig. 4, we see that the path kernel encodes a feature space having more clearly defined clusters corresponding to the different waveforms. Additionally, the clusters also have a simpler structure. This signifies that the path kernel should be better suited for generalisation purposes, where it is beneficial to have a large continuous region of support which gracefully describes the variations in the data—rather than working in a space that barely separates the classes.

We now generate a new dataset consisting of 100 noisy sine-waves ($\sigma_{\{l,i,o\}} = 5$) shifted in phase between 0 and π. The data is split uniformly into two halves and the first is used for training and the second for testing. We want to evaluate how well the kernel is capable of generalising over the training data. To that end we regress from the proportion of the training data to the test data and evaluate how the prediction error changes by altering this proportion. The prediction is performed using simple least-square regression over the kernel induced feature space. Figure 5b shows the results using different sizes of the training data; The path kernel performs significantly better compared to the global alignment kernel and the results improve with the size of the training dataset. Interestingly, the global alignment kernel produces very different results dependent on the size of the training dataset indicating that it is severely over-fitting the data.

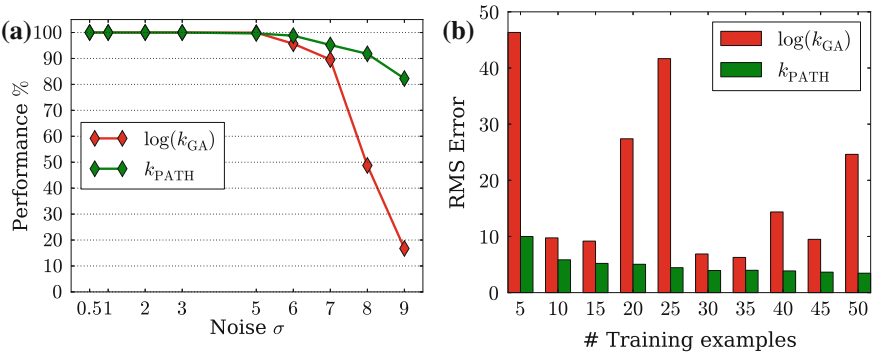


Fig. 5 The *left* figure displays the classification rate for predicting the waveform type using an SVM classifier in the feature space defined by the Global Alignment Kernel (*blue*) and the Path Kernel (*green*). The x-axis depicts the noise level parametrized by $\sigma_{\{l,i,o\}}$. The *right* figure depicts the RMS error when predicting the phase shift from a noisy sine waveform by a regression over the feature space induced by the kernels. The *red bars* correspond to the Global Alignment Kernel and the *green bars* to the Path Kernel. The y-axis shows the error in percentage of phase, while the *x-axis* indicates the size of the training dataset. The test set has a constant size of 50

5 Conclusions

In this paper, we have presented a novel kernel for encoding sequences. Our kernel reflects and encodes all possible alignments between two sequences by associating a cost to each. This cost encodes a preference towards specific paths. The kernel is applicable to any kind of symbolic or numerical data as it requires only the existence of a kernel between symbols. We have presented both qualitative and quantitative experiments exemplifying the benefits of the path kernel compared to competing methods. We show that the proposed method significantly improves results both with respect to discrimination and generalisation especially in noisy scenarios. The computational cost associated with the kernel is considerably lower than competing methods, making it applicable to data-sets that could previously not be investigated using kernels.

In this paper, we have chosen a very simple dataset in order to evaluate our kernel. Given our encouraging results, we are currently working on applying our kernel to more challenging real-world datasets. Additionally, we are investigating the possibility of optimizing the kernel parameters to further improve classification results.

References

1. Bahlmann, C., Haasdonk, B., Burkhardt, H.: Online handwriting recognition with support vector machines—a kernel approach. In: 8th International Workshop on Frontiers in Handwriting Recognition, pp. 49–54 (2002)
2. Berlinet, A., Thomas-Agnan, C.: Reproducing Kernel Hilbert Spaces in Probability and Statistics. Kluwer, Dordrecht (2004)
3. Buhmann, M.D., Martin, D.: Radial Basis Functions: Theory and Implementations. Cambridge University Press, Cambridge (2003)
4. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001)
5. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines: And Other Kernel-Based Learning Methods. Cambridge University Press, Cambridge (2006)
6. Cuturi, M.: Fast global alignment kernels. In: Proceedings of the International Conference on Machine Learning (2010)
7. Cuturi, M., Vert, J.P., Birkenes, O., Matsui, T.: A kernel for time series based on global alignments. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. II-413–II-416 (2007)
8. Gudmundsson, S., Runarsson, T.P., Sigurdsson, S.: Support vector machines and dynamic time warping for time series. In: Proceedings of the IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence) (x), pp. 2772–2776 (2008). http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4634188
9. Haasdonk, B.: Feature space interpretation of SVMs with indefinite kernels. IEEE Trans. Pattern Anal. Mach. Intell. **27**(4), 482–492 (2005)
10. Haasdonk, B., Burkhardt, H.: Invariant kernel functions for pattern analysis and machine learning. Mach. learn. **68**(1), 35–61 (2007)
11. Haussler, D.: Convolution kernels on discrete structures. Tech. rep. (1999)
12. Leslie, C., Kuang, R.: Fast string kernels using inexact matching for protein sequences. J. Mach. Learn. Res. **5**, 1435–1455 (2004)

13. Li, M., Zhu, Y.: Image classification via LZ78 based string kernel: a comparative study. Advances in Knowledge Discovery and Data Mining. Springer, Berlin (2006)
14. Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., Watkins, C.: Text classification using string kernels. J. Mach. Learn. Res. **2**, 419–444 (2002)
15. Luo, G., Bergström, N., Ek, C.H., Kragic, D.: Representing actions with Kernels. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2028–2035 (2011)
16. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. Acoust. Speech Signal Process. **26**(1), 43–49 (1978)
17. Saunders, C., Tschach, H., Shawe-Taylor, J.: Syllables and other string kernel extensions. In: Proceedings of the Nineteenth International Conference on Machine Learning (ICML'02) (2002)
18. Sivic, J., Zisserman, A.: Efficient visual search of videos cast as text retrieval. IEEE Trans. Pattern Anal. Mach. Intell. **31**(4), 591–606 (2009)
19. Watkins, C.: Dynamic alignment kernels. In: Proceedings of the Advances in Neural Information Processing Systems (1999)

A MAP Approach to Evidence Accumulation Clustering

André Lourenço, Samuel Rota Bulò, Nicola Rebagliati, Ana Fred, Mário Figueiredo and Marcello Pelillo

Abstract The Evidence Accumulation Clustering (EAC) paradigm is a clustering ensemble method which derives a consensus partition from a collection of base clusterings obtained using different algorithms. It collects from the partitions in the ensemble a set of pairwise observations about the co-occurrence of objects in a same cluster and it uses these co-occurrence statistics to derive a similarity matrix, referred to as co-association matrix. The Probabilistic Evidence Accumulation for Clustering Ensembles (PEACE) algorithm is a principled approach for the extraction of a consensus clustering from the observations encoded in the co-association matrix based on a probabilistic model for the co-association matrix parameterized by the unknown assignments of objects to clusters. In this paper we extend the PEACE algorithm by deriving a consensus solution according to a MAP approach with Dirichlet priors

An erratum to this chapter is available at DOI 10.1007/978-3-319-12610-4_20

A. Lourenço (✉)
Instituto Superior de Engenharia de Lisboa, Instituto de Telecomunicações, Lisbon, Portugal
e-mail: alourenco@deetc.isel.ipl.pt

A. Lourenço · M. Figueiredo
Instituto de Telecomunicações, Instituto Superior Técnico, Lisbon, Portugal
e-mail: afred@lx.it.pt

M. Figueiredo
e-mail: mtf@lx.it.pt

A. Fred
Instituto de Telecomunicações, Scientific Area of Networks and Multimedia, Lisbon, Portugal
e-mail: afred@lx.it.pt

S. Rota Bulò
Fondazione Bruno Kessler, Trento, Italy
e-mail: rotabulo@fbk.eu

M. Pelillo
DAIS, Università Ca' Foscari Venezia,Venice, Italy
e-mail: pelillo@dsi.unive.it

N. Rebagliati
VTT, Espoo, Finland
e-mail: nicola.rebagliati@gmail.com

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_6

defined for the unknown probabilistic cluster assignments. In particular, we study the positive regularization effect of Dirichlet priors on the final consensus solution with both synthetic and real benchmark data.

Keywords Clustering algorithm · Clustering ensembles · Probabilistic modeling · Evidence accumulation clustering · Prior knowledge

1 Introduction

The Evidence Accumulation Clustering algorithm (EAC) [1–3] was proposed in 2001 as one possible approach for the clustering combination problem. In this setting, a consensus clustering is determined from an ensemble of clusterings constructed from a set of base clustering algorithms. The leverage of an ensemble of clusterings is considerably more difficult than combining an ensemble of classifiers, due to the correspondence problem between the cluster labels produced by the different clustering algorithms. This problem becomes more serious if clusterings with different numbers of clusters are allowed in the ensemble.

In the EAC framework the clustering ensemble is summarized into a pair-wise *co-association matrix*, where each entry counts the number of clusterings in the ensemble in which a given pair of objects is placed in the same cluster, thus sidestepping the cluster label correspondence problem. This matrix, which is regarded to as a similarity matrix, is then used to feed a pairwise similarity clustering algorithm to deliver the final consensus clustering [3]. The drawback of this approach is that the information about the very nature of the co-association matrix is not properly exploited during the consensus clustering extraction.

A first work in the direction of finding a more principled way of using the information in the co-association matrix is [4]. There, the problem of extracting a consensus partition is formulated as a matrix factorization problem, under probability simplex constraints on each column of the factor matrix. Each of these columns can then be interpreted as the multinomial distribution that expresses the probabilities of each object being assigned to each cluster. The drawback of that approach is that the matrix factorization criterion is not supported on any probabilistic estimation rationale.

The Probabilistic Evidence Accumulation for Clustering Ensembles (PEACE) algorithm [5] is a probabilistic model for the co-association matrix, which regards its entries as independent observations of binomial random variables counting the number of times two objects occur in a same cluster. These random variables are indirectly parametrized by the unknown assignments of objects to clusters which are in turn estimated by adopting a Maximum-Likelihood Estimation (MLE) approach.

In this paper we extend PEACE by estimating the solution using a maximum a-posteriori (MAP) approach with Dirichlet prior for the unknown probabilistic assignments. With this Dirichlet prior the MAP estimate can be regarded as a regularized MLE estimate. Our formulation translates into a non-linear optimization problem, which is addressed by means of a primal line-search procedure that guarantees to find a local solution of the MAP estimation problem. In the experiments,

based on synthetic and real-world datasets from the UCI machine learning repository, we study the regularization and its positive effect on results.

The remainder of the paper is organized as follows. In Sect. 2, we describe our probabilistic model for the co-association matrix and the related MAP estimation of the unknown cluster assignments. Section 3 is devoted to solving the optimization problem arising for the unknown cluster assignments estimation. Section 4 contextualizes this model on related work. Finally, Sect. 5 reports experimental results and Sect. 6 presents some concluding remarks.

2 Probabilistic Model

Let $\mathcal{O} = \{1, \ldots, \mathsf{n}\}$ be the indices of a set of n objects to be clustered into k classes and let $\mathcal{E} = \{p_u\}_{u=1}^{\mathsf{m}}$ be a clustering ensemble, *i.e.*, a set of m clusterings (partitions) obtained by different algorithms (e.g. different parametrizations and/or initializations) on (possibly) sub-sampled versions of the object set. Each clustering $p_u \in \mathcal{E}$ is a function $p_u : \mathcal{O}_u \to \{1, ..., \mathsf{k}_u\}$, where $\mathcal{O}_u \subseteq \mathcal{O}$ is a sub-sample of $\mathcal{O}$ used as input to the uth clustering algorithm, and k_u is the corresponding number of clusters, which can be different on each $p_u \in \mathcal{E}$. Let $\Omega_{ij} \subseteq \{1, ..., \mathsf{m}\}$ denote the set of clustering indices where both objects i and j have been clustered, i.e. $(u \in \Omega_{ij}) \Leftrightarrow \big((i \in \mathcal{O}_u) \wedge (j \in \mathcal{O}_u)\big)$, and let $\mathtt{N}$ be a $\mathsf{n} \times \mathsf{n}$ matrix where $\mathtt{N}_{ij} = |\Omega_{ij}|$ for all $i, j \in \mathcal{O}$. The ensemble of clusterings is summarized in the co-association matrix $\mathtt{C} \in \{0, ..., \mathsf{m}\}^{\mathsf{n} \times \mathsf{n}}$. Each entry $\mathtt{C}_{ij}$ of this matrix with $i \neq j$ counts the number of times objects i and j are observed as clustered together in the ensemble $\mathcal{E}$, i.e.

$$\mathtt{C}_{ij} = \sum_{l \in \Omega_{ij}} \mathbb{1}_{p_l(i) = p_l(j)}$$

where $\mathbb{1}_P$ is an indicator function returning 1 or 0 according to whether the condition P given as argument is true or false. Of course, $0 \leq \mathtt{C}_{ij} \leq \mathtt{N}_{ij}$.

Our basic assumption is that each object has an (unknown) probability of being assigned to each cluster independently of other objects. We denote by $\boldsymbol{y}^i = (y^i_1, ..., y^i_{\mathsf{k}})^\top$ the probability distribution over the set of class labels $\{1, \ldots, \mathsf{k}\}$, that is $y^i_k = \mathbb{P}[i \in \mathcal{C}_k]$, where $\mathcal{C}_k$ denotes the subset of $\mathcal{O}$ that constitutes the kth cluster. Of course, $\boldsymbol{y}^i$ belongs to the probability simplex $\Delta = \{\boldsymbol{x} \in \mathbb{R}^{\mathsf{k}}_+ : \sum_{j=1}^{\mathsf{k}} x_j = 1\}$. Finally, we collect all the $\boldsymbol{y}^i$'s in a $\mathsf{k} \times \mathsf{n}$ matrix $\mathtt{Y} = [\boldsymbol{y}^1, \ldots, \boldsymbol{y}^{\mathsf{n}}] \in \Delta^{\mathsf{n}}$.

In our model, the probability that objects i and j are co-clustered is

$$\sum_{k=1}^{\mathsf{k}} \mathbb{P}[i \in \mathcal{C}_k, j \in \mathcal{C}_k] = \sum_{k=1}^{K} y^i_k \, y^j_k = \boldsymbol{y}^{i\top} \boldsymbol{y}^j$$

Let $\mathbb{C}_{ij}$ be a Binomial random variable representing the number of times that objects i and j are co-clustered; from the assumptions above, we have that $\mathbb{C}_{ij} \sim \text{Binomial}\big(\mathtt{N}_{ij},\ \boldsymbol{y}^{i\top}\boldsymbol{y}^{j}\big)$, that is,

$$\mathbb{P}\big[\mathbb{C}_{ij} = c | \boldsymbol{y}^i, \boldsymbol{y}^j\big] = \binom{\mathtt{N}_{ij}}{c}\big(\boldsymbol{y}^{i\top}\boldsymbol{y}^{j}\big)^{c}\big(1 - \boldsymbol{y}^{i\top}\boldsymbol{y}^{j}\big)^{\mathtt{N}_{ij}-c}.$$

Each element $\mathtt{C}_{ij}$ of the co-association matrix is interpreted as a sample of the random variable $\mathbb{C}_{ij}$, and the different $\mathbb{C}_{ij}$ are all assumed independent. Consequently,

$$\mathbb{P}[\mathtt{C}|\mathtt{Y}] = \prod_{\substack{i,j\in\mathcal{O}\\ i\neq j}} \binom{\mathtt{N}_{ij}}{\mathtt{C}_{ij}} (\boldsymbol{y}^{i\top}\boldsymbol{y}^{j})^{\mathtt{C}_{ij}} (1 - \boldsymbol{y}^{i\top}\boldsymbol{y}^{j})^{\mathtt{N}_{ij}-\mathtt{C}_{ij}}\ .$$

In [5], the MAP estimate of $\mathtt{Y}$ was taken, herein, we take a different approach the maximum a posterior estimate of $\mathtt{Y}$. The posterior probability of $\mathtt{Y}$ given the evidence $\mathtt{C}$ is defined as

$$\mathbb{P}[\mathtt{Y}|\mathtt{C}] \propto \mathbb{P}[\mathtt{C}|\mathtt{Y}]\mathbb{P}[\mathtt{Y}|\Theta]$$

where $\mathbb{P}[\mathtt{Y}|\Theta]$ is the prior distribution on the probabilistic cluster assignments. Let $\Theta = [\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^{\mathsf{n}}] \in \mathbb{R}_{+}^{\mathsf{k}\times\mathsf{n}}$. In our setting, we assume each assignment $\boldsymbol{y}^i$ to be an independent realization of a Dirichlet prior distribution with parameter $\boldsymbol{\theta}^i = (\theta^i_1, \ldots, \theta^i_{\mathsf{k}})^\top \in \mathbb{R}^{\mathsf{k}}$. Therefore, we have that

$$\mathbb{P}[\mathtt{Y}|\Theta] = \prod_{i\in\mathcal{O}} \mathbb{P}[\boldsymbol{y}^i|\boldsymbol{\theta}^i] = \prod_{i\in\mathcal{O}} B(\boldsymbol{\theta}^i)^{-1} \prod_{k=1}^{\mathsf{k}} (y^i_k)^{\theta^i_k - 1}\ ,$$

where B is the multinomial Beta function.

We compute the maximum a-posteriori estimate of $\mathtt{Y}$ by maximizing $\log \mathbb{P}[\mathtt{Y}|\mathtt{C}]$. By simple algebraic manipulations, this yields the following optimization problem:

$$\mathtt{Y}^* \in \arg\max_{\mathtt{Y}\in\Delta^{\mathsf{n}}} f(\mathtt{Y}) \tag{1}$$

where

$$\begin{aligned} f(\mathtt{Y}) = &\sum_{\substack{i,j\in\mathcal{O}\\ i\neq j}} \mathtt{C}_{ij} \log\Big(\boldsymbol{y}^{i\top}\boldsymbol{y}^{j}\Big) + (\mathtt{N}_{ij} - \mathtt{C}_{ij}) \log\Big(1 - \boldsymbol{y}^{i\top}\boldsymbol{y}^{j}\Big) \\ &+ \sum_{j\in\mathcal{O}} \sum_{k=1}^{\mathsf{k}} (\theta^j_k - 1) \log(y^j_k) + \text{constant}. \end{aligned} \tag{2}$$

Hereafter, we use $\log 0 \equiv -\infty$, $0 \log 0 \equiv 0$, and denote by $\mathbf{dom}(f) = \{\mathtt{Y} : f(\mathtt{Y}) \neq -\infty\}$ the domain of f.

In this paper we focus on a uniform regularization by assuming that $\Theta_{kj} = \lambda$ for all $k \in \{1, \ldots, \mathsf{k}\}$ and $j \in \mathcal{O}$, where $\lambda > 1$ is the regularization parameter. When $\lambda = 1$ the MAP estimate coincides with a MLE estimate. As λ tends to infinity, the MAP solution $\mathtt{Y}^*$ tends to a constant matrix.

3 Optimization Algorithm

The optimization method described in this section belongs to the class of primal line-search procedures. This method iteratively finds a direction which is *feasible*, i.e. satisfying the constraints, and *ascending*, i.e. guaranteeing a (local) increase of the objective function, along which a better solution is sought. The procedure is iterated until it converges or a maximum number of iterations is reached.

The first part of this section describes the procedure to determine the search direction in the optimization algorithm. The second part is devoted to determining an optimal step size to be taken in the direction found.

3.1 Computation of a Search Direction

Consider the Lagrangian of (1):

$$\mathcal{L}(\mathtt{Y}, \boldsymbol{\lambda}, \mathtt{M}) = f(\mathtt{Y}) + \mathrm{Tr}\Big[\mathtt{M}^\top \mathtt{Y}\Big] - \boldsymbol{\lambda}^\top (\mathtt{Y}^\top \boldsymbol{e}_{\mathsf{k}} - \boldsymbol{e}_{\mathsf{n}})$$

where $\mathrm{Tr}[\cdot]$ is the matrix trace operator, $\boldsymbol{e}_k$ is a k-dimensional column vector of all 1s, $\mathtt{Y} \in \mathbf{dom}(f)$ and $\mathtt{M} = [\boldsymbol{\mu}^1, \ldots, \boldsymbol{\mu}^{\mathsf{n}}] \in \mathbb{R}_+^{\mathsf{k} \times \mathsf{n}}$, $\boldsymbol{\lambda} \in \mathbb{R}^n$ are the Lagrangian multipliers. By derivating $\mathcal{L}$ with respect to $\boldsymbol{y}^i$ and $\boldsymbol{\lambda}$ and considering the complementary slackness conditions, we obtain the first order Karush-Kuhn-Tucker (KKT) conditions [6] for local optimality:

$$\begin{cases} g_i(\mathtt{Y}) - \lambda_i \boldsymbol{e}_{\mathsf{n}} + \boldsymbol{\mu}^i = \mathbf{0}, & \forall i \in \mathcal{O} \\ \mathtt{Y}^\top \boldsymbol{e}_{\mathsf{k}} - \boldsymbol{e}_{\mathsf{n}} = \mathbf{0} \\ \mathrm{Tr}\big[\mathtt{M}^\top \mathtt{Y}\big] = 0, \end{cases} \tag{3}$$

where

$$g_i(\mathtt{Y}) = \left[\sum_{j \in \mathcal{O} \setminus \{i\}} \mathtt{C}_{ij} \frac{\boldsymbol{y}^j}{\boldsymbol{y}^{i\top} \boldsymbol{y}^j} - (\mathtt{N}_{ij} - \mathtt{C}_{ij}) \frac{\boldsymbol{y}^j}{1 - \boldsymbol{y}^{i\top} \boldsymbol{y}^j} \right] + \mathrm{diag}\Big[\boldsymbol{y}^i\Big]^{-1} (\boldsymbol{\theta}^i - \boldsymbol{e}_{\mathsf{k}}).$$

Here, $\mathrm{diag}[\boldsymbol{v}]^{-1}$ is the inverse of the diagonal matrix having $\boldsymbol{v}$ on the diagonal. We can express the Lagrange multipliers $\boldsymbol{\lambda}$ in terms of $\mathtt{Y}$ by noting that

$$\boldsymbol{y}^{i\top}\left[g_i(\mathtt{Y}) - \lambda_i \boldsymbol{e}_{\mathsf{n}} + \boldsymbol{\mu}^i\right] = 0\,,$$

yields $\lambda_i = \boldsymbol{y}^{i\top} g_i(\mathtt{Y})$ for all $i \in \mathcal{O}$.

Let $r_i(\mathtt{Y})$ be given as

$$r_i(\mathtt{Y}) = g_i(\mathtt{Y}) - \lambda_i \boldsymbol{e}_{\mathsf{k}} = g_i(\mathtt{Y}) - \boldsymbol{y}^{i\top} g_i(\mathtt{Y}) \boldsymbol{e}_{\mathsf{k}},$$

and let $\sigma(\boldsymbol{y}^i)$ denote the support of $\boldsymbol{y}^i$, i.e. the set of indices corresponding to (strictly) positive entries of $\boldsymbol{y}^i$. An alternative characterization of the KKT conditions, where the Lagrange multipliers do not appear, is

$$\begin{cases} [r_i(\mathtt{Y})]_k = 0, & \forall i \in \mathcal{O}, \forall k \in \sigma(\boldsymbol{y}^i), \\ [r_i(\mathtt{Y})]_k \leq 0, & \forall i \in \mathcal{O}, \forall k \notin \sigma(\boldsymbol{y}^i), \\ \mathtt{Y}^\top \boldsymbol{e}_{\mathsf{k}} - \boldsymbol{e}_{\mathsf{n}} = \boldsymbol{0}. \end{cases} \tag{4}$$

The two characterizations (4) and (3) are equivalent. This can be verified by exploiting the non negativity of both matrices $\mathtt{M}$ and $\mathtt{Y}$, and the complementary slackness conditions.

The following proposition plays an important role in the selection of the search direction.

Proposition 1 *Assume* $\mathtt{Y} \in \mathbf{dom}(f)$ *to be feasible for* (1)*, i.e.* $\mathtt{Y} \in \Delta^{\mathsf{n}} \cap \mathbf{dom}(f)$. *Consider*

$$J \in \arg\max_{i \in \mathcal{O}} \left\{ [g_i(\mathtt{Y})]_{U_i} - [g_i(\mathtt{Y})]_{V_i} \right\},$$

where

$$U_i \in \arg\max_{k \in \{1 \ldots \mathsf{k}\}} [g_i(\mathtt{Y})]_k \quad \textit{and}$$
$$V_i \in \arg\min_{k \in \sigma(\boldsymbol{y}^j)} [g_i(\mathtt{Y})]_k\,.$$

Let $U = U_J$ *and* $V = V_J$. *Then the following holds:*

- $[g_J(\mathtt{Y})]_U \geq [g_J(\mathtt{Y})]_V$ *and*
- $\mathtt{Y}$ *satisfies the KKT conditions for* (1) *if and only if* $[g_J(\mathtt{Y})]_U = [g_J(\mathtt{Y})]_V$.

Proof We prove the first point by simple derivations as follows:

$$\begin{aligned} [g_J(\mathtt{Y})]_U \geq \boldsymbol{y}^{J\top} g_J(\mathtt{Y}) &= \sum_{k \in \sigma(\boldsymbol{y}^J)} y_k^J [g_J(\mathtt{Y})]_k \\ &\geq \sum_{k \in \sigma(\boldsymbol{y}^J)} y_k^J [g_J(\mathtt{Y})]_V = [g_J(\mathtt{Y})]_V. \end{aligned}$$

By subtracting $\boldsymbol{y}^{J\top} g_J(\mathtt{Y})$ we obtain the equivalent relation

$$[r_J(\mathtt{Y})]_U \geq 0 \geq [r_J(\mathtt{Y})]_V, \tag{5}$$

where equality holds if and only if $[g_J(\mathtt{Y})]_V = [g_J(\mathtt{Y})]_U$.

As for the second point, assume that $\mathtt{Y}$ satisfies the KKT conditions. Then $[r_J(\mathtt{Y})]_V = 0$ because $V \in \sigma(\boldsymbol{y}^J)$. It follows by (5) and (4) that also $[r_J(\mathtt{Y})]_U = 0$ and therefore $[g_J(\mathtt{Y})]_V = [g_J(\mathtt{Y})]_U$. On the other hand, if we assume that $[g_J(\mathtt{Y})]_V = [g_J(\mathtt{Y})]_U$ then by (5) and by definition of J we have that $[r_i(\mathtt{Y})]_{U_i} = [r_i(\mathtt{Y})]_{V_i} = 0$ for all $i \in \mathcal{O}$. By exploiting the definition of U_i and V_i it is straightforward to verify that $\mathtt{Y}$ satisfies the KKT conditions.

Given $\mathtt{Y}$ a non-optimal feasible solution of (1), we can determine the indices U, V and J as stated in Proposition 1. The next proposition shows how to build a feasible and ascending search direction by using these indices. Later on, we will point out some desired properties of this search direction. We denote by $\boldsymbol{e}_{\mathsf{n}}^{(j)}$ the jth column of the n-dimensional identity matrix.

Proposition 2 *Let $Y \in \Delta^{\mathsf{n}} \cap \mathbf{dom}(f)$ and assume that the KKT conditions do not hold. Let $D = \left(\boldsymbol{e}_{\mathsf{k}}^{(U)} - \boldsymbol{e}_{\mathsf{k}}^{(V)}\right)\left(\boldsymbol{e}_{\mathsf{n}}^{(J)}\right)^\top$, where J, U and V are computed as in Proposition 1. Then, for all $0 \leq \epsilon \leq y_V^J$, we have that $Z_\epsilon = Y + \epsilon D$ belongs to Δ^{n}, and for all small enough, positive values of ϵ, we have $f(Z_\epsilon) > f(Y)$.*

Proof Let $\mathtt{Z}_\epsilon = \mathtt{Y} + \epsilon\,\mathtt{D}$. Then for any ϵ,

$$\begin{aligned}\mathtt{Z}_\epsilon{}^\top \boldsymbol{e}_{\mathsf{k}} = (\mathtt{Y} + \epsilon\,\mathtt{D})^\top \boldsymbol{e}_{\mathsf{k}} &= \mathtt{Y}^\top \boldsymbol{e}_{\mathsf{k}} + \epsilon\,\mathtt{D}^\top \boldsymbol{e}_{\mathsf{k}} \\ &= \boldsymbol{e}_{\mathsf{n}} + \epsilon\,\boldsymbol{e}_{\mathsf{n}}^{(J)}\left(\boldsymbol{e}_{\mathsf{k}}^{(U)} - \boldsymbol{e}_{\mathsf{k}}^{(V)}\right)^\top \boldsymbol{e}_{\mathsf{k}} = \boldsymbol{e}_{\mathsf{n}}.\end{aligned}$$

As ϵ increases, only the (V, J)th entry of $\mathtt{Z}_\epsilon$, which is given by $y_V^J - \epsilon$, decreases. This entry is non-negative for all values of ϵ satisfying $\epsilon \leq y_V^J$. Hence, $\mathtt{Z}_\epsilon \in \Delta^{\mathsf{n}}$ for all positive values of ϵ not exceeding y_V^J as required.

As for the second point, the Taylor expansion of f at $\mathtt{Y}$ gives, for all small enough positive values of ϵ:

$$\begin{aligned}f(\mathtt{Z}_\epsilon) - f(\mathtt{Y}) &= \epsilon\left[\lim_{\epsilon \to 0} \frac{d}{d\epsilon} f(\mathtt{Z}_\epsilon)\right] + O(\epsilon^2) \\ &= \left(\boldsymbol{e}_{\mathsf{k}}^{(U)} - \boldsymbol{e}_{\mathsf{k}}^{(V)}\right)^\top g_J(\mathtt{Y}) + O(\epsilon^2) > 0 \\ &= [g_J(\mathtt{Y})]_U - [g_J(\mathtt{Y})]_V + O(\epsilon^2) > 0\end{aligned}$$

The last inequality derives from Proposition 1 because if $\mathtt{Y}$ does not satisfy the KKT conditions then $[g_J(\mathtt{Y})]_U - [g_J(\mathtt{Y})]_V > 0$.

3.2 Computation of an Optimal Step Size

Proposition 2 provides a direction $\mathtt{D}$ that is both feasible and ascending for $\mathtt{Y}$ with respect to (1). We will now address the problem of determining an optimal step ϵ^* to be taken along the direction $\mathtt{D}$. This optimal step is given by the following one dimensional optimization problem:

$$\epsilon^* \in \arg\max_{0 \leq \epsilon \leq y_V^J} f(\mathtt{Z}_\epsilon)\,, \tag{6}$$

where $\mathtt{Z}_\epsilon = \mathtt{Y} + \epsilon\mathtt{D}$. We prove this problem to be concave.

Proposition 3 *The optimization problem in (6) is concave, provided that $\theta_k^j \geq 1$ for all $j \in \mathcal{O}$ and $k \in \{1, \ldots, \mathsf{k}\}$.*

Proof The direction $\mathtt{D}$ is everywhere null except in the Jth column. Since the first sum in (2) is taken over all pairs (i, j) such that $i \neq j$ we have that the argument of every log function (which is a concave function) is linear in ϵ. The same holds true for the rest of the function since each coefficient of the log function is nonnegative. Concavity is preserved by the composition of concave functions with linear ones and by the sum of concave functions [7]. Hence, the maximization problem is concave.

Let $\rho(\epsilon')$ denote the first order derivative of f with respect to ϵ evaluated at ϵ', i.e.

$$\rho(\epsilon') = \lim_{\epsilon \to \epsilon'} \frac{d}{d\epsilon} f(\mathtt{Z}_\epsilon) = \left(\boldsymbol{e}_{\mathsf{k}}^{(U)} - \boldsymbol{e}_{\mathsf{k}}^{(V)}\right)^\top g_J(\mathtt{Z}_{\epsilon'}).$$

By the concavity of (6) and Kachurovskii's theorem [8] we derive that ρ is non-increasing in the interval $0 \leq \epsilon \leq y_V^J$. Moreover, $\rho(0) > 0$ since $\mathtt{D}$ is an ascending direction as stated by Proposition 2. In order to compute the optimal step ϵ^* in (6) we distinguish 2 cases:

- if $\rho(y_V^J) \geq 0$ then $\epsilon^* = y_V^J$ for $f(\mathtt{Z}_\epsilon)$ is non-decreasing in the feasible set of (6);
- if $\rho(y_V^J) < 0$ then ϵ^* is a zero of ρ that can be found by dichotomic search.

Suppose the second case holds, i.e. assume $\rho(y_V^J) < 0$. Then ϵ^* can be found by iteratively updating the search interval as follows:

$$\begin{aligned}
\left(\ell^{(0)}, r^{(0)}\right) &= \left(0, y_V^J\right) \\
\left(\ell^{(t+1)}, r^{(t+1)}\right) &= \begin{cases} \left(\ell^{(t)}, m^{(t)}\right) & \text{if } \rho\left(m^{(t)}\right) < 0\,, \\ \left(m^{(t)}, r^{(t)}\right) & \text{if } \rho\left(m^{(t)}\right) > 0 \\ \left(m^{(t)}, m^{(t)}\right) & \text{if } \rho\left(m^{(t)}\right) = 0\,, \end{cases}
\end{aligned} \tag{7}$$

for all $t > 0$, where $m^{(t)}$ denotes the center of segment $[\ell^{(t)}, r^{(t)}]$, i.e. $m^{(t)} = (\ell^{(t)} + r^{(t)})/2$.

We are not in general interested in determining a precise step size ϵ^* but an approximation is sufficient. Hence, the dichotomic search is carried out until the interval size is below a given threshold. If δ is this threshold, the number of iterations required is expected to be $\log_2(y_V^J/\delta)$ in the worst case.

3.3 Complexity

Consider a generic iteration t of our algorithm and assume $A^{(t)} = \mathtt{Y}^\top \mathtt{Y}$ and $g_i^{(t)} = g_i(\mathtt{Y})$ given for all $i \in \mathcal{O}$, where $\mathtt{Y} = \mathtt{Y}^{(t)}$. The computation of ϵ^* requires the evaluation of function ρ at different values of ϵ. Each function evaluation can be carried out in $O(\mathsf{n})$ steps by exploiting $\mathtt{A}^{(t)}$ as follows:

$$\rho(\epsilon) = \left[\sum_{i \in \mathcal{O} \setminus \{J\}} \mathtt{C}_{Ji} \frac{\boldsymbol{d}^\top \boldsymbol{y}^i}{\mathtt{A}_{Ji}^{(t)} + \epsilon\, \boldsymbol{d}^\top \boldsymbol{y}_i} + (\mathtt{N}_{Ji} - \mathtt{C}_{Ji}) \frac{\boldsymbol{d}^\top \boldsymbol{y}_i}{1 - \mathtt{A}_{Ji}^{(t)} - \epsilon\, \boldsymbol{d}^\top \boldsymbol{y}^i} \right] + \boldsymbol{d}^\top \mathrm{diag}\Big[\boldsymbol{y}^J + \epsilon\, \boldsymbol{d} \Big]^{-1} (\boldsymbol{\theta}^J - \boldsymbol{e}_{\mathsf{k}}) \,,$$

where $\boldsymbol{d} = \left(\boldsymbol{e}_{\mathsf{k}}^{(U)} - \boldsymbol{e}_{\mathsf{k}}^{(V)} \right)$. The complexity of the computation of the optimal step size is thus $O(\mathsf{n}\gamma)$ where γ is the average number of iterations needed by the dichotomic search. Next, we can efficiently update $\mathtt{A}^{(t)}$ as follows:

$$\mathtt{A}^{(t+1)} = \left(\mathtt{Y}^{(t+1)} \right)^\top \mathtt{Y}^{(t+1)} = \mathtt{A}^{(t)} + \epsilon^* \left(\mathtt{D}^\top \mathtt{Y} + \mathtt{Y}^\top \mathtt{D} + \epsilon^* \mathtt{D}^\top \mathtt{D} \right) . \tag{8}$$

Indeed, since $\mathtt{D}$ has only two non-zero entries, namely (V, J) and (U, J), the terms within parenthesis can be computed in $O(\mathsf{n})$. The computation of $\mathtt{Y}^{(t+1)}$ can be performed in constant time by exploiting the sparsity of $\mathtt{D}$ as $\mathtt{Y}^{(t+1)} = \mathtt{Y}^{(t)} + \epsilon^* \mathtt{D}$. The computation of $g_i^{(t+1)} = g_i(\mathtt{Y}^{(t+1)})$ for each $i \in \mathcal{O} \setminus \{J\}$ can be efficiently accomplished in $O(\mathsf{k})$ (it requires $O(\mathsf{nk})$ to update all of them) as follows:

$$g_i^{(t+1)} = g_i^{(t)} + \mathtt{C}_{iJ} \left(\frac{(\boldsymbol{y}^J)^{(t+1)}}{\mathtt{A}_{iJ}^{(t+1)}} - \frac{(\boldsymbol{y}^J)^{(t)}}{\mathtt{A}_{iJ}^{(t)}} \right) + (\mathtt{N}_{iJ} - \mathtt{C}_{iJ}) \left(\frac{(\boldsymbol{y}^J)^{(t+1)}}{1 - \mathtt{A}_{iJ}^{(t+1)}} - \frac{(\boldsymbol{y}^J)^{(t)}}{1 - \mathtt{A}_{iJ}^{(t)}} \right) . \tag{9}$$

The complexity of the computation of $g_J^{(t+1)}$, instead, requires $O(\mathsf{nk})$ steps:

$$g_J^{(t+1)} = \left[\sum_{i \in \mathcal{O} \setminus \{J\}} \mathtt{C}_{Ji} \frac{(\boldsymbol{y}^i)^{(t+1)}}{\mathtt{A}_{Ji}^{(t+1)}} - (\mathtt{N}_{Ji} - \mathtt{C}_{Ji}) \frac{(\boldsymbol{y}^i)^{(t+1)}}{1 - \mathtt{A}_{Ji}^{(t+1)}} \right] + \operatorname{diag}\left[(\boldsymbol{y}^J)^{(t+1)} \right]^{-1} (\boldsymbol{\theta}^J - \boldsymbol{e}_{\mathsf{k}}). \tag{10}$$

By iteratively updating the quantities $\mathtt{A}^{(t)}$, $g_i^{(t)}$'s and $\mathtt{Y}^{(t)}$ according to the aforementioned procedures, we can keep a per-iteration complexity of $O(\mathsf{nk})$, that is linear in the number of variables in $\mathtt{Y}$. Iterations stop when KKT conditions of proposition (1) are satisfied under a given tolerance τ, i.e. $[g_J(\mathtt{Y})]_U - [g_J(\mathtt{Y})]_V < \tau$.

Algorithm 1: PEACE-MAP.

Require: $\mathtt{Y}^{(0)} \in \Delta^{\mathsf{n}} \cap \mathbf{dom}(f)$
  Define the prior distribution parameters $\Theta = [\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^{\mathsf{n}}] \in \mathbb{R}_+^{\mathsf{k} \times \mathsf{n}}$
  Initialize $g_i^{(0)} \leftarrow g_i(\mathtt{Y})$ for all $i \in \mathcal{O}$
  Initialize $\mathtt{A}_i^{(0)} \leftarrow \left(\mathtt{Y}^{(0)}\right)^\top \mathtt{Y}^{(0)}$
  $t \leftarrow 0$
  while termination-condition **do**
    Compute U, V, J as in Proposition 1
    Compute ϵ^* as described in Sec. 3.2/3.3
    Update $\mathtt{A}^{(t+1)}$ as described in Sec. 3.3
    Update $\mathtt{Y}^{(t+1)}$ as described in Sec. 3.3
    Update $g_i^{(t+1)}$ as described in Sec. 3.3
    $t \leftarrow t + 1$
  end while
  return $\mathtt{Y}^{(t)}$

4 Related Work

The topic of clustering combination, also known as consensus clustering, has been very active in the last decade. A very recent and complete survey can be found in [9]. Several consensus methods have been proposed in the literature [1, 3, 10–14]. Some of them are based on estimates of similarity between partitions, others cast the problem as a categorical clustering problem, and finally others on similarity between data points (induced by the clustering ensemble). Our work belongs to this last type, where similarities are aggregated on the co-association matrix. Moreover there are methods that produce a crisp partition from the information provided by the ensemble and methods that induce a probabilistic solution, as our work.

In [15] the entries of the co-association matrix are also exploited and modeled using a generative aspect model for dyadic data, and producing a soft assignment. The consensus solution is found by solving a maximum likelihood estimation problem, using the Expectation-Maximization (EM) algorithm.

In a different fashion, other probabilistic approaches to consensus clustering that do not exploit the co-association matrix are [11, 16]. There, the input space directly consists of the labellings from the clustering ensemble. The model is based on a finite mixture of multinomial distribution. As usual, the model's parameters are found according to a maximum-likelihood criterion by using an EM algorithm. In [17], the idea was extended introducing a Bayesian version of the multinomial mixture model, the *Bayesian cluster ensembles*. Although the posterior distribution cannot be calculated in closed-form, it is approximated using variational inference and Gibbs sampling, in a very similar procedure as in *latent Dirichlet allocation* model [18, 19], but applied to a different input feature space. Finally, in [20], a nonparametric version of this work was proposed.

5 Experiments and Results

In this section we present the evaluation of our algorithm, using synthetic datasets, UCI data and two text-data benchmark datasets. We compare its performance against an algorithm that rely on the same type of data, and on similar assumptions, the Baum-Eagon (BE) [4] algorithm, which also extracts a soft consensus partition from the co-association matrix.

As in similar works, the performance of the algorithms is assessed using an external criterion of clustering quality, comparing the obtained partitions with the known ground truth partition. Given $\mathcal{O}$, the set of data objects to cluster, and two clusterings, $p_i = \{p_i^1, \ldots, p_i^h\}$ and $p_j = \{p_l^1, \ldots, p_l^k\}$, we chose the Consistency Index (CI) [1].

The Consistency Index, also called H index [21], gives the accuracy of the obtained partitions and is obtained by matching the clusters in the combined partition with the ground truth labels:

$$CI(p_i, p_l) = \frac{1}{\mathsf{k}} \sum_{k'=match(k)} m_{k,k'}, \tag{11}$$

where $m_{k,k'}$ denotes the contingency table, i.e. $m_{k,k'} = |p_i^k \cap p_l^{k'}|$. It corresponds to the percentage of correct labellings when the number of clusters in p_i and p_l is the same.

5.1 UCI and Synthetic Data

Following the usual strategy of producing clustering ensembles, and combining them on the co-association matrix, we created two different types of ensembles were created: (1) using k-means with random initialization and random number of clusters

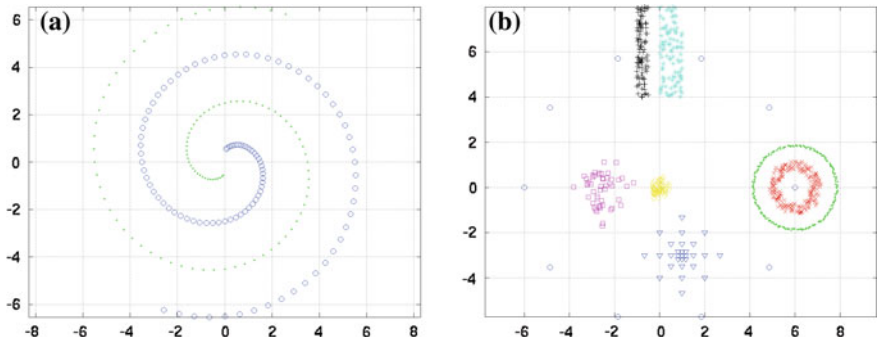


Fig. 1 Sketch of the synthetic datasets

Table 1 Benchmark datasets

Data-Sets	k	n	Ensemble
			k_u
Spiral	2	200	2–8
Cigar	4	250	4–20
Rings	3	450	3–20
Image-c	7	739	7–15, 20, 30
Image-1	8	1,000	8–15, 20, 30
Iris	3	150	3–10, 15, 20
Wine	3	178	3–10, 15, 20
House-votes	2	232	2–10, 15, 20
Ionosphere	2	351	2–10, 15, 20
std-yeast-cell	5	384	5–10, 15, 20
Breast-cancer	2	683	2–10, 15, 20
Optdigits	10	1,000	10, 12, 15, 20, 35, 50

[22]; (2) combining multiple algorithms (agglomerative hierarchical algorithms: single, average, ward, centroid link; k-means [23]; spectral clustering [24]) applied over subsampled versions of the datasets (subsampling percentage 0.9).

Table 1 summarizes the main characteristics of the UCI and synthetic datasets used on the evaluation, and the parameters used for generating ensemble (2). Figure 1 illustrates the synthetic datasets used in the evaluation: (a) spiral; (b) image-c. Tables 2 and 3 summarize the average performance of both algorithms over ensembles (1) and (2), after several runs, accounting for possible different solutions due to initialization, in terms of Consistency Index (CI). The λ parameter controls the regularization and when $\lambda = 1$ PEACE-MAP reduces to PEACE.

The performance of PEACE is globally better than BE, mainly when taking into account the effect of regularization. Comparing the performance of both ensembles: on ensemble (1), regularized PEACE has better performance than BE on 7 datasets

Table 2 Results for benchmark datasets, ensemble (1), in terms of CI index

Data-Sets	λ					BE
	1	5	10	20	50	
Spiral	**0.500**	**0.500**	**0.500**	**0.500**	**0.500**	**0.500**
Cigar	0.708	0.792	**0.836**	0.804	0.588	0.708
img-c	0.585	**0.831**	0.407	0.215	0.164	0.676
Rings	**0.682**	**0.682**	0.373	0.373	0.464	0.527
Image	**0.647**	0.660	0.443	0.267	0.198	0.608
Iris	0.967	0.967	**0.973**	0.513	0.493	0.969
Wine	0.966	**0.978**	**0.978**	0.927	0.573	0.970
House	**0.905**	**0.905**	**0.905**	**0.905**	**0.905**	**0.905**
Ionosphere	**0.778**	0.769	0.758	0.746	0.658	0.735
std-yeast	0.693	0.690	0.693	0.378	0.315	**0.695**
Breast	**0.969**	**0.969**	**0.969**	**0.969**	**0.969**	**0.969**
Optdigits	0.602	0.415	0.205	0.165	0.148	**0.741**

Table 3 Results for benchmark datasets, ensemble (2), in terms of CI index

Data-Sets	λ					BE
	1	5	10	20	50	
Cigar	0.624	0.640	0.668	0.672	0.316	**0.708**
Rings	0.558	0.407	0.504	0.664	0.540	**0.571**
Spiral	**0.505**	**0.505**	**0.505**	**0.505**	0.500	**0.505**
img-c	0.434	**0.590**	0.553	0.252	0.175	0.482
Image	0.602	0.604	**0.631**	0.235	0.188	0.465
Iris	**0.907**	0.900	0.707	0.513	0.613	0.707
Wine	0.961	0.961	**0.972**	0.601	0.646	**0.972**
House	0.875	**0.892**	**0.892**	0.875	0.858	0.875
Ionosphere	**0.632**	**0.632**	**0.632**	**0.632**	0.618	0.615
std-yeast	0.544	0.544	**0.615**	0.333	0.331	0.542
Breast	0.734	0.734	0.734	0.734	**0.736**	**0.736**
Optdigits	0.786	0.833	0.426	0.162	0.162	**0.886**

(over 12), and equal on 3 datasets, while on ensemble (2) it has better or equal performance that the other on 7 (over 12), and equal on 2 datasets. The regularized PEACE has almost every time better performance than unregularized PEACE (column $\lambda = 1$), but when λ increases to larger values the performance decreases drastically, indicating the regularization may have a negative effect if it is too strong.

5.2 Text Data

We also evaluated the proposed algorithm over two well known text-data benchmark datasets: the KDD mininewsgroups[1] and the webKD dataset.[2] The mininewsgroups dataset, is composed by usenet articles from 20 newsgroups. After removing three newsgroups not corresponding to a clear concept ('talk.politics.misc', 'talk.religion.misc', 'comp.os.ms-windows.misc'), we ended up analyzing 17 newsgroups, grouped in 7 macro-categories ('rec', 'comp', 'soc', 'sci', 'talk', 'alt', 'misc'). In this collection there are only 100 documents on each newsgroups, adding up to 1,700 documents.

The webKD dataset corresponds to WWW-pages collected from computer science departments of various universities in January 1997. We concentrated our analysis on 4 categories ('project', 'student', 'course', 'faculty'). For each, we analyzed only the documents belonging to universities ('texas', 'washington', 'wisconsin', 'cornell'), adding up to 1041 documents.

The analysis followed the usual steps for text-processing [25]: tokenization, stopword-removal, stemming (Porter Stemmer), feature weighting (using Tf-Idf) and feature removal. In the feature removal step, we removed tokens that appeared in less than 40 documents and words that had a low variance of occurrence. On the mininewsgroups dataset this feature removal step, led to 500-dimension term frequency vector, while on the webKD led to 335-dimension term frequency vector.

We build the clustering ensembles based on the split and merge strategy (ensemble (1)) using: K-means with cosine similarity - ensemble. For the generation we assumed that each partition had a random number of clusters, chosen in the interval $K = \{\sqrt{ns}/2; \sqrt{ns}\}$, where ns is the number of samples.

Figure 2 illustrates an example of the obtained co-association matrices. To allow a better understanding of obtained matrices, samples are aligned according to ground

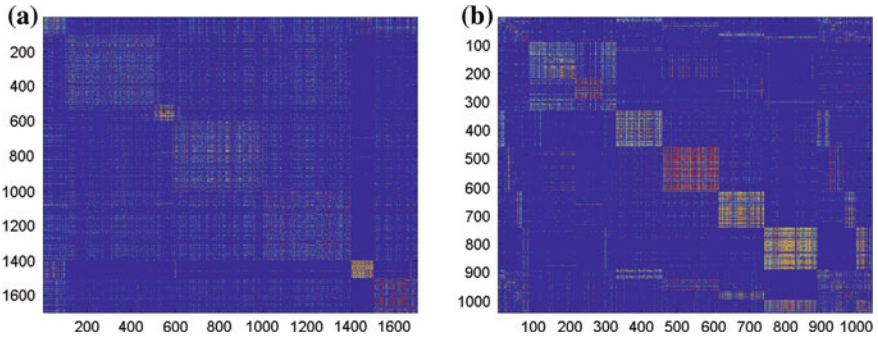


Fig. 2 Examples of obtained co-associations for miniNewsgroups and webKD datasets using an ensemble of K-means with cosine similarity. **a** miniNewsgroups. **b** webKD

[1] http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html.

[2] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/.

Table 4 Results for text datasets, in terms of CI index

Data-Sets	λ					BE
	1	5	10	20	50	
miniN	0.420	0.385	0.111	0.094	0.087	**0.435**
webKD	0.426	**0.493**	0.462	0.385	0.306	0.387

truth. The block-diagonal structure of the co-association of webKD dataset is much more evident than on the miniNewsgroups.

In Table 4 we summarize the obtained results for the PEACE (with and without regularization) and BE algorithm, indicating average result after several runs. Highest values for each data set are in bold. PEACE algorithm has better performance in ensembles exhibiting higher compactness properties. In this case we also experience a positive effect from the regularization. However, in situations where the co-association matrices have a less evident structure, with a lot of noise connecting clusters, its performances tend to decrease.

6 Conclusions

In this paper we have presented a probabilistic approach to consensus clustering based on the EAC paradigm. In our model, the entries of the co-association matrix are regarded as realizations of binomial random variables parameterized by unknown probabilistic assignments of objects to clusters. These parameters are estimated by adopting a maximum a-posteriori approach with Dirichlet prior distributions defined for each probabilistic cluster label assignment. We have studied the effect of the regularization on the consensus solution found. From the computational perspective, the optimization problem resulting from the MAP estimation is non-linear and non-convex and we addressed it using a primal line-search algorithm. Evaluation on both synthetic and real benchmarks data assessed the effectiveness of our approach.

It is interesting to note that if we have a-priori knowledge about one or more labels a point belongs to, we can add this knowledge through the Dirichlet priors by tuning the related parameters in Θ. This would allow to consider a semisupervised setting within the EAC framework.

Acknowledgments This work was partially financed by an ERCIM "Alain Bensoussan" Fellowship Programme under the European Union Seventh Framework Programme (FP7/2007–2013), grant agreement n. 246016, by FCT under grants SFRH /PROTEC/49512/2009, PTDC/EEI-SII/2312/2012 (LearningS project) and PEst-OE/ EEI/LA0008/2011, and by the *Área Departamental de Engenharia Electronica e Telecomunicações e de Computadores* of *Instituto Superior de Engenharia de Lisboa*, whose support the authors gratefully acknowledge.

References

1. Fred, A.: Finding consistent clusters in data partitions. In: Kittler, J., Roli, F. (eds.) Multiple Classifier Systems, pp. 309–318. Springer, Heidelberg (2001)
2. Fred, A., Jain, A.: Data clustering using evidence accumulation. In: Proceedings of the 16th International Conference on Pattern Recognition, pp. 276–280 (2002)
3. Fred, A., Jain, A.: Combining multiple clustering using evidence accumulation. IEEE Trans. Pattern Anal. Mach. Intell. **27**(6), 835–850 (2005)
4. Rota Bulò, S., Lourenço, A., Fred, A., Pelillo, M.: Pairwise probabilistic clustering using evidence accumulation. In: Proceedings of 2010 International Conference on Structural, Syntactic, and Statistical Pattern Recognition. SSPR&SPR'10, pp. 395–404 (2010)
5. Lourenço, A., Rota Bulò, S., Rebagliati, N., Figueiredo, M.A.T., Fred, A.L.N., Pelillo, M.: Probabilistic evidence accumulation for clustering ensembles (2013)
6. Luenberger, D.G., Ye, Y.: Linear and Nonlinear Programming, 3rd edn. Springer, Heidelberg (2008)
7. Boyd, S., Vandenberghe, L.: Convex Optimization, 1st edn. Cambridge University, Cambridge (2004)
8. Kachurovskii, I.R.: On monotone operators and convex functionals. Uspekhi Mat. Nauk **15**(4), 213–215 (1960)
9. Ghosh, J., Acharya, A.: Cluster ensembles. Wiley Interdisc. Rew. Data Min. Knowl. Disc. **1**(4), 305–315 (2011)
10. Strehl, A., Ghosh, J.: Cluster ensembles—a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. **3**, 583–617 (2002)
11. Topchy, A., Jain, A., Punch, W.: A mixture model of clustering ensembles. In: Proceedings of the SIAM Conference on Data Mining, April 2004
12. Dimitriadou, E., Weingessel, A., Hornik, K.: A combination scheme for fuzzy clustering. In: AFSS'02, pp. 332–338 (2002)
13. Ayad, H., Kamel, M.S.: Cumulative voting consensus method for partitions with variable number of clusters. IEEE Trans. Pattern Anal. Mach. Intell. **30**(1), 160–173 (2008)
14. Fern, X.Z., Brodley, C.E.: Solving cluster ensemble problems by bipartite graph partitioning. In: Proc ICML'04 (2004)
15. Lourenço, A., Fred, A., Figueiredo, M.: A generative dyadic aspect model for evidence accumulation clustering. In: Proceedings of 1st International Conference Similarity-based Pattern Recognition. SIMBAD'11, pp. 104–116. Springer, Heidelberg (2011)
16. Topchy, A., Jain, A.K., Punch, W.: Clustering ensembles: models of consensus and weak partitions. IEEE Trans. Pattern Anal. Mach. Intell. **27**(12), 1866–1881 (2005)
17. Wang, H., Shan, H., Banerjee, A.: Bayesian cluster ensembles. In: 9th SIAM International Conference on Data Mining (2009)
18. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proc. Nat. Acad. Sci. USA **101**(Suppl 1), 5228–5235 (2004)
19. Steyvers, M., Griffiths, T.: Latent semantic analysis: a road to meaning. In: Probabilistic Topic Models. Laurence Erlbaum (2007)
20. Wang, P., Domeniconi, C., Laskey, K. B.: Nonparametric bayesian clustering ensembles. In: ECML PKDD'10, pp. 435–450 (2010)
21. Meila, M.: Comparing clusterings by the variation of information. In: Proceedings of the Sixteenth Annual Conference of Computational Learning Theory (COLT). Springer, Heidelberg (2003)
22. Lourenço, A., Fred, A., Jain, A.K.: On the scalability of evidence accumulation clustering. In: 20th International Conference on Pattern Recognition (ICPR), Istanbul Turkey, pp. 782–785, Aug 2010
23. Jain, A.K., Dubes, R.: Algorithms for Clustering Data. Prentice Hall, New Jersey (1988)
24. Ng, A.Y., Jordan, M.I., Weiss, Y.: On spectral clustering: analysis and an algorithm. In: NIPS, pp. 849–856. MIT, Cambridge (2001)
25. Manning, C.D., Raghavan, P., Schtze, H.: Introduction to Information Retrieval. Cambridge University, New York (2008)

Feature Discretization with Relevance and Mutual Information Criteria

Artur J. Ferreira and Mário A.T. Figueiredo

Abstract *Feature discretization* (FD) techniques often yield adequate and compact representations of the data, suitable for machine learning and pattern recognition problems. These representations usually decrease the training time, yielding higher classification accuracy while allowing for humans to better understand and visualize the data, as compared to the use of the original features. This paper proposes two new FD techniques. The first one is based on the well-known Linde-Buzo-Gray quantization algorithm, coupled with a relevance criterion, being able perform unsupervised, supervised, or semi-supervised discretization. The second technique works in supervised mode, being based on the maximization of the mutual information between each discrete feature and the class label. Our experimental results on standard benchmark datasets show that these techniques scale up to high-dimensional data, attaining in many cases better accuracy than existing unsupervised and supervised FD approaches, while using fewer discretization intervals.

Keywords Classification · Feature discretization · Linde-Buzo-Gray · Mutual information · Quantization · Relevance · Supervised learning

1 Introduction

Feature discretization (FD) [5, 20] represents a numeric (integer or real) feature by a set of discrete values from a finite alphabet. FD has been extensively considered in

A.J. Ferreira (✉)
ADEETC, Instituto Superior de Engenharia de Lisboa, Gab. 16,
Rua Conselheiro Emidio Navarro, 1, 1959-007 Lisbon, Portugal
e-mail: arturj@isel.pt

M.A.T. Figueiredo
Instituto Superior Tecnico Instituto de Telecomunicacoes - Torre Norte,
piso 10, Av. Rovisco Pais, 1, 1049-001 Lisbon, Portugal
e-mail: mario.figueiredo@lx.it.pt

A.J. Ferreira · M.A.T. Figueiredo
Instituto de Telecomunicações, Lisbon, Portugal

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_7

the past, due to its several benefits. In some cases, features have noisy values or show minor fluctuations that are irrelevant or even harmful for the learning task at hand. For such features, the performance of machine learning and data mining algorithms can be improved by discretization. The storage space necessary for the data can be dramatically decreased, after FD. Another common benefit of FD techniques is the improvement of the quality of data visualization. Moreover, some learning algorithms require a discrete representation of the data. In summary, FD provides compact representations, with lower memory usage, while at the same time it may reduce the training time and improve the classification accuracy. The literature on FD includes many techniques; see for instance [5, 11, 16, 20].

1.1 Our Contribution

In this paper, we propose two new FD techniques to discretize data with a variable number of bits per feature. Both techniques start with a coarse discretization and assess relevance as discretization is carried out. Whenever there is not a significant increase in the relevance of a given feature, discretization is halted for that feature. Otherwise, it allocates one more bit for that feature.

The first technique applies the Linde-Buzo-Gray (LBG) [15] quantization algorithm, guided by a relevance criterion, being applicable to unsupervised, supervised, or semi-supervised learning. It aims at finding features with low *mean square error* (MSE) with the continuous feature and high relevance with the class label. The second technique is supervised being based on maximization of the *mutual information* (MI) [4] between each (discretized) feature and the class label. MI acts as a relevance measure for discretization purposes, allocating more bits to the most relevant features.

The remainder of this paper is organized as follows. Section 2 reviews some background on information theory measures, as defined by both Shannon and Renyi and discusses the key issues of FD techniques, describing some unsupervised and supervised approaches. Section 3 details both our methods for FD. Section 4 reports the experimental evaluation of our methods compared against other unsupervised and supervised techniques, on public benchmarks. Finally, Sect. 5 provides some concluding remarks and directions for future work.

2 Background

In this section, we briefly review the concept of MI from information theory for discrete random variables, as defined by Shannon and Renyi (Sect. 2.1). We also review a taxonomy on FD techniques, describing their key benefits and drawbacks (Sect. 2.2) along with a description of successful unsupervised and supervised techniques in Sects. 2.3 and 2.4, respectively.

2.1 Entropy and Mutual Information

Let X and Y be two discrete *random variables* (RV) that take values in their corresponding sets $\mathcal{X}_i$ and $\mathcal{Y}_i$. The entropy [4], in the Shannon sense, of X, denoted $H_s(X)$ is defined as

$$H_s(X) = -\sum_{X_i \in \mathcal{X}_i} p(X_i) \log_2(p(X_i)), \tag{1}$$

ranging from $0 \leq H_s(X) \leq \log_2(N)$, with $N = |\mathcal{X}_i|$. The entropy is zero if one of the outcomes happens with probability one (there is no uncertainty about X). It attains its maximum value for an uniform probability mass function of X, which represents maximum uncertainty about the outcomes.

Mutual information (MI) [4] as defined by Shannon, measures the dependency between two random variables

$$MI_s(X; Y) = \sum_{X_i \in \mathcal{X}_i} \sum_{Y_j \in \mathcal{Y}_j} p(X_i, Y_j) \log_2 \left(\frac{p(X_i, Y_j)}{p(X_i)\, p(Y_j)} \right), \tag{2}$$

in which $p(X_i, Y_j)$ is the joint probability of X_i and Y_j outcomes. MI is non-negative, being zero if and only if X and Y are statistically independent. In the opposite case, if X and Y are dependent we get $MI_s(X; Y) = \min\{H_s(X), H_s(Y)\}$. MI can also be expressed by

$$MI_s(X; Y) = H_s(X) - H_s(X|Y) = H_s(Y) - H_s(Y|X), \tag{3}$$

as functions of the individual $H_s(.)$ and conditional entropies $H_s(.|.)$. In order to maximize (3), one must minimize one of the conditional entropies. For instance, we can choose to minimize $H_s(X|Y)$, that is, the uncertainty about X, given a known Y. We have $0 \leq H_s(X|Y) \leq H_s(X)$, with $H_s(X|Y) = 0$ meaning deterministic dependence. On the other extreme case, we get $H_s(X|Y) = H_s(X)$ corresponding to statistical independence between X and Y. The concepts of entropy and MI as proposed by Shannon were later generalized by Renyi [17] in the decade of 1960. The MI as defined by Renyi is

$$MI_r^{\alpha}(X; Y) = \frac{1}{\alpha - 1} \log_2 \left(\sum_{X_i \in \mathcal{X}_i} \sum_{Y_j \in \mathcal{Y}_j} \frac{p^{\alpha}(X_i, Y_j)}{p^{\alpha-1}(X_i)\, p^{\alpha-1}(Y_j)} \right), \tag{4}$$

with $\alpha \neq 1$; notice that $\lim_{\alpha \to 1} MI_r^{\alpha} = MI_s$, as defined by (2). For a detailed discussion of the extensions proposed by Renyi, their properties and applications the interested reader is referred to [17].

2.2 Feature Discretization

A typical dataset is usually composed by categorical and numeric features. The former are discrete by nature whereas the latter use real or integer representations. In some cases, the numeric features may have noise or may exhibit minor fluctuations which degrade the performance of the learning task.

Regardless of the type of classifier considered, FD techniques aim at finding a representation of each feature that contains enough information for the learning task at hand, while ignoring minor fluctuations that may be irrelevant for that task. The use of a discretization technique will lead to a more compact (using less memory), and hopefully to a better representation of the data for learning purposes, as compared to the use of the original features. For these reasons, the use of discretization algorithms has played an important role in data mining. They produce a concise representation of continuous features allowing the users to understand the data more easily, making learning more accurate and faster.

It has been found that the use of FD techniques may improve the results of many learning methods [5, 20]. Although *supervised* discretization (i.e., making use of the class labels) may, in principle, lead to better classifiers, the literature on FD reports that *unsupervised* FD methods (which do not use the class labels) perform well on different types of data (see for instance [21]). FD methods can also be classified as *dynamic* or *static* [5, 20]; while static methods treat each feature independently, dynamic methods try to quantize all features simultaneously, thus taking into account feature interdependencies. FD methods can also be categorized as *local* (discretization of some features based on a decision mechanism such as learning a tree) or *global* (discretize all the features). These methods can work in a *top-down* (splitting) or a *bottom-up* (merging) approach, regarding on how they construct the binary codes for each feature. Finally, in *direct* FD methods, one decides *a priori* on the number of bits per feature, whereas *incremental* methods start with a coarse discretization pass for all features and subsequently allocate more bits to each feature.

The quality of discretization is usually assessed by two indicators: the *generalization error* and the *complexity*, *i.e.*, the number of intervals or equivalently the number of bits used to represent each instance. A possible drawback of FD is arguably the (time and memory) cost of the discretization procedure.

For reviews on FD methods please see the works of [5, 11, 13, 16] and the many references therein. A very recent survey of FD algorithms classified according to the taxonomy described above, along with an experimental evaluation can be found in [9].

2.3 Unsupervised Discretization

The most common techniques for unsupervised FD are [20]:

- *equal-interval binning* (EIB), which performs uniform quantization;

- *equal-frequency binning* (EFB) [3], which obtains a non-uniform quantizer with intervals such that, for each feature, the number of occurrences in each interval is the same;
- *proportional k-interval discretization* (PkID) [21], which computes the number and size of the discretization intervals as functions of the number of training instances.

EIB is the simplest and easiest to implement, but is sensitive to outliers. In EFB, the quantization intervals are smaller in regions where there are more occurrences of the values of each feature; EFB is thus less sensitive to outliers, as compared to EIB. In the EIB and EFB methods, the user can choose the number of discretization bins (static discretization). In contrast, PkID is an incremental method since it sets the number and size of the discretization intervals as a function of the number of training instances, seeking a trade-off between bias and variance of the class probability estimate of a naïve Bayes classifier [21].

Recently, we have proposed [7] an unsupervised scalar FD method based on the LBG algorithm [15]. For a given number of discretization intervals, LBG discretizes the data seeking the minimum MSE with respect to the original representation. This incremental approach, named *unsupervised* LBG (U-LBG 1) applies the LBG algorithm to each feature independently and stops when the MSE falls below a threshold Δ or when the maximum number of bits q per feature is reached. A static variant of U-LBG1, named U-LBG2, using a fixed number of bits per feature q was also proposed. Both U-LBG1 and U-LBG2 rely on the idea that a discrete representation with low MSE with the original feature representation is adequate for learning.

2.4 Supervised Discretization

This Section briefly reviews the most common techniques for supervised FD. The *information entropy minimization* (IEM) method [6], based on the *minimum description length* (MDL) principle, is one of the oldest and most often used methods for supervised FD. The key idea of using the MDL principle is that the most informative features to discretize are the most compressible ones. The IEM method is based on the use of the entropy minimization heuristic for discretization of a continuous value into multiple intervals. IEM adopts a recursive approach computing the discretization cut-points in such a way that they minimize the amount of bits needed to represent the data. It follows a top-down approach, starting with one interval and splits intervals as discretization is carried out. The method termed *IEM variant* (IEMV) [12] is also based on an entropy minimization heuristic to choose the discretization intervals. It applies a function, based on the MDL principle, which decreases as the number of different values for a feature increases.

The supervised static *class-attribute interdependence maximization* (CAIM) [14] algorithm aims to maximize the class-attribute interdependence and to generate a (possibly) minimal number of discrete intervals. The algorithm does not require a predefined number of intervals, as opposed to some other FD methods. Experimental

results reported show that CAIM compares favorably with six other FD discretization algorithms, in that the discrete attributes generated by CAIM almost always have the lowest number of intervals and the highest class-attribute interdependency, achieving the highest classification accuracy [14]. The *class-attribute contingency coefficient* (CACC) algorithm [19], is an incremental, supervised, top-down FD method, that has been shown to achieve promising results regarding execution time, number of discretization intervals, and training time of the classifiers. For a very recent survey with an extensive list of supervised FD algorithms please see [9].

3 Proposed Methods

3.1 Relevance-Based LBG

As in U-LBG1, our first FD proposal, named *relevance-based* LBG (R-LBG) and described in Algorithm 1, uses the LBG algorithm, discretizing data with a variable number of bits per feature. We use a relevance function, denoted $@rel$, and a (non-negative) stopping factor, ϵ. The relevance function, producing non-negative values, is applied after each discretization. R-LBG behaves differently, depending on the value of ϵ. If ϵ is positive, whenever there is an increase below ϵ on the relevance between two subsequent discrete versions (with b and $b+1$ bits), discretization is halted and b bits are kept, for that feature; otherwise, with a significant (larger than ϵ) increase on the relevance, it discretizes with one more bit, assessing the new relevance.

R-LBG discretizes a feature with an increasing number of bits, stopping only when there is no significant increase on the relevance of the recently discretized feature. If $\epsilon = 0$, each feature is discretized from 1 up to the maximum q bits and the corresponding relevance is assessed on each discretization. Then, the minimum number of bits that ensures the highest relevance is kept and applied to discretize that feature. Regardless of the value of ϵ, the method discretizes data with a variable number of bits per feature aiming at producing discrete features with high relevance and low MSE with the original representation. The relevance assessment $r_{ib} = @rel(Q_b^i(X_i); ...)$, of feature i with b bits, in line 5 of Algorithm 1, can refer to unsupervised, supervised, or semi-supervised learning. This depends on how the relevance function makes use (or not) of the class labels. The value of ϵ, when different from zero, should be set according to the range of the $@rel$ function. There are many different choices for the relevance criterion to apply in R-LBG. In the unsupervised case, if we consider $@rel = MSE$ (between original and discrete features) we have the unsupervised U-LBG1 approach. Another relevance criterion is given by the quotient between the variance of the discrete feature and the number of discretization intervals

$$NVAR(\widetilde{X}_i) = \text{var}(\widetilde{X}_i)/2^{b_i}, \tag{5}$$

Algorithm 1: R-LBG - Relevance-based LBG.

Input: X: $n \times d$ matrix training set (n patterns, d features).
y: n-length vector with class labels (supervised).
q: maximum number of bits per feature.
$@rel, \epsilon$ (≥ 0): relevance function, stopping factor.
Output: $\widetilde{X}$: $n \times d$ matrix, discrete feature training set.
$Q^1_{b_1}, ..., Q^d_{b_d}$: set of d quantizers (one per feature).

1: **for** $i = 1$ to d **do**
2: pRel = 0; {/* Initial/previous rel. for feature i. */}
3: **for** $b = 1$ to q **do**
4: Apply LBG to the i-th feature to obtain a b-bit quantizer $Q^i_b(\cdot)$;
5: Compute $r_{ib} = @rel(Q^i_b(X_i); ...)$, relevance of feature i with b bits;
6: **if** $(\epsilon == 0)$ **then**
7: continue; {/* Discretize up to q bits. */}
8: **end if**
9: **if** $((r_{ib} - pRel) > \epsilon)$ **then**
10: $Q^i(\cdot) = Q_b(\cdot)$; {/* High increase. Store quantizer. */}
11: $\widetilde{X}_i = Q^i_b(X_i)$; {/* Discretize the feature. */}
12: **else**
13: break; {/* Small increase. Break loop. Move on to the next feature. */}
14: **end if**
15: pRel = r_{ib}; {/* Keep previous relevance. */}
16: **end for**
17: **end for**
18: **if** $(\epsilon == 0)$ **then**
19: **for** $i = 1$ to d **do**
20: Get $b_i = \arg \max_{b \in \{1,...,q\}} r_{i*}$ {/* Minimum bits for maximum relevance. */}
21: $Q^i(\cdot) \leftarrow$ Apply LBG (b_i bits) to the i-th feature;
22: $\widetilde{X}_i = Q^i_{b_i}(X_i)$; {/* Discretize feature. */}
23: **end for**
24: **end if**

where b_i is the number of bits of the discrete feature. For the supervised case, we propose to compute relevance by the MI, defined by Shannon (3) or Renyi (4) (see Sect. 2.1) between the discretized features $\widetilde{X}_i$, with b_i bits and the class label vector $\mathbf{y}$. There are many other (unsupervised and supervised) feature relevance criteria; in fact, all the criteria used in *feature selection* (FS) methods to rank features are suited to serve as the relevance measure in R-LBG. The relevance function can also use the class label for those instances for which it is available, thus being usable in semi-supervised learning.

As an illustration of the supervised case, Fig. 1 (top) shows the evolution of MI_s defined by (3) between the class label and some of the features discretized by the R-LBG algorithm, using $\epsilon = 0.1$ and $q \in \{1, \ldots, 10\}$ bits per feature, for the two-class Hepatitis and three-class Wine datasets. In the bottom plot, we compare the MI values obtained by discretizing with $q = 1$ and $q = 10$ bits, for all the features in each dataset. On the Hepatitis dataset, the top plot shows that for features 1, 12, and 14, the MI grows with the number of bits and then it levels off. For feature 12 (which is categorical, thus originally discrete), as expected, an increasing number

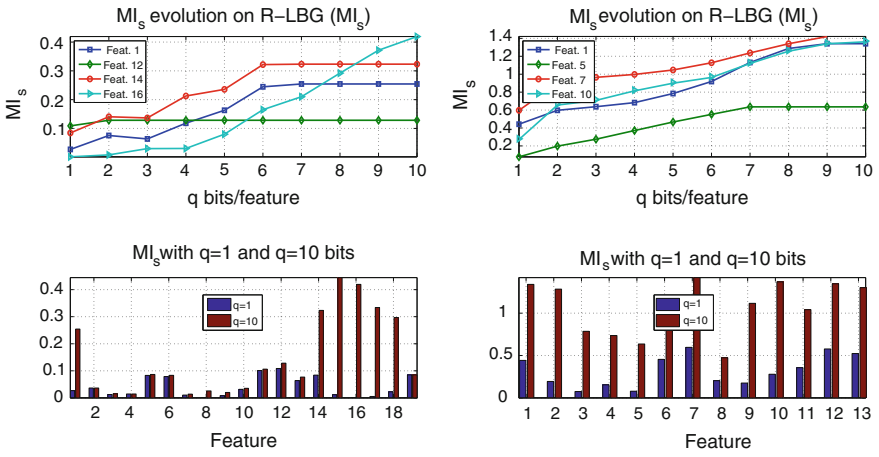


Fig. 1 R-LBG (MI_s,$\epsilon = 0.1$) discretization on the Hepatitis (*left*) and Wine (*right*) datasets. *Top*: MI_s as a function of the number of bits $q \in \{1, \ldots, 10\}$, for four features on each dataset. *Bottom*: MI_s with $q = 1$ and $q = 10$ bits, for all the features

of bits does not lead to a higher MI. Thus, R-LBG can handle both numeric and categorical features and the relevance values provide a feature ranking score. In the bottom plot, we see that some features, such as 3, 4, 5, and 6, show no significant MI increase, when moving from $q = 1$ to $q = 10$. On the other hand, for features 14 to 18, we have a strong MI increase, which usually corresponds to numeric informative features. On the three-class problem posed by the Wine dataset, in the top plot we see that MI levels off at $q = 7$ bits for feature 5, whereas features 1, 7, and 10 keep on increasing their MI with more than 7 bits. In the bottom plot, we observe a large increase on the MI by moving from $q = 1$ to $q = 10$ bit. In practice, the choice of adequate values for ϵ, which depends on the type of data, can be done using these plots, by checking how the MI increases on each particular type of data.

3.2 Mutual Information Discretization

In this Section, we present our supervised FD method, named *mutual information discretization* (MID). Essentially, the MID method consists in discretizing each feature individually, computing the discretization cut-points in order to maximize the MI of the discrete feature with the class label. The key motivation for this FD proposal is that the MI between features and class labels has been successfully and extensively used as a relevance criterion for FS purposes, in the past two decades; see the seminal work in [1] and [2] for a review of MI-based FS methods. It is thus expectable that a good criterion for FS will also be adequate for FD. The usual argument on the adequacy of MI for learning purposes is based on bounds for the probability of error

which depend on the MI between the observations and the class label, namely the Hellman-Raviv [10] and Santhi-Vardi bounds [2, 18]. The Hellman-Raviv bound on the Bayes risk is given by

$$err_{Bayes}(\widetilde{X}_i) \leq \frac{1}{2} H_s(\mathbf{y}|\widetilde{X}_i), \tag{6}$$

and the Santhi-Vardy bound [18] is

$$err_{Bayes}(\widetilde{X}_i) \leq 1 - 2^{-H_s(\mathbf{y}|\widetilde{X}_i)}. \tag{7}$$

Applying (3) between a discrete feature $\widetilde{X}_i$ and the class label vector $\mathbf{y}$, we get

$$MI_s(\widetilde{X}_i; \mathbf{y}) = H_s(\widetilde{X}_i) - H_s(\widetilde{X}_i|\mathbf{y}) = H_s(\mathbf{y}) - H_s(\mathbf{y}|\widetilde{X}_i). \tag{8}$$

Thus, in order to maximize the MI in (8), one must minimize $H_s(\widetilde{X}_i|\mathbf{y})$, that is, the uncertainty about the feature value, given a known class label. We have $0 \leq H_s(\widetilde{X}_i|\mathbf{y}) \leq H_s(\widetilde{X}_i)$, with $H_s(\widetilde{X}_i|\mathbf{y}) = 0$ meaning deterministic dependence (an ideal feature) and $H_s(\widetilde{X}_i|\mathbf{y}) = H_s(\widetilde{X}_i)$ corresponding to independence between the feature and the class label (a useless feature). On the other hand, $H_s(\mathbf{y})$ does not change with discretization; thus, maximizing (8) is equivalent to minimizing $H_s(\mathbf{y}|\widetilde{X}_i)$, that is, the uncertainty about the class label given the feature. We have $0 \leq H_s(\mathbf{y}|\widetilde{X}_i) \leq H_s(\mathbf{y})$, with $H_s(\mathbf{y}|\widetilde{X}_i) = 0$ corresponding to deterministic dependence (again, the ideal case) and $H_s(\mathbf{y}|\widetilde{X}_i) = H_s(\mathbf{y})$ meaning independence (a useless feature). For an ideal feature (one that is a deterministic injective function of the class label), we have

$$MI_s(\widetilde{X}_i; \mathbf{y}) = \min\{H_s(\widetilde{X}_i), H_s(\mathbf{y})\}. \tag{9}$$

The maximum possible value for $MI_s(\widetilde{X}_i; \mathbf{y})$ depends on both the number of bits used to discretize X_i and the number of classes K. If we discretize X_i with b_i bits, its maximum entropy is $Hs_{max}(\widetilde{X}_i) = b_i$ bit/symbol; the maximum value of the class entropy is $Hs_{max}(\mathbf{y}) = \log_2(K)$ bit/symbol, which corresponds to K equiprobable classes. We thus conclude that the maximum value of the MI between the class label and a discretized feature (with b_i bits) is

$$MIs_{max}(\widetilde{X}_i; \mathbf{y}) = \min\{b_i,\ \log_2(K)\}. \tag{10}$$

In the binary case $K = 2$, we have $MIs_{max}(\widetilde{X}_i; \mathbf{y}) = 1$ bit. Moreover, to attain the maximum possible value for the MI, one must choose the maximum number of bits q taking into account this expression; this implies that $q \geq \lceil \log_2(K) \rceil$, which is more meaningful for multi-class problems.

At the discretization stage, we search for discretization boundaries such that the resulting discrete feature has the highest MI with the class label. Thus, as described above, by maximizing the MI at each cut-point we are aiming at leveraging the performance of the discrete feature, leading to higher accuracy. The method works

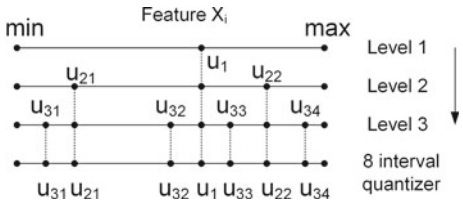


Fig. 2 The incremental and recursive partition procedure for FD, using $q = 3$ bits, leading to a 8-interval quantizer. On each level, the cut points u_* are computed to maximize the MI between the resulting discrete feature and the class label

in a recursive way, by successively breaking each feature into intervals, as depicted in Fig. 2 with $q = 3$ bits, yielding a 8-interval non-uniform quantizer.

We propose two versions of the MID technique. The first, named ***MID fixed***, uses a fixed number of q bits per feature. In summary, given a training set with n instances and d features, $\mathbf{X}$ and a maximum number of bits per feature q, the MID fixed method applies the recursive discretization method described in Fig. 2, using up to q bits per feature, yielding quantizer $Q_i(\cdot)$ for feature i and the discretized feature $\widetilde{X}_i = Q_i(X_i)$. The second version, named ***MID variable*** allocates *up to* q bits per feature, leading to a variable number of bits per feature. As in R-LBG, we halt the bit allocation for feature X_i with b bits, whenever its discretization with $b + 1$ bits does not lead to a significant increase (larger than ϵ) on the $MI(\widetilde{X}_i; \mathbf{y})$. As a consequence, the MID variable version will produce fewer discretization intervals (and thus fewer bits per instance), as compared to the MID fixed method. By setting $\epsilon = 0$, MID variable discretizes feature $\widetilde{X}_i$, with maximum MI, using the smallest possible number of bits $b_i \leq q$ (it acts in a similar fashion as R-LBG). The number of discretization intervals depends on the value of ϵ; larger values will lead to fewer intervals, since discretization is stopped at earlier stages.

Figure 3 (top) plots the evolution of MI_s for some features of the Hepatitis and Wine datasets. In the bottom plot, we compare the MI_s values obtained by discretizing with $q \in \{1, 2, 3, 4\}$ bits, for all the features in each dataset. On both datasets, we observe in the top plots an increase in the first few bits and then the values of MI_s level off. In the bottom plot, we see an overall increase of the MI_s when moving from 1 to 3 bits; however, using one more bit per feature (with $q = 4$), there is no appreciable increase on the MI. As compared to Fig. 1, that uses the same features of the same datasets, we have a much faster growing in the MI_s values, yielding high values of MI_s with less bits.

4 Experimental Evaluation

This section reports experimental results of our FD techniques on several public benchmark datasets, for the task of supervised classification. Table 1 presents the 14 datasets used in the experiments, publicly available from the *university of California*

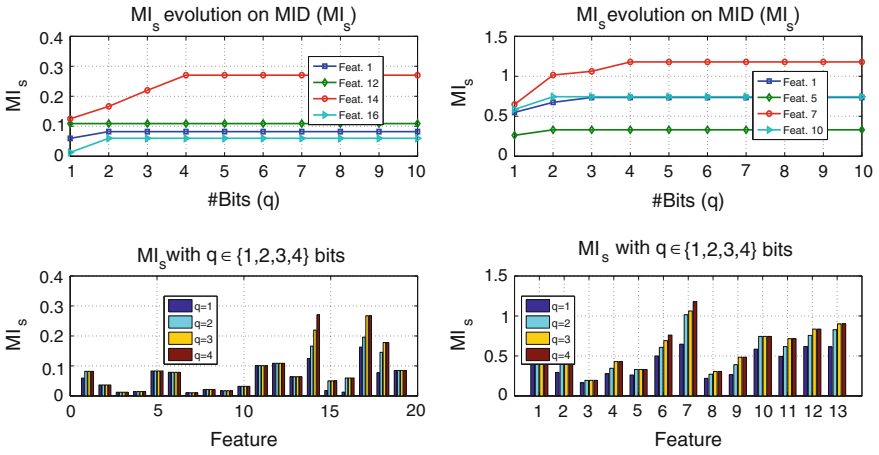


Fig. 3 MID (MI_s) discretization on the Hepatitis (*left*) and Wine (*right*) datasets. *Top* MI as a function of the number of bits $q \in \{1, \ldots, 10\}$, for 4 features on each dataset. *Bottom* MI with $q \in \{1, 2, 3, 4\}$ bits, for all the features

at Irvine (UCI) repository [8], from the *gene expression model selector* (GEMS),[1] the *Arizona state university* (ASU) repository,[2] or the *knowledge extraction based on evolutionary learning* (KEEL) repository.[3] In order to assess the performance of our

Table 1 The 14 datasets used in the experiments; d, K, and n are the number of features, classes, and instances, respectively

Dataset	d	K	n	Problem	Dataset	d	K	n	Problem
Car	6	4	1,728	Car acceptability	SpamBase	57	2	4,601	Email SPAM
Bupa	6	2	345	Liver disorders	Sonar	60	2	208	Sonar signal
Heart	13	2	270	Heart disease detection	Colon	2,000	2	62	Colon cancer detection
Wine	13	3	178	Classify wine cultivar	AR10P	2,400	10	130	Face database
Zoo	17	7	101	Classify animals	PIE10P	2,420	10	210	Face database
Hepatitis	19	2	155	Hepatitis detection	Leukemia	7,129	2	72	Cancer detection
Ionosphere	33	2	351	Radar signal return	Dexter	20,000	2	2,600	Text classification

[1] www.gems-system.org.

[2] http://featureselection.asu.edu/datasets.php.

[3] http://sci2s.ugr.es/keel/datasets.php.

FD methods, we have used two classifiers, namely: linear *support vector machines* (SVM) and *naïve Bayes* (NB) from the Weka toolkit,[4] with their default parameters. The classification accuracy is assessed using 10-fold *cross validation* (CV); on each CV fold, the FD methods are applied to the training set to learn a quantizer, which is then applied to the test set.

The experimental results are organized as follows. Section 4.1, evaluates the behavior of our supervised FD methods using a variable number of bits per feature. In Sect. 4.2, we compare our methods with existing unsupervised and supervised FD techniques (reviewed in Sects. 2.3 and 2.4, respectively). This evaluation is focused on both the *complexity* and the *generalization error*.

4.1 Comparison Between Our Approaches

In this Section, we analyze the behavior of our approaches. For both the R-LBG and MID variable algorithms, we assess the number of discretization intervals and the generalization error. Table 2 reports experimental results with the average number of bits per instance (with $q = 3$ and $\epsilon \in \{0, 0.05\}$) and the test set error rate for the linear SVM classifier (*No FD* denotes the use of the original features). On the R-LBG algorithm, $\epsilon = 0$ usually leads to a larger number of bits per instance, as compared with $\epsilon = 0.05$. This happens because with $\epsilon = 0$ we are aiming at finding the maximum relevance, whereas with $\epsilon > 0$ we halt the discretization process at earlier stages. For the MID variable algorithm, $\epsilon = 0$ leads to the choice of the minimum bits per feature that ensure the maximum MI; for this reason, with $\epsilon = 0$ we usually have fewer bits per instance as compared with $\epsilon > 0$. Regarding the classification accuracy, $\epsilon = 0$ usually attains the best results with a few exceptions and the MID variable algorithm attains better results than R-LBG. The use of Renyi's MI usually leads to less bits per instance, as compared to Shannon's MI; for the Sonar dataset, it also leads to lower generalization error.

Figure 4 shows the evolution of both the number of bits/instance and the test set error rate for a 10-fold CV of the NB classifier on data discretized by R-LBG and MID variable on the Wine dataset. On the left-hand side, we use $q = 5$ bits and ϵ in the real interval from 0 to 0.3; on the right-hand-side, we show the effect of varying the maximum number of bits for discretization, $q \in \{1, \ldots, 10\}$, while keeping a fixed $\epsilon = 0.05$.

As ϵ increases, the number of discretization intervals and thus the number of bits per instance decreases. The test set error rate is unacceptably high for $\epsilon > 0.15$ for R-LBG, whereas MID variable shows a stable behavior with respect to the increase of this parameter. Using MID variable, the test set error rate does not increase so fast as in R-LBG, whenever the number of bits per instance decreases. By increasing the maximum number of bits per feature, MID variable uses fewer bits per instance as compared to the R-LBG algorithm. The test set error rates are similar and both

[4] www.cs.waikato.ac.nz/ml/weka.

Table 2 Evaluation of R-LBG ($@rel = MI$) and MID variable with $q = 3$ and $\epsilon \in \{0, 0.05\}$

		MI by Shannon, as in (2)				MI by Renyi, $\alpha = 2$, as in (4)			
		R-LBG (MI)		MID var.		R-LBG (MI)		MID var.	
Dataset	No FD	$\epsilon = 0$	$\epsilon = 0.05$	$\epsilon = 0$	$\epsilon = 0.05$	$\epsilon = 0$	$\epsilon = 0.05$	$\epsilon = 0$	$\epsilon = 0.05$
Car		12.0	11.0	**6.0**	10.0	12.0	12.0	**6.0**	10.0
	21.1	21.7	26.3	**13.9**	**13.9**	21.7	23.3	**13.9**	**13.9**
Bupa		17.2	18.0	16.1	17.8	17.2	17.8	**6.0**	13.6
	42.0	40.3	42.0	**31.9**	32.5	40.3	42.0	34.8	35.1
Heart		27.5	24.6	22.1	21.2	27.5	23.4	**13.0**	18.4
	15.9	15.2	15.6	**14.1**	**14.1**	15.2	15.9	15.2	14.4
Wine		39.0	33.5	36.5	33.9	38.9	33.8	**13.0**	**13.0**
	1.1	2.2	1.6	2.2	2.2	1.6	1.6	1.6	1.6
Zoo		18.0	18.4	17.0	17.4	18.0	18.0	**16.0**	**16.0**
	7.9	6.9	6.9	**3.9**	**3.9**	6.9	6.9	**3.9**	**3.9**
Hepatitis		39.6	42.9	28.1	40.9	39.7	40.7	**19.0**	33.0
	22.0	20.1	21.4	**14.9**	15.0	19.4	22.1	19.4	18.1
Ionosphere		97.0	88.3	69.8	60.4	96.8	89.4	**33.0**	**33.0**
	11.9	11.9	12.5	7.4	**5.4**	11.9	11.6	8.0	8.0
SpamBase		159.6	152.6	128.4	106.0	159.9	147.9	**54.0**	92.0
	10.2	9.4	13.2	**6.4**	6.5	9.3	12.1	6.6	6.7
Sonar		176.2	160.3	159.1	144.1	176.0	149.2	**60.0**	89.4
	24.6	21.2	26.6	22.6	20.7	21.2	23.2	**20.3**	20.7
Colon		5,885.3	5,396.7	4,675.1	4,540.0	5,869.1	5,334.5	**2,000.0**	2,319.0
	16.2	19.3	21.0	17.6	17.6	19.3	21.0	22.1	22.1
AR10P		7,198.9	7,190.0	7,050.5	7,050.1	7,199.4	7,195.3	**2,400.0**	**2,400.0**
	1.0	**0.0**	**0.0**	2.0	2.0	**0.0**	**0.0**	6.5	6.5
PIE10P		7,259.3	7,241.1	7,164.9	7,164.7	7,259.4	7,252.2	**2,420.0**	**2,420.0**
	0.5	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
Leukemia		20,974.5	19,248.6	17,353.3	17,562.9	20,913.1	18,948.4	**7,129.0**	8,411.6
	1.3	2.5	2.5	**1.3**	**1.3**	2.5	**1.3**	4.2	4.2
Dexter		11,206.5	21,894.0	7,851.3	21,855.7	11,195.4	21,851.4	**7,312.7**	21,744.9
	8.3	9.0	16.0	6.3	6.3	v8.7	12.0	7.0	**5.7**

For each dataset, the first row contains the number of bits per instance and the second row the test set error rate (%), of a 10-fold CV for the linear SVM classifier. The discretization with less bits and the best error rate are in bold face

algorithms exhibit a stable behavior in the sense that an (excessive) increase on the maximum number of bits per feature q does not lead to a degradation on this indicator, due to the incremental procedure that stops allocating bits, whenever the relevance criterion is not fulfilled. Our methods show stability regarding the variation of their input parameters q and ϵ. We have found that in most cases, by setting ϵ from 0 to 10 % of the maximum relevance is adequate for different kinds of data.

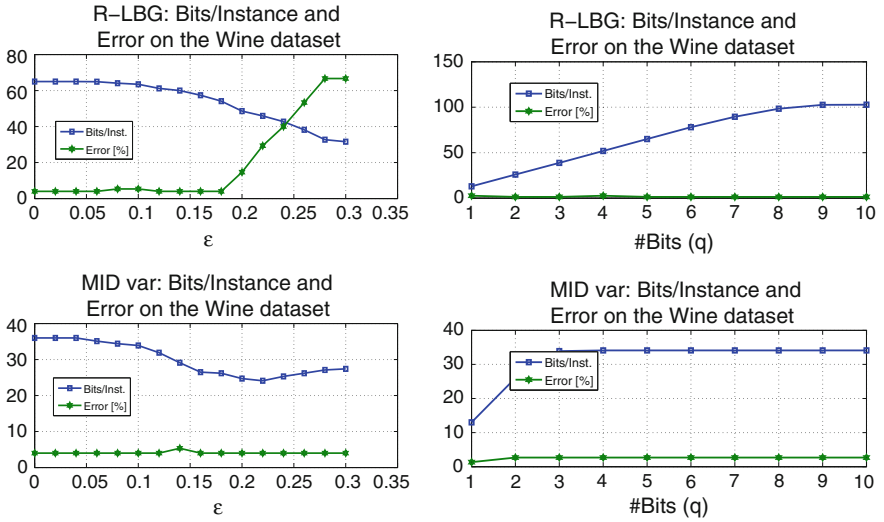


Fig. 4 Number of bits/instance and the test set error rate (%) for a 10-fold CV of the NB classifier on data discretized by R-LBG and MID variable on the Wine dataset for R-LBG (*top*) and MID variable (*bottom*). *Left*: $q = 5$ bits and ϵ in the real interval from 0 to 0.3. *Right*: $q \in \{1, \ldots, 10\}$ and $\epsilon = 0.05$

4.2 Comparison with Existing Methods

We now compare our methods with some well-known existing approaches for FD. First, in Table 3 we assess the behavior of R-LBG in unsupervised mode, comparing it with five existing unsupervised FD methods (see Sect. 2.3). We evaluate the average number of discretization intervals and the 10-fold CV error (%), for the linear SVM classifier, using each FD method with $q = 3$ bits. R-LBG uses $@rel = NVAR$ and $\epsilon = 0.05$. For all datasets, the use of a FD technique leads to equal or better results as compared to the use of the original features, with the exception of the Wine dataset. In these unsupervised settings, the R-LBG algorithm seems to be adequate for the higher-dimensional datasets since it computes fewer discretization intervals, as compared to the other techniques. R-LBG does not show improvement over the other techniques on lower and medium-dimensional data.

We now assess the behavior of our methods for supervised FD. The MID fixed, MID variable, and R-LBG methods, with $q = 3$ bits and $\epsilon = 0.1$, are compared against the four supervised FD techniques described in Sect. 2.4. R-LBG uses MI_s as the relevance measure. Table 4 reports the average number of bits per instance (first row for each dataset) and the test error (%) for a 10-fold CV of the linear SVM classifier. In Table 4, the use of a FD technique improves on the test set error rate, as compared to the use of the original features, in the majority of the datasets. The CAIM and CACC algorithms are not suitable for the higher-dimensional datasets, since they both take a prohibitive running time (several hours) as compared to other

Table 3 For each dataset, the first row presents the average total number of bits per instance and the second row has the test set error rate (%), using a 10-fold CV for the linear SVM classifier (the best results are in bold face)

		Existing unsupervised FD methods					Proposed
Dataset	No FD	EIB	EFB	PkID	U-LBG1	U-LBG2	R-LBG
Car		18.0	18.0	**12.0**	**12.0**	18.0	18.0
	21.2	19.1	**17.4**	18.3	21.7	19.1	19.1
Bupa		18.0	18.0	29.0	18.0	18.0	**15.2**
	42.0	40.0	27.3	**26.9**	41.1	41.1	35.6
Heart		39.0	39.0	33.0	**26.0**	39.0	36.7
	16.7	16.7	**15.9**	16.3	16.3	17.0	16.3
Wine		39.0	39.0	52.0	**22.0**	39.0	38.0
	1.1	2.8	2.8	2.2	6.0	2.8	2.8
Zoo		48.0	48.0	**18.0**	**18.0**	48.0	48.0
	5.8	5.8	**4.8**	5.8	5.8	5.8	5.8
Hepatitis		57.0	57.0	46.0	**28.8**	57.0	56.9
	21.9	21.3	**16.7**	21.2	18.0	20.7	20.1
Ionosphere		99.0	99.0	145.0	**44.7**	99.0	99.0
	12.9	**10.9**	14.8	17.1	11.7	12.9	12.9
SpamBase		162.0	162.0	324.0	**55.8**	162.0	60.0
	10.0	20.5	**6.4**	6.8	19.9	9.3	18.3
Sonar		180.0	180.0	240.0	**60.0**	180.0	175.7
	22.6	18.8	18.8	**17.4**	27.9	23.1	22.6
Colon		6,000.0	6,000.0	6,000.0	6,000.0	6,000.0	**5,994.4**
	21.4	18.3	**15.0**	**15.0**	18.3	18.3	18.3
AR10P		7,200.0	7,200.0	9,270.0	7,200.0	7,200.0	**7,192.2**
	1.0	**0.0**	5.5	5.0	**0.0**	**0.0**	**0.0**
PIE10P		7,260.0	7,260.0	9,680.0	7,260.0	7,260.0	**7,242.6**
	0.5	**0.0**	**0.0**	0.5	**0.0**	**0.0**	**0.0**
Leukemia		21,387.0	21,387.0	21,387.0	21,387.0	21,387.0	**21,087.3**
	1.7	2.9	**1.7**	**1.7**	2.9	2.9	2.9
Dexter		21,932.4	21,932.4	12,455.7	**7,310.8**	21,932.4	8,083.2
	7.7	10.0	**7.0**	7.7	11.0	8.7	8.3

We have used $q = 3$ bit/feature, $\Delta = 0.05\,\text{range}(X_i)$ for U-LBG1, $@rel = NVAR$, and $\epsilon = 0.05$ on R-LBG

approaches (symbol N/A in Table 4). On the Wine and Colon datasets, the use of FD techniques do not show improvement, as compared to the use of the original features. Regarding the test set error, one of our approaches usually attains the best result, except in the case of the SpamBase dataset. Within our approaches, the MID fixed and MID variable methods attain the best results, which suggests: (1) the adequacy of MI between features and class labels for FD purposes; (2) that our incremental FD methods are adequate for different types of data; (3) that a variable number of bits per feature is adequate, regarding both complexity and generalization error.

Table 4 For each dataset, the first row presents the average total number of bits per instance and the second row has the test set error rate (%), using a 10-fold CV for the linear SVM classifier (the best results are in bold face)

		Existing supervised FD methods				Proposed methods		
Dataset	No FD	IEM	IEMV	CAIM	CACC	R-LBG	MID fixed	MID variable
Car		9.8	**9.1**	12.0	12.0	15.0	18.0	14.0
	21.4	14.1	15.1	18.8	18.8	27.7	**13.9**	**13.9**
Bupa		33.2	29.6	18.1	18.1	**18.0**	**18.0**	**18.0**
	41.7	43.2	39.1	32.7	32.7	42.0	**31.5**	**31.5**
Heart		26.5	**26.0**	29.0	29.0	31.8	39.0	26.8
	16.7	15.6	14.8	16.3	16.3	23.0	**14.1**	**14.1**
Wine		**20.0**	21.2	39.1	39.1	27.5	39.0	22.8
	1.1	1.7	1.7	3.3	3.3	5.0	1.7	2.2
Zoo		**17.0**	**17.0**	18.0	18.0	21.2	48.0	20.2
	8.0	**6.0**	**6.0**	7.0	7.0	8.0	**6.0**	**6.0**
Hepatitis		45.2	**42.8**	43.6	43.6	53.0	57.0	50.6
	19.2	18.0	22.0	19.2	19.2	23.1	**17.8**	20.5
Ionosphere		84.2	85.0	96.5	96.5	85.6	99.0	**74.5**
	11.9	12.3	10.8	11.4	11.4	12.0	6.6	**5.4**
SpamBase		**87.7**	88.7	112.6	112.6	161.1	162.0	138.6
	10.1	6.5	**6.4**	6.5	6.5	21.5	6.6	6.6
Sonar		304.9	280.2	221.1	221.1	177.5	180.0	**165.0**
	23.5	19.7	22.1	19.6	19.6	34.6	20.2	**17.9**
Colon		11,329.0	11,089.4	N/A	N/A	5,758.1	6,000.0	**5,117.3**
	14.8	18.1	19.8	N/A	N/A	22.6	17.9	16.4
AR10P		12,942.9	7,155.3	N/A	N/A	7,144.2	7,200.0	**7,065.5**
	1.0	2.0	6.0	N/A	N/A	**1.0**	3.5	3.5
PIE10P		9,090.2	**5,223.5**	N/A	N/A	7,071.6	7,260.0	7,152.4
	0.3	**0.0**	**0.0**	N/A	N/A	0.3	**0.0**	**0.0**
Leukemia		38,733.1	37,196.5	N/A	N/A	20,299.9	21,387.0	**18,254.9**
	1.3	**1.3**	**1.3**	N/A	N/A	**1.3**	**1.3**	**1.3**
Dexter		12,559.6	**12,550.4**	N/A	N/A	21,915.1	21,930.0	21,898.6
	7.3	7.3	7.0	N/A	N/A	29.7	**5.0**	5.7

We have used $q = 3$ bit/feature, $@rel = MI_s$, and $\epsilon = 0.1$ for both R-LBG and MID variable

5 Conclusions

FD is a useful pre-processing step for many machine learning and data mining tasks, leading to compact representations of the data improving on the generalization error. Even in the cases that it is not required, it may help improving the performance of machine learning and data mining tasks. In this paper, we have proposed two FD techniques. The first one is based on the unsupervised Linde-Buzo-Gray algorithm with a relevance criterion. This technique works in unsupervised, supervised,

or semi-supervised problems. The second technique is supervised and is based on mutual information maximization between the discrete feature and the class label. It uses a recursive approach that finds the optimal cut points in the mutual information sense, discretizing with a fixed or variable number of bits per feature. The experimental evaluation of these techniques has shown that both techniques improve on the results of existing FD approaches for supervised learning tasks. The first technique has obtained similar results, when compared to its unsupervised counterparts, being more adequate for the increasingly common high-dimensional datasets. For the supervised FD tests, the second technique has proved to be more effective regarding the number of discretization intervals and the generalization error. Both techniques scale well for high-dimensional datasets and multi-class problems. As future work, we will fine tune the use of Renyi's mutual information in the supervised discretization process. Another issue that will be addressed is the research for different relevance functions to guide unsupervised and supervised discretization.

References

1. Battiti, R.: Using mutual information for selecting features in supervised neural net learning. IEEE Trans. Neural Networks **5**, 537–550 (1994)
2. Brown, G., Pocock, A., Zhao, M., Luján, M.: Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. J. Mach. Learn. Res. **13**, 27–66 (2012)
3. Chiu, D., Wong, A., Cheung, B.: Information discovery through hierarchical maximum entropy discretization and synthesis. In: Proceedings of the Knowledge Discovery in Databases, pp. 125–140 (1991)
4. Cover, T., Thomas, J.: Elements of Information Theory. Wiley, Hoboken (1991)
5. Dougherty, J., Kohavi, R., Sahami, M.: Supervised and unsupervised discretization of continuous features. In: International Conference on Machine Learning (ICML), pp. 194–202 (1995)
6. Fayyad, U., Irani, K.: Multi-interval discretization of continuous-valued attributes for classification learning. In: Proceedings of International Joint Conference on Artifficial Intelligence (IJCAI), pp. 1022–1027 (1993)
7. Ferreira, A., Figueiredo, M.: An unsupervised approach to feature discretization and selection. Pattern Recog. **45**, 3048–3060 (2012)
8. Frank, A., Asuncion, A.: UCI machine learning repository, available at http://archive.ics.uci.edu/ml (2010)
9. Garcia, S., Luengo, J., Saez, J., Lopez, V., Herrera, F.: A survey of discretization techniques: taxonomy and empirical analysis in supervised learning. IEEE Trans. Knowl. Data Eng. **25**(4), 734–750 (2013)
10. Hellman, M.: Probability of error, equivocation, and the Chernoff bound. IEEE Trans. Inf. Theory **16**(4), 368–372 (1970)
11. Jin, R., Breitbart, Y., Muoh, C.: Data discretization unification. Knowl. Inf. Syst. **19**(1), 1–29 (2009)
12. Kononenko, I.: On biases in estimating multi-valued attributes. In: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), pp. 1034–1040 (1995)
13. Kotsiantis, S., Kanellopoulos, D.: Discretization techniques: a recent survey. GESTS Int. Trans. Comput. Sci. Eng. **32**(1), 47–58 (2006)
14. Kurgan, L., Cios, K.: CAIM discretization algorithm. IEEE Trans. Knowl. Data Eng. **16**(2), 145–153 (2004)
15. Linde, Y., Buzo, A., Gray, R.: An algorithm for vector quantizer design. IEEE Trans. Commun. **28**, 84–94 (1980)

16. Liu, H., Hussain, F., Tan, C., Dash, M.: Discretization: an enabling technique. Data Min. Knowl. Disc. **6**(4), 393–423 (2002)
17. Principe, J.: Information Theoretic Learning: Renyi's Entropy and Kernel Perspectives, 1st edn. Springer, Heidelberg (2010)
18. Santhi, N., Vardy, A.: On an improvement over Rényi's equivocation bound. In: 44-th Annual Allerton Conference on Communication, Control, and Computing (2006)
19. Tsai, C.-J., Lee, C.-I., Yang, W.-P.: A discretization algorithm based on class-attribute contingency coefficient. Inf. Sci. **178**, 714–731 (2008)
20. Witten, I., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Elsevier, Morgan Kauffmann, Burlington (2005)
21. Yang, Y., Webb, G.: Proportional k-interval discretization for naïve-Bayes classifiers. In: 12th European Conference on Machine Learning, (ECML), pp. 564–575 (2001)

Multiclass Semi-supervised Learning on Graphs Using Ginzburg-Landau Functional Minimization

Cristina Garcia-Cardona, Arjuna Flenner and Allon G. Percus

Abstract We present a graph-based variational algorithm for classification of high-dimensional data, generalizing the binary diffuse interface model to the case of multiple classes. Motivated by total variation techniques, the method involves minimizing an energy functional made up of three terms. The first two terms promote a stepwise continuous classification function with sharp transitions between classes, while preserving symmetry among the class labels. The third term is a data fidelity term, allowing us to incorporate prior information into the model in a semi-supervised framework. The performance of the algorithm on synthetic data, as well as on the COIL and MNIST benchmark datasets, is competitive with state-of-the-art graph-based multiclass segmentation methods.

Keywords Diffuse interfaces · Learning on graphs · Semi-supervised methods

1 Introduction

Many tasks in pattern recognition and machine learning rely on the ability to quantify local similarities in data, and to infer meaningful global structure from such local characteristics [1]. In the classification framework, the desired global structure is a descriptive partition of the data into categories or classes. Many studies have been devoted to the binary classification problems. The multiple-class case, where data are partitioned into more than two clusters, is more challenging. One approach is to treat the problem as a series of binary classification problems [2]. In this paper, we

C. Garcia-Cardona (✉) · A.G. Percus
Institute of Mathematical Sciences, Claremont Graduate University, Claremont , CA, 91711, USA
e-mail: cristina.cgarcia@gmail.com

A.G. Percus
e-mail: allon.percus@cgu.edu

A. Flenner
Physics and Computational Sciences, Naval Air Warfare Center, China Lake, CA 93555, USA

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_8

develop an alternative method, involving a multiple-class extension of the diffuse interface model introduced in [3].

The diffuse interface model by Bertozzi and Flenner combines methods for diffusion on graphs with efficient partial differential equation techniques to solve binary segmentation problems. As with other methods inspired by physical phenomena [4–6], it requires the minimization of an energy expression, specifically the Ginzburg-Landau (GL) energy functional. The formulation generalizes the GL functional to the case of functions defined on graphs, and its minimization is related to the minimization of weighted graph cuts [3]. In this sense, it parallels other techniques based on inference on graphs via diffusion operators or function estimation [1, 7–13].

Multiclass segmentation methods that cast the problem as a series of binary classification problems use a number of different strategies: (i) deal directly with some binary coding or indicator for the labels [10, 14], (ii) build a hierarchy or combination of classifiers based on the one-vs-all approach or on class rankings [15, 16] or (iii) apply a recursive partitioning scheme consisting of successively subdividing clusters, until the desired number of classes is reached [12, 13]. While there are advantages to these approaches, such as possible robustness to mislabeled data, there can be a considerable number of classifiers to compute, and performance is affected by the number of classes to partition.

In contrast, we propose an extension of the diffuse interface model that obtains a simultaneous segmentation into multiple classes. The multiclass extension is built by modifying the GL energy functional to remove the prejudicial effect that the order of the labelings, given by integer values, has in the smoothing term of the original binary diffuse interface model. A new term that promotes homogenization in a multiclass setup is introduced. The expression penalizes data points that are located close in the graph but are not assigned to the same class. This penalty is applied *independently* of how different the integer values are, representing the class labels. In this way, the characteristics of the multiclass classification task are incorporated directly into the energy functional, with a measure of smoothness independent of label order, allowing us to obtain high-quality results. Alternative multiclass methods minimize a Kullback-Leibler divergence function [17] or expressions involving the discrete Laplace operator on graphs [10, 18].

This paper is organized as follows. Section 2 reviews the diffuse interface model for binary classification, and describes its application to semi-supervised learning. Section 3 discusses our proposed multiclass extension and the corresponding computational algorithm. Section 4 presents results obtained with our method. Finally, Sect. 5 draws conclusions and delineates future work.

2 Data Segmentation with the Ginzburg-Landau Model

The diffuse interface model [3] is based on a continuous approach, using the Ginzburg-Landau (GL) energy functional to measure the quality of data segmentation. A good segmentation is characterized by a state with small energy. Let $u(\boldsymbol{x})$

be a scalar field defined over a space of arbitrary dimensionality, and representing the state of the system. The GL energy is written as the functional

$$\mathrm{GL}(u) = \frac{\epsilon}{2} \int |\nabla u|^2 \, d\boldsymbol{x} + \frac{1}{\epsilon} \int \Phi(u) \, d\boldsymbol{x}, \tag{1}$$

with ∇ denoting the spatial gradient operator, $\epsilon > 0$ a real constant value, and Φ a double well potential with minima at ± 1:

$$\Phi(u) = \frac{1}{4} \left(u^2 - 1\right)^2. \tag{2}$$

Segmentation requires minimizing the GL functional. The norm of the gradient is a smoothing term that penalizes variations in the field u. The potential term, on the other hand, compels u to adopt the discrete labels of $+1$ or -1, clustering the state of the system around two classes. Jointly minimizing these two terms pushes the system domain towards homogeneous regions with values close to the minima of the double well potential, making the model appropriate for binary segmentation.

The smoothing term and potential term are in conflict at the interface between the two regions, with the first term favoring a gradual transition, and the second term penalizing deviations from the discrete labels. A compromise between these conflicting goals is established via the constant ϵ. A small value of ϵ denotes a small length transition and a sharper interface, while a large ϵ weights the gradient norm more, leading to a slower transition. The result is a diffuse interface between regions, with sharpness regulated by ϵ.

It can be shown that in the limit $\epsilon \to 0$ this function approximates the total variation (TV) formulation in the sense of functional (Γ) convergence [19], producing piecewise constant solutions but with greater computational efficiency than conventional TV minimization methods. Thus, the diffuse interface model provides a framework to compute piecewise constant functions with diffuse transitions, approaching the ideal of the TV formulation, but with the advantage that the smooth energy functional is more tractable numerically and can be minimized by simple numerical methods such as gradient descent.

The GL energy has been used to approximate the TV norm for image segmentation [3] and image inpainting [4, 20]. Furthermore, a calculus on graphs equivalent to TV has been introduced in [12, 21].

2.1 Application of Diffuse Interface Models to Graphs

An undirected, weighted neighborhood graph is used to represent the local relationships in the data set. This is a common technique to segment classes that are not linearly separable. In the N-neighborhood graph model, each vertex $v_i \in V$ of the graph corresponds to a data point with feature vector $\boldsymbol{x}_i$, while the weight w_{ij}

is a measure of similarity between v_i and v_j. Moreover, it satisfies the symmetry property $w_{ij} = w_{ji}$. The neighborhood is defined as the set of N closest points in the feature space. Accordingly, edges exist between each vertex and the vertices of its N-nearest neighbors. Following the approach of [3], we calculate weights using the local scaling of Zelnik-Manor and Perona [22],

$$w_{ij} = \exp\left(-\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{\tau(\boldsymbol{x}_i)\ \tau(\boldsymbol{x}_j)}\right). \tag{3}$$

Here, $\tau(\boldsymbol{x}_i) = ||\boldsymbol{x}_i - \boldsymbol{x}_i^M||$ defines a local value for each $\boldsymbol{x}_i$, where $\boldsymbol{x}_i^M$ is the position of the Mth closest data point to $\boldsymbol{x}_i$, and M is a global parameter.

It is convenient to express calculations on graphs via the graph Laplacian matrix, denoted by $\mathbf{L}$. The procedure we use to build the graph Laplacian is as follows.

1. Compute the similarity matrix $\mathbf{W}$ with components w_{ij} defined in (3). As the neighborhood relationship is not symmetric, the resulting matrix $\mathbf{W}$ is also not symmetric. Make it a symmetric matrix by connecting vertices v_i and v_j if v_i is among the N-nearest neighbors of v_j or if v_j is among the N-nearest neighbors of v_i [23].
2. Define $\mathbf{D}$ as a diagonal matrix whose ith diagonal element represents the degree of the vertex v_i, evaluated as

$$d_i = \sum_j w_{ij}. \tag{4}$$

3. Calculate the graph Laplacian: $\mathbf{L} = \mathbf{D} - \mathbf{W}$.

Generally, the graph Laplacian is normalized to guarantee spectral convergence in the limit of large sample size [23]. The symmetric normalized graph Laplacian $\mathbf{L}_s$ is defined as

$$\mathbf{L}_s = \mathbf{D}^{-1/2}\,\mathbf{L}\,\mathbf{D}^{-1/2} = \mathbf{I} - \mathbf{D}^{-1/2}\,\mathbf{W}\,\mathbf{D}^{-1/2}. \tag{5}$$

Data segmentation can now be carried out through a graph-based formulation of the GL energy. To implement this task, a fidelity term is added to the functional as initially suggested in [24]. This enables the specification of a priori information in the system, for example the known labels of certain points in the data set. This kind of setup is called semi-supervised learning (SSL). The discrete GL energy for SSL on graphs can be written as [3]:

$$\mathrm{GL}_{\mathrm{SSL}}(\boldsymbol{u}) = \frac{\epsilon}{2}\langle \boldsymbol{u}, \mathbf{L}_s \boldsymbol{u}\rangle + \frac{1}{\epsilon}\sum_{v_i \in V} \Phi(u(v_i)) + \sum_{v_i \in V} \frac{\mu(v_i)}{2}\left(u(v_i) - \hat{u}(v_i)\right)^2 \quad (6)$$

$$= \frac{\epsilon}{4}\sum_{v_i, v_j \in V} w_{ij}\left(\frac{u(v_i)}{\sqrt{d_i}} - \frac{u(v_j)}{\sqrt{d_j}}\right)^2 + \frac{1}{\epsilon}\sum_{v_i \in V} \Phi(u(v_i)) + \sum_{v_i \in V} \frac{\mu(v_i)}{2}\left(u(v_i) - \hat{u}(v_i)\right)^2. \quad (7)$$

In the discrete formulation, $\boldsymbol{u}$ is a vector whose component $u(v_i)$ represents the state of the vertex v_i, $\epsilon > 0$ is a real constant characterizing the smoothness of the transition between classes, and $\mu(v_i)$ is a fidelity weight taking value $\mu > 0$ if the label $\hat{u}(v_i)$ (i.e. class) of the data point associated with vertex v_i is known beforehand, or $\mu(v_i) = 0$ if it is not known (semi-supervised).

Minimizing the functional simulates a diffusion process on the graph. The information of the few labels known is propagated through the discrete structure by means of the smoothing term, while the potential term clusters the vertices around the states ± 1 and the fidelity term enforces the known labels. The energy minimization process itself attempts to reduce the interface regions. Note that in the absence of the fidelity term, the process could lead to a trivial steady-state solution of the diffusion equation, with all data points assigned the same label.

The final state $u(v_i)$ of each vertex is obtained by thresholding, and the resulting homogeneous regions with labels of $+1$ and -1 constitute the two-class data segmentation.

3 Multiclass Extension

The double-well potential in the diffuse interface model for SSL drives the state of the system towards two definite labels. Multiple-class segmentation requires a more general potential function $\Phi_M(u)$ that allows clusters around more than two labels. For this purpose, we use the periodic-well potential suggested by Li and Kim [6],

$$\Phi_M(u) = \frac{1}{2}\{u\}^2\left(\{u\} - 1\right)^2, \quad (8)$$

where $\{u\}$ denotes the fractional part of u,

$$\{u\} = u - \lfloor u \rfloor, \quad (9)$$

and $\lfloor u \rfloor$ is the largest integer not greater than u.

This periodic potential well promotes a multiclass solution, but the graph Laplacian term in Eq. (6) also requires modification for effective calculations due to the fixed ordering of class labels in the multiple class setting. The graph Laplacian term

Fig. 1 Three-class segmentation. *Black* Class 0. *Gray* Class 1. *White* Class 2

penalizes large changes in the spatial distribution of the system state more than smaller gradual changes. In a multiclass framework, this implies that the penalty for two spatially contiguous classes with different labels may vary according to the (arbitrary) ordering of the labels.

This phenomenon is shown in Fig. 1. Suppose that the goal is to segment the image into three classes: class 0 composed by the black region, class 1 composed by the gray region and class 2 composed by the white region. It is clear that the horizontal interfaces comprise a jump of size 1 (analogous to a two class segmentation) while the vertical interface implies a jump of size 2. Accordingly, the smoothing term will assign a higher cost to the vertical interface, even though from the point of view of the classification, there is no specific reason for this. In this example, the problem cannot be solved with a different label assignment. There will always be an interface with higher costs than others independent of the integer values used.

Thus, the multiclass approach breaks the symmetry among classes, influencing the diffuse interface evolution in an undesirable manner. Eliminating this inconvenience requires restoring the symmetry, so that the difference between two classes is always the same, regardless of their labels. This objective is achieved by introducing a new class difference measure.

3.1 Generalized Difference Function

The final class labels are determined by thresholding each vertex $u(v_i)$, with the label y_i set to the nearest integer:

$$y_i = \left\lfloor u(v_i) + \frac{1}{2} \right\rfloor . \tag{10}$$

The boundaries between classes then occur at half-integer values corresponding to the unstable equilibrium states of the potential well. Define the function $\hat{r}(x)$ to represent the distance to the nearest half-integer:

$$\hat{r}(x) = \left| \frac{1}{2} - \{x\} \right| . \tag{11}$$

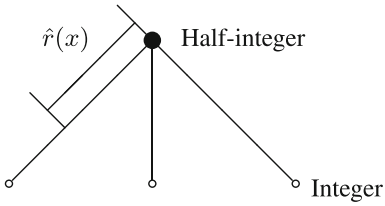


Fig. 2 Schematic interpretation of generalized difference: $\hat{r}(x)$ measures distance to nearest half-integer, and ρ is a tree distance measure

A schematic of $\hat{r}(x)$ is depicted in Fig. 2. The $\hat{r}(x)$ function is used to define a generalized difference function between classes that restores symmetry in the energy functional. Define the generalized difference function ρ as:

$$\rho(u(v_i), u(v_j)) = \begin{cases} \hat{r}(u(v_i)) + \hat{r}(u(v_j)) & y_i \neq y_j \\ \left|\hat{r}(u(v_i)) - \hat{r}(u(v_j))\right| & y_i = y_j \end{cases} \tag{12}$$

Thus, if the vertices are in different classes, the difference $\hat{r}(x)$ between each state's value and the nearest half-integer is added, whereas if they are in the same class, these differences are subtracted. The function $\rho(x, y)$ corresponds to the tree distance (see Fig. 2). Strictly speaking, ρ is not a metric since it does not satisfy $\rho(x, y) = 0 \Rightarrow x = y$. Nevertheless, the cost of interfaces between classes becomes the same regardless of class labeling when this generalized distance function is implemented.

The GL energy functional for SSL, using the new generalized difference function ρ and the periodic potential, is expressed as

$$\begin{aligned} \mathrm{MGL}_{\mathrm{SSL}}(\boldsymbol{u}) \; &= \frac{\epsilon}{2} \sum_{v_i \in V} \sum_{v_j \in V} \frac{w_{ij}}{\sqrt{d_i d_j}} \left[\rho(u(v_i), u(v_j))\right]^2 \\ &+ \frac{1}{2\epsilon} \sum_{v_i \in V} \{u(v_i)\}^2 \left(\{u(v_i)\} - 1\right)^2 \\ &+ \sum_{v_i \in V} \frac{\mu(v_i)}{2} \left(u(v_i) - \hat{u}(v_i)\right)^2. \end{aligned} \tag{13}$$

Note that the smoothing term in this functional is composed of an operator that is not just a generalization of the normalized symmetric Laplacian $\mathbf{L}_s$. The new smoothing operation, written in terms of the generalized distance function ρ, constitutes a non-linear operator that is a symmetrization of a different normalized Laplacian, the random walk Laplacian $\mathbf{L}_w = \mathbf{D}^{-1}\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{W}$ [23]. The reason is as follows. The Laplacian $\mathbf{L}$ satisfies

$$(\mathbf{L}\boldsymbol{u})_i = \sum_j w_{ij} \left(u_i - u_j\right)$$

and $\mathbf{L}_{\boldsymbol{w}}$ satisfies

$$(\mathbf{L}_{\boldsymbol{w}}\boldsymbol{u})_i = \sum_j \frac{w_{ij}}{d_i}\left(u_i - u_j\right).$$

Now replace w_{ij}/d_i in the latter expression with the symmetric form $w_{ij}/\sqrt{d_i d_j}$. This is equivalent to constructing a reweighted graph with weights $\widehat{w}_{ij}$ given by:

$$\widehat{w}_{ij} = \frac{w_{ij}}{\sqrt{d_i d_j}}.$$

The corresponding reweighted Laplacian $\widehat{\mathbf{L}}$ satisfies:

$$(\widehat{\mathbf{L}}\boldsymbol{u})_i = \sum_j \widehat{w}_{ij}\left(u_i - u_j\right) = \sum_j \frac{w_{ij}}{\sqrt{d_i d_j}}\left(u_i - u_j\right), \tag{14}$$

and

$$\langle \boldsymbol{u}, \widehat{\mathbf{L}}\boldsymbol{u}\rangle = \frac{1}{2}\sum_{i,j} \frac{w_{ij}}{\sqrt{d_i d_j}}\left(u_i - u_j\right)^2. \tag{15}$$

While $\widehat{\mathbf{L}} = \widehat{\mathbf{D}} - \widehat{\mathbf{W}}$ is not a standard normalized Laplacian, it does have the desirable properties of stability and consistency with increasing sample size of the data set, and of satisfying the conditions for Γ-convergence to TV in the $\epsilon \to 0$ limit [25]. It also generalizes to the tree distance more easily than does $\mathbf{L}_{\boldsymbol{s}}$. Replacing the difference $\left(u_i - u_j\right)^2$ with the generalized difference $\left[\rho(u_i, u_j)\right]^2$ then gives the new smoothing multiclass term of Eq. (13). Empirically, this new term seems to perform well even though the normalization procedure differs from the binary case.

By implementing the generalized difference function on a tree, the cost of interfaces between classes becomes the same regardless of class labeling.

3.2 Computational Algorithm

The GL energy functional given by (13) may be minimized iteratively, using gradient descent:

$$u_i^{n+1} = u_i^n - dt\left[\frac{\delta \mathrm{MGL}_{\mathrm{SSL}}}{\delta u_i}\right], \tag{16}$$

where u_i is a shorthand for $u(v_i)$, dt represents the time step and the gradient direction is given by:

$$\frac{\delta \mathrm{MGL}_{\mathrm{SSL}}}{\delta u_i} = \epsilon\, \hat{R}(u_i^n) + \frac{1}{\epsilon}\Phi'_M(u_i^n) + \mu_i\left(u_i^n - \hat{u}_i\right) \tag{17}$$

$$\hat{R}(u_i^n) = \sum_j \frac{w_{ij}}{\sqrt{d_i d_j}} \left[\hat{r}(u_i^n) \pm \hat{r}(u_j^n) \right] \hat{r}'(u_i^n) \tag{18}$$

$$\Phi'_M(u_i^n) = 2\,\{u_i^n\}^3 - 3\,\{u_i^n\}^2 + \{u_i^n\} \tag{19}$$

The gradient of the generalized difference function ρ is not defined at half integer values. Hence, we modify the method using a greedy strategy: after detecting that a vertex changes class, the new class that minimizes the smoothing term is selected, and the fractional part of the state computed by the gradient descent update is preserved. Consequently, the new state of vertex i is the result of gradient descent, but if this causes a change in class, then a new state is determined.

Algorithm 1: Calculate $\boldsymbol{u}$.

Require: $\epsilon, dt, N_D, n_{\max}, K$
Ensure: out $= \boldsymbol{u}^{\text{end}}$
  for $i = 1 \to N_D$ **do**
    ${u_i}^0 \leftarrow rand((0, K)) - \frac{1}{2}$. If $\mu_i > 0$, ${u_i}^0 \leftarrow \hat{u}_i$
  end for
  for $n = 1 \to n_{\max}$ **do**
    for $i = 1 \to N_D$ **do**
      $u_i^{n+1} \leftarrow u_i^n - dt \left(\epsilon\, \hat{R}(u_i^n) + \frac{1}{\epsilon}\, \Phi'_M(u_i^n) + \mu_i \left(u_i^n - \hat{u}_i \right) \right)$
      if Label$(u_i^{n+1}) \neq$ Label(u_i^n) **then**
        $\hat{k} = \arg\min_{0 \le k < K} \sum_j \frac{w_{ij}}{\sqrt{d_i d_j}} \left[\rho(k + \{u_i^{n+1}\}, u_j^{n+1}) \right]^2$
        $u_i^{n+1} \leftarrow \hat{k} + \{u_i^{n+1}\}$
      end if
    end for
  end for

Specifically, let k represent an integer in the range of the problem, i.e. $k \in [0, K-1]$, where K is the number of classes in the problem. Given the fractional part $\{u\}$ resulting from the gradient descent update, find the integer k that minimizes $\sum_j \frac{w_{ij}}{\sqrt{d_i d_j}} \left[\rho(k + \{u_i\}, u_j) \right]^2$, the smoothing term in the energy functional, and use $k + \{u_i\}$ as the new vertex state. A summary of the procedure is shown in Algorithm 1 with N_D representing the number of points in the data set and $n_{\max}$ denoting the maximum number of iterations.

4 Results

The performance of the multiclass diffuse interface model is evaluated using a number of data sets from the literature, with differing characteristics. Data and image segmentation problems are considered on synthetic and real data sets.

4.1 Synthetic Data

4.1.1 Three Moons

A synthetic three-class segmentation problem is constructed following an analogous procedure to the one used in [11] for "two moon" binary classification. Three half circles ("three moons") are generated in $\mathbb{R}^2$. The two top circles have radius 1 and are centered at $(0, 0)$ and $(3, 0)$. The bottom half circle has radius 1.5 and is centered at $(1.5, 0.4)$. 1,500 data points (500 from each of these half circles) are sampled and embedded in $\mathbb{R}^{100}$. The embedding is completed by adding Gaussian noise with $\sigma^2 = 0.02$ to *each* of the 100 components for each data point. The dimensionality of the data set, together with the noise, make this a nontrivial problem.

The symmetric normalized graph Laplacian is computed for a local scaling graph using $N = 10$ nearest neighbors and local scaling based on the $M = 10$th closest point. The fidelity term is constructed by labeling 25 points per class, 75 points in total, corresponding to only 5 % of the points in the data set. The multiclass GL method was further refined by geometrically decreasing ϵ over the course of the minimization process, from ϵ_0 to ϵ_f by factors of $1 - \Delta_\epsilon$ ($n_{\max}$ iterations per value of ϵ), to allow sharper transitions between states as in [3]. Table 1 specifies the parameters used. Average accuracies and computation times are reported over 100 runs. Results for k-means and spectral clustering (obtained by applying k-means to the first 3 eigenvectors of $\mathbf{L}_s$) are included as reference.

Segmentations obtained for spectral clustering and for multiclass GL with adaptive ϵ methods are shown in Fig. 3. The figure displays the *best* result obtained over 100 runs, corresponding to accuracies of 81.3 % (spectral clustering) and 97.9 % (multiclass GL with adaptive ϵ). The same graph structure is used for the spectral clustering decomposition and the multiclass GL method.

For comparison, we note the results from the literature for the simpler two-moon problem (also $\mathbb{R}^{100}$, $\sigma^2 = 0.02$ noise). The best results reported include: 94 % for p-Laplacian [11], 95.4 % for ratio-minimization relaxed Cheeger cut [12], and 97.7 % for binary GL [3]. While these are not SSL methods, the last of these does involve other prior information in the form of a mass balance constraint. It can be seen that

Table 1 Three-moons results

Method	Parameters	Correct % (stddev %)	Time [s]
k-means	–	72.1 (0.35)	0.66
Spectral clustering	3 eigenvectors	80.0 (0.59)	0.02
Multiclass GL	$\mu = 30, \epsilon = 1, dt = 0.01,$ $n_{\max} = 1,000$	95.1 (2.33)	0.89
Multiclass GL (adaptive ϵ)	$\mu = 30, \epsilon_0 = 2, \epsilon_f = 0.01,$ $\Delta_\epsilon = 0.1, dt = 0.01,$ $n_{\max} = 40$	96.2 (1.59)	1.61

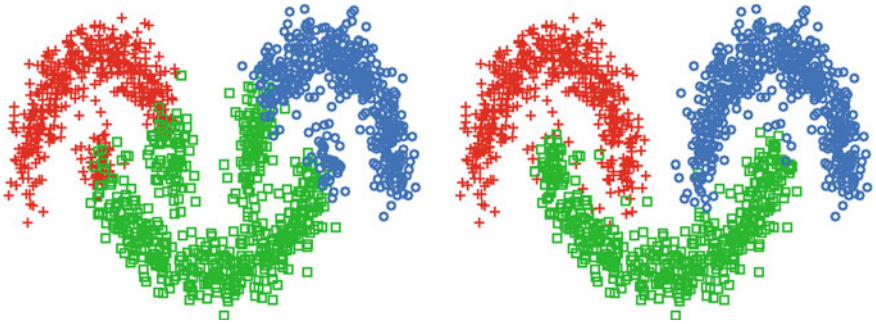


Fig. 3 Three-moons segmentation. *Left* Spectral clustering. *Right* Multiclass GL with adaptive ϵ

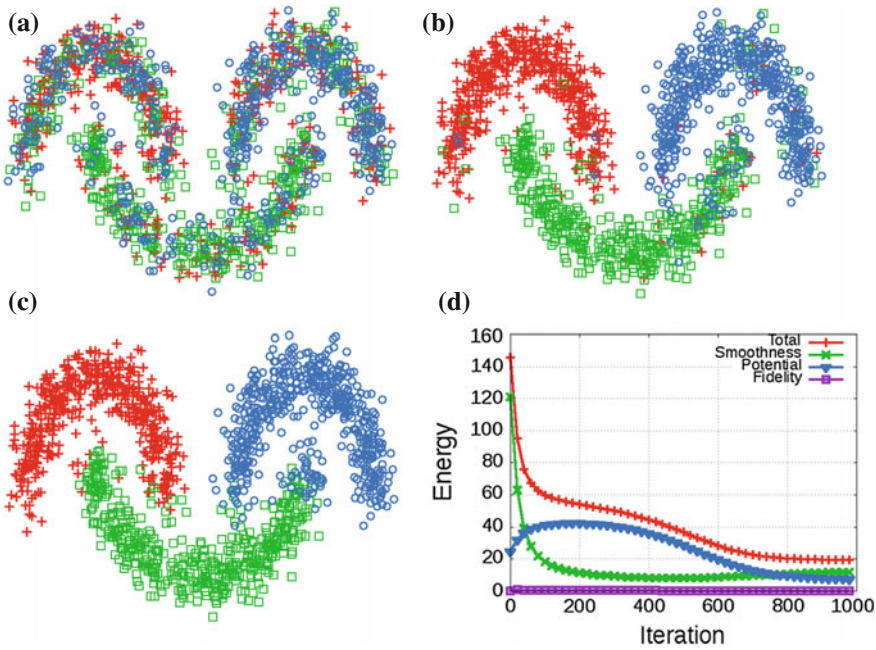


Fig. 4 Evolution of label values in three moons, using multiclass GL (fixed ϵ): $\mathbb{R}^2$ projections at 100, 300 and 1,000 iterations, and energy evolution **a** 100 iterations, **b** 300 iterations, **c** 1,000 iterations, **d** Energy evolution

our procedures produce similarly high-quality results even for the more complex three-class segmentation problem.

It is instructive to observe the evolution of label values in the multiclass method. Figure 4 displays $\mathbb{R}^2$ projections of the results of multiclass GL (with fixed ϵ), at 100, 300 and 1,000 iterations. The system starts from a random configuration. Notice that after 100 iterations, the structure is still fairly inhomogeneous, but small uniform regions begin to form. These correspond to islands around fidelity points and become

seeds for further homogenization. The system progresses fast, and by 300 iterations the configuration is close to the final result: some points are still incorrectly labeled, mostly on the boundaries, but the classes form nearly uniform clusters. By 1,000 iterations the procedure converges to a steady state and a high-quality multiclass segmentation (95 % accuracy) is obtained.

In addition, the energy evolution for one typical run is shown in Fig. 4d for the case with fixed ϵ. The figure includes plots of the total energy (red) as well as the partial contributions of each of the three terms, namely smoothing (green), potential (blue) and fidelity (purple). Observe that at the initial iterations, the principal contribution to the energy comes from the smoothing term, but it has a fast decay due to the homogenization taking place. At the same time, the potential term increases, as ρ pushes the label values toward half-integers. Eventually, the minimization process is driven by the potential term, while small local adjustments are made. The fidelity term is satisfied quickly and has almost negligible influence after the first few iterations. This picture of the "typical" energy evolution can serve as a useful guide in evaluating the performance of the method when no ground truth is available.

4.1.2 Swiss Roll

A synthetic four-class segmentation problem is constructed using the Swiss roll mapping, following the procedure in [26]. The data are created in $\mathbb{R}^2$ by randomly sampling from a Gaussian mixture model of four components with means at (7.5, 7.5), (7.5, 12.5), (12.5, 7.5) and (12.5, 12.5), and all covariances given by the 2×2 identity matrix. 1,600 points are sampled (400 from each of the Gaussians). The data are then converted from 2 to 3 dimensions, with the following Swiss roll mapping: $(x, y) \rightarrow (x \cos(x), y, x \sin(x))$.

As before, we construct the weight matrix for a local scaling graph, with $N = 10$ and scaling based on the $M = 10$th closest neighbor. The fidelity set is formed by labeling 5% of the points selected randomly.

Table 2 gives a description of the parameters used, as well as average results over 100 runs for k-means, spectral clustering and multiclass GL. The *best* results achieved over these 100 runs are shown in Fig. 5. These correspond to accuracies of 50.1 % (spectral clustering) and 96.4 % (multiclass GL). Notice that spectral clustering produces results composed of compact classes, but with a configuration that does

Table 2 Swiss roll results

Method	Parameters	Correct % (stddev %)	Time s
k-means	–	37.9 (0.91)	0.05
Spectral clustering	4 eigenvectors	49.7 (0.96)	0.05
Multiclass GL	$\mu = 50, \epsilon = 1, dt = 0.01$ $n_{\max} = 1{,}000$	91.0 (2.72)	0.75

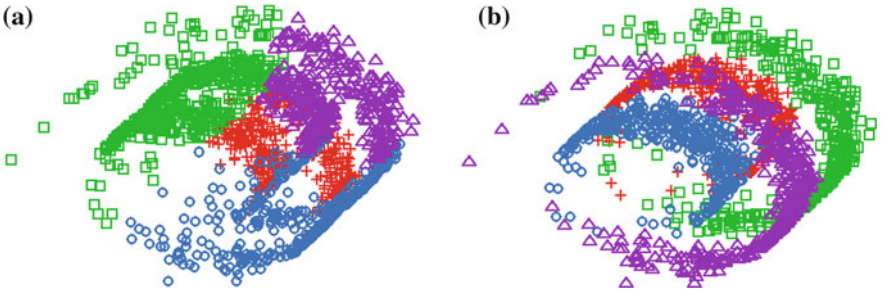


Fig. 5 Swiss roll results **a** Spectral clustering, **b** Multiclass GL

not follow the manifold structure. In contrast, the multiclass GL method is capable of segmenting the manifold structure correctly, achieving higher accuracies.

4.2 Image Segmentation

We apply our algorithm to the color image of cows shown in Fig. 6a. This is a 213×320 color image, to be divided into four classes: sky, grass, black cow and red cow. To construct the weight matrix, we use feature vectors defined as the set of intensity values in the neighborhood of a pixel. The neighborhood is a patch of size 5×5. Red, green and blue channels are appended, resulting in a feature vector of dimension 75. A local scaling graph with $N = 30$ and $M = 30$ is constructed. For the fidelity term, 2.6 % of labeled pixels are used (Fig. 6b).

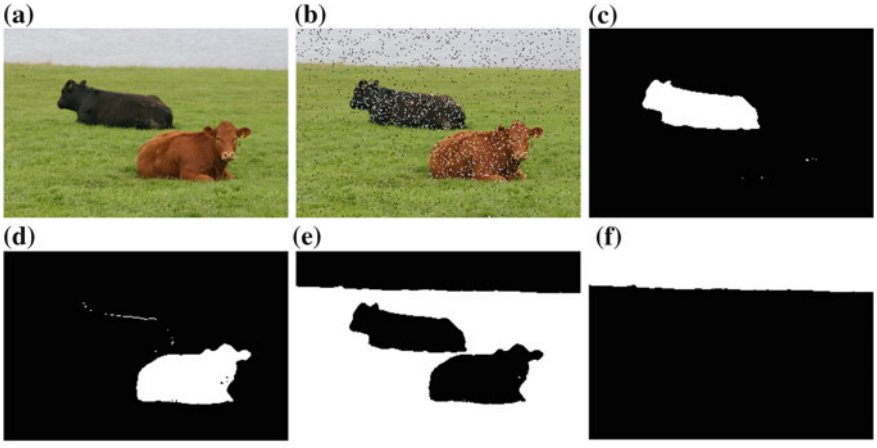


Fig. 6 Color (multi-channel) image. Original image, sampled fidelity and results **a** Original, **b** Sampled, **c** Black cow, **d** Red cow, **e** Grass, **f** Sky

The multiclass GL method used the following parameters: $\mu = 30$, $\epsilon = 1$, $dt = 0.01$ and $n_{\max} = 800$. The average time for segmentation using different fidelity sets was 19.9s. Results are depicted in Fig. 6c–f. Each class image shows in white the pixels identified as belonging to the class, and in black the pixels of the other classes. It can be seen that all the classes are clearly segmented. The few mistakes made are in identifying some borders of the black cow as part of the red cow, and vice-versa.

4.3 Benchmark Sets

4.3.1 COIL-100

The Columbia object image library (COIL-100) is a set of 7,200 color images of 100 different objects taken from different angles (in steps of 5 degrees) at a resolution of 128×128 pixels [27]. This image database has been preprocessed and made available by [28] as a benchmark for SSL algorithms. In summary, the red channel of each image is downsampled to 16×16 pixels by averaging over blocks of 8×8 pixels. Then 24 of the objects are randomly selected and partitioned into six arbitrary classes: 38 images are discarded from each class, leaving 250 per class or 1,500 images in all. The downsampled 16×16 images are further processed to hide the image structure by rescaling, adding noise and masking 15 of the 256 components. The result is a data set of 1,500 data points, of dimension 241.

We build a local scaling graph, with $N = 4$ nearest neighbors and scaling based on the $M = 4$th closest neighbor. The fidelity term is constructed by labeling 10% of the points, selected at random. The multiclass GL method used the following parameters: $\mu = 100$, $\epsilon = 4$, $dt = 0.02$ and $n_{\max} = 1{,}000$. An average accuracy of 93.2 %, with standard deviation of 1.27 %, is obtained over 100 runs, with an average time for segmentation of 0.29s.

For comparison, we note the results reported in [17]: 83.5 % (k-nearest neighbors), 87.8 % (LapRLS), 89.9 % (sGT), 90.9 % (SQ-Loss-I) and 91.1 % (MP). All these are SSL methods (with the exception of k-nearest neighbors which is supervised), using 10 % fidelity just as we do. As can be seen, our results are of greater accuracy.

4.3.2 MNIST Data

The MNIST data set [29] is composed of 70,000 28×28 images of handwritten digits 0 through 9. The task is to classify each of the images into the corresponding digit. Hence, this is a 10-class segmentation problem.

The weight matrix constructed corresponds to a local scaling graph with $N = 8$ nearest neighbors and scaling based on the $M = 8$th closest neighbor. We perform no preprocessing, so the graph directly uses the 28×28 images. This yields a data set of 70,000 points of dimension 784. For the fidelity term, 250 images per class (2,500

images, corresponding to 3.6 % of the data) are chosen randomly. The multiclass GL method used the following parameters: $\mu = 50, \epsilon = 1, dt = 0.01$ and $n_{\max} = 1{,}500$. An average accuracy of 96.9 %, with standard deviation of 0.04 %, is obtained over 50 runs. The average time for segmentation using different fidelity sets was 60.89 s.

Comparative results from other methods reported in the literature include: 87.1 % (p-Laplacian [11]), 87.64 % (multicut normalized 1-cut [13]), 88.2 % (Cheeger cuts [12]), 92.6 % (transductive classification [9]). As with the three-moon problem, some of these are based on unsupervised methods but incorporate enough prior information that they can fairly be compared with SSL methods. Comparative results from *supervised* methods are: 88 % (linear classifiers [29, 30]), 92.3–98.74 % (boosted stumps [29]), 95.0–97.17 % (k-nearest neighbors [29, 30]), 95.3–99.65 % (neural/convolutional nets [29, 30]), 96.4–96.7 % (nonlinear classifiers [29, 30]), 98.75–98.82 % (deep belief nets [31]) and 98.6–99.32 % (SVM [30]). Note that all of these take 60,000 of the digits as a training set and 10,000 digits as a testing set [29], in comparison to our approach where we take only 3.6 % of the points for the fidelity term. Our SSL method is nevertheless competitive with these supervised methods. Moreover, we perform no preprocessing or initial feature extraction on the image data, unlike most of the other methods we compare with (we have excluded from the comparison, however, methods that explicitly deskew the image). While there is a computational price to be paid in forming the graph when data points use all 784 pixels as features, this is a simple one-time operation.

5 Conclusions

We have proposed a new multiclass segmentation procedure, based on the diffuse interface model. The method obtains segmentations of several classes simultaneously without using one-vs-all or alternative sequences of binary segmentations required by other multiclass methods. The local scaling method of Zelnik-Manor and Perona, used to construct the graph, constitutes a useful representation of the characteristics of the data set and is adequate to deal with high-dimensional data.

Our modified diffusion method, represented by the non-linear smoothing term introduced in the Ginzburg-Landau functional, exploits the structure of the multiclass model and is not affected by the ordering of class labels. It efficiently propagates class information that is known beforehand, as evidenced by the small proportion of fidelity points (2 % – 10 % of dataset) needed to perform accurate segmentations. Moreover, the method is robust to initial conditions. As long as the initialization represents all classes uniformly, different initial random configurations produce very similar results. The main limitation of the method appears to be that fidelity points must be representative of class distribution. As long as this holds, such as in the examples discussed, the long-time behavior of the solution relies less on choosing the "right" initial conditions than do other learning techniques on graphs.

State-of-the-art results with small classification errors were obtained for all classification tasks. Furthermore, the results do not depend on the particular class label

assignments. Future work includes investigating the diffuse interface parameter ϵ. We conjecture that the proposed functional converges (in the Γ-convergence sense) to a total variational type functional on graphs as ϵ approaches zero, but the exact nature of the limiting functional is unknown.

Acknowledgments This research has been supported by the Air Force Office of Scientific Research MURI grant FA9550-10-1-0569 and by ONR grant N0001411AF00002.

References

1. Coifman, R.R., Lafon, S., Lee, A.B., Maggioni, M., Nadler, B., Warner, F., Zucker, S.W.: Geometric diffusions as a tool for harmonic analysis and structure definition of data: diffusion maps. Proc. Natl. Acad. Sci. **102**, 7426–7431 (2005)
2. Allwein, E.L., Schapire, R.E., Singer, Y.: Reducing multiclass to binary: a unifying approach for margin classifiers. J. Mach. Learn. Res. **1**, 113–141 (2000)
3. Bertozzi, A.L., Flenner, A.: Diffuse interface models on graphs for classification of high dimensional data. Multiscale Model. Simul. **10**, 1090–1118 (2012)
4. Bertozzi, A., Esedoḡlu, S., Gillette, A.: Inpainting of binary images using the Cahn-Hilliard equation. IEEE Trans. Image Process. **16**, 285–291 (2007)
5. Jung, Y.M., Kang, S.H., Shen, J.: Multiphase image segmentation via Modica-Mortola phase transition. SIAM J. Appl. Math. **67**, 1213–1232 (2007)
6. Li, Y., Kim, J.: Multiphase image segmentation using a phase-field model. Comput. Math. Appl. **62**, 737–745 (2011)
7. Chung, F.R.K.: Spectral graph theory. In: Regional Conference Series in Mathematics. Conference Board of the Mathematical Sciences (CBMS), vol. 92. Washington (1997)
8. Zhou, D., Schölkopf, B.: A regularization framework for learning from graph data. In: Workshop on Statistical Relational Learning. International Conference on Machine Learning. Banff (2004)
9. Szlam, A.D., Maggioni, M., Coifman, R.R.: Regularization on graphs with function-adapted diffusion processes. J. Mach. Learn. Res. **9**, 1711–1739 (2008)
10. Wang, J., Jebara, T., Chang, S.F.: Graph transduction via alternating minimization. In: Proceedings of the 25th International Conference on Machine Learning (2008)
11. Bühler, T., Hein, M.: Spectral clustering based on the graph p-Laplacian. In: Bottou, L., Littman, M. (eds.) Proceedings of the 26th International Conference on Machine Learning, pp. 81–88. Omnipress, Montreal (2009)
12. Szlam, A., Bresson, X.: Total variation and cheeger cuts. In: Fürnkranz, J., Joachims, T. (eds.) Proceedings of the 27th International Conference on Machine Learning, pp. 1039–1046. Omnipress, Haifa (2010)
13. Hein, M., Setzer, S.: Beyond spectral clustering—tight relaxations of balanced graph cuts. In: Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K. (eds.) Advances in Neural Information Processing Systems, vol. 24, pp. 2366–2374 (2011)
14. Dietterich, T.G., Bakiri, G.: Solving multiclass learning problems via error-correcting output codes. J. Artif. Intell. Res. **2**, 263–286 (1995)
15. Hastie, T., Tibshirani, R.: Classification by pairwise coupling. In: Advances in Neural Information Processing Systems, vol. 10. MIT Press, Cambridge (1998)
16. Har-Peled, S., Roth, D., Zimak, D.: Constraint classification for multiclass classification and ranking. In: Becker, S.T.S., Obermayer, K. (eds.) Advances in Neural Information Processing Systems, vol. 15, pp. 785–792. MIT Press, Cambridge (2003)
17. Subramanya, A., Bilmes, J.: Semi-supervised learning with measure propagation. J. Mach. Learn. Res. **12**, 3311–3370 (2011)

18. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. In: Thrun, S., Saul, L.K., Schölkopf, B. (eds.) Advances in Neural Information Processing Systems, vol. 16, pp. 321–328. MIT Press, Cambridge (2004)
19. Kohn, R.V., Sternberg, P.: Local minimizers and singular perturbations. Proc. R. Soc. Edinburgh Sect. A **111**, 69–84 (1989)
20. Dobrosotskaya, J.A., Bertozzi, A.L.: A wavelet-Laplace variational technique for image deconvolution and inpainting. IEEE Trans. Image Process. **17**, 657–663 (2008)
21. Gilboa, G., Osher, S.: Nonlocal operators with applications to image processing. Multiscale Model. Simul. **7**, 1005–1028 (2008)
22. Zelnik-Manor, L., Perona, P.: Self-tuning spectral clustering. In: Saul, L.K., Weiss, Y., Bottou, L. (eds.) Advances in Neural Information Processing Systems, vol. 17. MIT Press, Cambridge (2005)
23. von Luxburg, U.: A tutorial on spectral clustering. Technical Report TR-149, Max Planck Institute for Biological Cybernetics (2006)
24. Dobrosotskaya, J.A., Bertozzi, A.L.: Wavelet analogue of the Ginzburg-Landau energy and its gamma-convergence. Interfaces Free Bound. **12**, 497–525 (2010)
25. Bertozzi, A., van Gennip, Y.: Gamma-convergence of graph Ginzburg-Landau functionals. Adv. Differ. Equ. **17**, 1115–1180 (2012)
26. Surendran, D.: Swiss roll dataset. http://people.cs.uchicago.edu/~dinoj/manifold/swissroll.html (2004)
27. Nene, S., Nayar, S., Murase, H.: Columbia object image library (COIL-100). Technical Report CUCS-006-96 (1996)
28. Chapelle, O., Schölkopf, B., Zien, A. (eds.): Semi-supervised Learning. MIT Press, Cambridge (2006)
29. LeCun, Y., Cortes, C.: The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/
30. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**, 2278–2324 (1998)
31. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Comput. **18**, 1527–1554 (2006)

Probabilistic Discriminative Dimensionality Reduction for Pose-Based Action Recognition

Valsamis Ntouskos, Panagiotis Papadakis and Fiora Pirri

Abstract We examine the problem of classifying action sequences given a small set of examples for each type of action. Based on the presumption that human motion resides in a low dimensional space, we introduce a probabilistic dimensionality reduction model able to recover the structure of a low-dimensional manifold where all the involved actions reside. Requiring that sequences of the same action are placed apart from other sequences, we are able to achieve higher classification rates, with respect to other commonly used techniques, by performing the classification on this manifold. The main contribution is the introduction of a new model, based on Back-constrained GP-LVM which can be used for the efficient classification of sequences. We compare our method with the classification based on the Dynamic Time Warping distance and with the V-GPDS model, adapted for classification. Results are provided for sequences taken from two publicly available datasets which highlight different aspects of the method.

Keywords Action recognition · Dimensionality reduction · Manifold learning · Time series models · Motion capture

1 Introduction

Human action recognition is one of the most challenging applications in the field of computer vision. It requires to infer an action model from the observation of a motion sequence, hence it requires the solution of an inverse problem [18]. Furthermore, the modelling process is based on several steps tackling, in turn, different sub-problems: data acquisition, motion analysis and segmentation in individual actions, alignment

V. Ntouskos (✉) · P. Papadakis · F. Pirri
ALCOR Laboratory, Sapienza University of Rome, Rome, Italy
e-mail: ntouskos@dis.uniroma1.it

P. Papadakis
e-mail: papadakis@dis.uniroma1.it

F. Pirri
e-mail: pirri@dis.uniroma1.it

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_9

between sequences and classification with respect to a given taxonomy. All these steps are computationally expensive, while ideally recognition should be performed online.

In this paper we address the alignment and classification part of the complete pipeline. Namely, we assume that a sequence that captures an individual action is already available and the task is to recognize the performed action. To this end we introduce a model based on the the Back-Constrained GP-LVM introduced in [9, 10], and extend it for the application of action recognition, exploiting the strength of a lower dimensional manifold. In detail, we derive a discriminative, probabilistic dimensionality reduction model for mapping motion capture sequences in a low dimensional latent space which assists the action classification process. The proposed model introduces a latent space featuring a fixed set of actions and constrains feature distances in data space to be suitably projected in the latent space, in order to preserve the clustering of common patterns. Actions are represented as a sequence of poses, which can be taken from motion capture (MoCap) data. This projection ensures a discriminative power to the GP-LVM model and it also exploits the peculiar property of action sequences of being reducible to a lower dimensional manifold [17].

In Sect. 2 we briefly review recent works on pose-based action recognition and dimensionality reduction, showing the major trends of research in this field. In Sect. 3 we overview the theoretical foundation of GP-LVM on which our model is based. In Sect. 4 we present our discriminative model. Section 5 demonstrates the latent space structure recovered by the proposed model and examines its performance on human action classification. We compare our method with a sequence classification method based on Dynamic Time Warping as well as the Variational Gaussian Process Dynamical Systems [6] recently proposed for modelling high dimensional dynamical systems. We conclude the work addressing possible extensions.

2 Related Work

In this section we review some of the main approaches to action recognition and mainly those which refer to manifold learning or treat the problem of action recognition in MoCap sequences.

So far many techniques have been proposed in the literature regarding action recognition where stochastic, volumetric or non-parametric models are most commonly employed. Detailed reviews of the techniques which have been considered in the research on human motion analysis and on action recognition can be found in [1, 12, 26]. Several works address the problem of modelling and recognizing human motion by learning the structure of the low dimensional manifold where it resides, and by recovering a mapping between the high dimensional observations and this manifold.

In [7] the authors consider MoCap sequences and they learn the structure of a unidimensional smooth manifold by applying the tensor voting technique [13]. A motion distance score is used to compute the similarity between the actions recorded

in two different sequences. The setting provides the possibility to compare also actions extracted from videos with actions taken from MoCap sequences.

In [34] the authors consider a two dimensional manifold with a toroidal topology in order to estimate human motion. They build on the idea of Gaussian Process Latent Variable Models (GP-LVM) [9] to identify a manifold which jointly captures gait and pose, via three different models. They introduce a new model (JGPM) which they compare to two constrained latent variable models based on GP-LVM and Local Linear GP-LVM [29] respectively.

In [23] the authors propose a non-linear generative model for human motion data that considers binary latent variables. The introduced architecture makes on-line inference efficient and allows for a simple approximate learning procedure. The method performance is evaluated by synthesizing various motion sequences and by performing on-line filling in of data, lost during motion capture.

Following a different perspective, in [21] the authors explore the space of actions, spanned by a set of action-bases, to identify some action invariants with respect to viewpoint, execution rate and subject's body shape. Action recognition is performed for four different kind of actions (sitting, standing, running and walking) and the results show that it is possible to correctly classify most of these actions using the proposed method.

The redundancy of the original representation of MoCap sequences is also exploited in [11] where a compressive sensing method is introduced. Here the authors argue that human actions are sparse in the action space domain as well as the time domain, and they seek therefore a sparse representation. The sparse representation introduced can assist in different applications regarding MoCap data like motion approximation, compression, action retrieval and action classification.

Finally, in [32] (see also [30, 33]) the authors examine whether and to what extent the use of information about the subject's pose assists recognition. In this case, several pose-based features are used, based on the relative pose features introduced in [14, 15]. Their results suggest that knowing the pose of the subject leads to better results, in terms of classification rate. It is also shown that pose based features alone are usually sufficient, as their combination with appearance based features is usually not leading to higher classification rate.

3 Gaussian Process Latent Variable Models

In this section we review Gaussian Process Latent Variable Models [9]. A Gaussian process is a collection of random variables such that any finite collection of them has a Gaussian distribution [19]. Namely, a random variable of a Gaussian process is $f(\mathbf{x}_i) = \mathcal{GP}(\mu(\mathbf{x}_i), k(\mathbf{x}_i, \mathbf{x}_j))$, with μ and $k(\mathbf{x}, \mathbf{x}')$ the mean and covariance function of the process respectively, indexed over the set $\mathcal{X}$ of all the possible inputs. The Gaussian process is a non parametric prior for the random variable $f(\mathbf{x}_i)$ where $\mathbf{x}_i$ is the deterministic input. Gaussian processes have been successfully used for both regression and classification tasks.

In [9] the author shows that Principal Component Analysis (PCA) can be interpreted as a product of Gaussian processes mapping latent-space points to points in data-space, when the covariance function is linear; when instead a non-linear covariance function is used, such as an RBF kernel then the mapping is non-linear. Lawrence shows the advantages in using Gaussian Processes Latent Variable Models (GP-LVM); for example, for optimization purposes, the data can be divided in active and inactive, according to some rule. Then, because points in the inactive set project into the data-space as Gaussian distributions, due to the properties of the variance the likelihood of each data point can be optimized independently.

In addition to the advantage in terms of visualization and computational efficiency highlighted in [9], GP-LVM turns out to be a powerful unsupervised learning algorithm. Indeed, GP-LVM can manage, via the non-linear mapping of the latent variables to the data-space, noisy or incomplete input data, when Gaussian processes are used as non parametric priors for them.

At this point, we introduce some preliminary definitions that we will refer throughout the following sections

Let $\mathbf{Y}$ be the normalized data in $\mathbb{R}^{N\times d}$, for example specifying the pose of a subject in space, with respect to a coordinate frame; let $\mathbf{X}$ be the mapped positions in latent-space, with $\mathbf{X} \in \mathbb{R}^{N\times q}$, with $q \leq d$. Let f be a mapping, such that:

$$y_{nj} = f(\mathbf{x}_n, \mathbf{w}_j) + \epsilon_{nj}, \tag{1}$$

Here, y_{nj} is the observed element of the nth row and jth column of $\mathbf{Y}$, ϵ_{nj} denotes the noise affecting the mapping and $\mathbf{x}_n$, the nth row of $\mathbf{X}$, and $\mathbf{w}_j$ are the parameters of the mapping f. Given a Gaussian process as a prior on f, when the prior is the same on each of the f functions one obtains [9]:

$$p(\mathbf{Y}|\mathbf{X}, \theta) = \prod_{j=1}^{d} \mathcal{N}(\mathbf{y}_j|\mathbf{0}, \mathbf{K}) \tag{2}$$

Here, $\mathbf{y}_j$ is the jth column of $\mathbf{Y}$ and $\mathbf{K}$ is the $N \times N$ kernel of the Gaussian process. We see that (2) suggests a conditional independence in the data space, given the latent space representation.

Learning amounts to maximizing the likelihood of the position of the latent variables $\mathbf{X}$ and θ, which are the parameters of the kernel:

$$L(\mathbf{X}, \theta) = -\frac{d}{2}\log|\mathbf{K}| - \frac{1}{2}\mathrm{Tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\top}\right) \tag{3}$$

In order to optimize the non-linear model, it is necessary to initialize the model using appropriate initial values for the positions of the latent-space points. It is also necessary to initialize the hyperparameters of the model. Optimization is obtained by an iterative minimization of the objective function, by using a gradient based algorithm. As the model is non-linear, the hypersurface is subject to local-minima,

so the initialization of the positions of the latent-space points is crucial. When non-linear dimensionality reduction methods are used for the initialization, like local linear embedding (LLE) [20] or ISOMAP [24], the structure of the manifold is expected to be more accurately recovered. GP-LVM have been exploited in many applications as for example in [27–29, 31].

4 Discriminative Sequence Back-Constrained GP-LVM

As mentioned in the previous sections, models from the family of GP-LVM methods are well suited for predicting missing values or missing samples of time sequences. However, they do not seem to perform equally well when they are used for clustering and classification problems, particularly for time-series data. This drawback of the classical GP-LVM methods can be also witnessed by observing that it is hard to recover the structure of a common latent-space for a set of sequences, as their latent space representations are scattered across the latent-space and no relation can be drawn between sequences corresponding to the same action. This is due to the fact that standard GP-LVM models do not provide a mechanism to encourage points to be placed closer to each other in the latent-space when they belong to the same class and the same also holds at the level of individual sequences.

Local distances can be directly used in GP-LVM to provide a common latent-space representation as they are well suited for classification purposes. In fact local distances in data-space provide some information regarding the intra-class variation. Lawrence and Quiñonero-Candela in [10] have introduced Back-Constrained GP-LVM which considers local distances in the data-space. The GP-LVM model uses a product of Gaussian processes to map from the latent-space to the data-space. Each of these processes refers to a different dimension of the data-space and it is governed by the coordinates of the latent-points. In order to obtain a smooth mapping in the opposite direction, the authors in [10] propose to construct this mapping by means of a kernel based regression. Adopting this technique, the latent points are constrained to be the product of a smooth mapping from the data-space. This forces small distances in data-space to lead to small distances between the corresponding points in the latent-space. The smoothness of the mapping from the data-space to the latent-space is determined by the kernel function. Using this mapping, it is not needed to perform a new optimization to approximate the latent-space representation of new data.

The previous method cannot be directly applied on data originating from sequences, as it is expected that individual elements of a sequence do not provide sufficient information regarding the characteristics of the entire sequence. Building on the same principle, namely the use of local distances in the data-space as back-constraints, we formulate a GP-LVM variant which considers entire sequences rather than individual data points.

Before introducing our model, we briefly review the Dynamic Time Warping (DTW) algorithm, as well as a set of sequence alignment kernels based on DTW and its variants, which will be used for the derivation of our model.

4.1 Dynamic Time Warping and Sequence Alignment Kernels

Dynamic Time Warping is used to match two time dependent sequences by nonlinearly warping one sequence onto the other. Let us consider two vector sequences $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)$ with $N \in \mathbb{N}$ and $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_M)$ with $M \in \mathbb{N}$. Each vector in the sequence belongs to a n-dimensional feature space $\mathcal{F}$ so $\mathbf{y}_n, \mathbf{z}_m \in \mathcal{F}$. A local distance measure is defined to compare a pair of features, provided by an appropriate kernel function:

$$\kappa : \mathcal{F} \times \mathcal{F} \to \mathbb{R}^+ \tag{4}$$

A warping path is a sequence $p = (p_1, \ldots, p_L)$ where each element is a pair $p_l = (n_l, m_l)$. The total cost of a warping path p, according to the predefined distance measure, is:

$$c_p(\mathbf{y}_n, \mathbf{z}_m) = \sum_{l=1}^{L} \kappa(\mathbf{y}_{n_l}, \mathbf{z}_{m_l}) \tag{5}$$

The Dynamic Time Warping distance between two sequences is defined as the minimal total cost among all possible warping paths. To obtain this value we have to solve the following optimization problem:

$$DTW(\mathbf{Y}, \mathbf{Z}) = \min_p \left\{ c_p(\mathbf{Y}, \mathbf{Z}) \right\} \tag{6}$$

We can also identify an optimal warping path (not necessarily unique):

$$p^* = \arg\min_p \left\{ c_p(\mathbf{Y}, \mathbf{Z}) \right\} \tag{7}$$

The DTW distance is well-defined, even though there may exist many warping paths of minimal total cost. Moreover, it is symmetric if the distance measure is also symmetric, but it is not a proper metric, as it does not satisfy the triangle inequality. In order to apply DTW on MoCap sequences, we must first define the local cost measure κ. Two popular choices are to use the sum of the geodesic distances between the unit-quaternions representing the joint angles, as well as the optimal alignment distance between the three dimensional positions of the joints [14].

Based on the notions of the DTW distance and the optimal warping path, alignment kernels have been proposed which consider entire sequences as a whole. As an example we cite here [2, 5, 22].

4.2 Sequence Back-Constrained GP-LVM

In this section we show how to enforce a clustering of the sequences in the latent-space, governed by their respective similarity, which will enable a more accurate classification of a new sequence. To ensure that data instances which are close to each other in the data-space, are mapped to positions which are close also in the latent-space, we apply a similarity measure for comparing different sequences and identify a characteristic feature, summarizing the entire sequence.

Here we consider that each frame of a motion sequence is represented as a d-dimensional array. An entire sequence, with index s, is represented thus as a set of d dimensional arrays of cardinality L_s, forming a matrix $\mathbf{Y}_s \in \mathbb{R}^{L_s \times d}$. A collection of S motion sequences is represented as the concatenation of the respective sub-matrices forming the data-matrix $\mathbf{Y} \in \mathbb{R}^{N \times d}$, with $N = \sum_{s=1}^{S} L_s$. Let $\mathcal{J}_s$ be the set of indices of the sth sequence in the data matrix, the corresponding representation of the data-points in the q dimensional latent-space form a matrix $\mathbf{X} \in \mathbb{R}^{N \times q}$. The coordinates of the centroid of the latent-space representation of the sth sequence, is defined as:

$$\mu_{sq} = \frac{1}{L_s} \sum_{n \in \mathcal{J}_s} x_{nq} \tag{8}$$

The likelihood of the GP-LVM model is given by (3). The centroid of the latent positions of the data points is taken to be the characteristic feature of the sequence. Therefore, we require that the local distances between the sequences in data-space, computed via the DTW technique, are preserved in latent-space; thus they are specified as the distances between the centroids $\boldsymbol{\mu}_s$. Hence, we consider a mapping to the latent-space governed by an alignment kernel k:

$$g_q(\mathbf{Y}_s) = \sum_{m=1}^{S} a_{mq} k(\mathbf{Y}_s, \mathbf{Y}_m) \tag{9}$$

The degree to which the local distances in the data-space are preserved depends on the particular characteristics of the kernel employed for the mapping.

We, thus, have to maximize a constrained likelihood, instead of maximizing the likelihood of the original GP-LVM model.

Each of the $S \cdot q$ constraints can be written as:

$$g_q(\mathbf{Y}_s) - \mu_{sq} = 0 \tag{10}$$

Maximizing the constrained likelihood of the model, we expect to obtain a latent-space representation, where similar sequences are better grouped together, with respect to the representation obtained by the original model. Another important advantage of this approach is that we can use the inverse mapping recovered in the

learning phase, for the purposes of fast inference. In this way, we avoid the costly operation of reoptimisation, which is otherwise necessary to obtain the latent-space representation of new sequences.

Up to this point, we did not consider the labels of each type of sequence. In the following section, we modify our model by replacing the Gaussian prior with a prior which will make the model more discriminative.

4.3 Discriminative Sequence Back-Constrained GP-LVM

Discriminative GP-LVM (D-GPLVM) has been originally introduced in [27]. In order to make the Sequence Back-Constrained GP-LVM (SB-GPLVM) model more discriminative, we can consider a measure of the between-group variation and the within-group separation. Referring to Fisher's Discriminant Analysis, in case we need to estimate a linear projection of the data, such that an optimal separation is achieved, we need to maximize the ratio of the *between-group-sum of squares* to the *within-group-sum of squares*.

We thus seek the direction of projection given by the vector $\mathbf{a}$ which provides a good separation of the data. Denoting as $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^{\mathrm{T}}$ the low dimensional representation of the data points $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]^{\mathrm{T}}$, the *between-group-sum of squares* is given as:

$$\mathbf{a}^{\mathrm{T}}\mathbf{B}\mathbf{a} = \sum_{c=1}^{C} \frac{N_c}{N} \mathbf{a}^{\mathrm{T}}(\mu_c - \mu_0)(\mu_c - \mu_0)^{\mathrm{T}}\mathbf{a} \tag{11}$$

The *within-group-sum of square* is given as:

$$\mathbf{a}^{\mathrm{T}}\mathbf{W}\mathbf{a} = \frac{1}{N} \sum_{c=1}^{C} \sum_{n=1}^{N_c} \mathbf{a}^{\mathrm{T}}(\mathbf{x}_n^{(c)} - \mu_c)(\mathbf{x}_n^{(c)} - \mu_c)^{\mathrm{T}}\mathbf{a} \tag{12}$$

Here $\mathbf{X}^{(c)} = [\mathbf{x}_1^{(c)}, \ldots, \mathbf{x}_{N_c}^{(c)}]^{\mathrm{T}}$ are the N_c points which belong to the class c, $\boldsymbol{\mu}_c$ is the mean of the elements of class c and $\boldsymbol{\mu}_0$ is the mean computed across all the points.

The criterion used for maximizing between-group separability and minimizing within-group variability is the following [8]:

$$J(\mathbf{X}) = \mathrm{Tr}(\mathbf{W}^{-1}\mathbf{B}) \tag{13}$$

Based on the previous discussion, in order to transform the SB-GPLVM model making it discriminative, it is necessary to replace the Gaussian prior with a prior which depends on (13). This prior takes the following form:

$$p(\mathbf{X}) = \frac{1}{\alpha} \exp\left\{-\frac{\gamma}{2} J^{-1}\right\} \tag{14}$$

where α is a normalization constant, possibly depending on p, and γ represents the scaling factor of the prior.

The log likelihood associated with the discriminative model becomes:

$$L = -\frac{d}{2}\log|\mathbf{K}| - \frac{1}{2}\mathrm{Tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\right) - \frac{\gamma}{2}\mathrm{Tr}\left(\mathbf{B}^{-1}\mathbf{W}\right) \tag{15}$$

The parameter γ controls the relative importance of the discriminative prior and it reflects the ability of the model to be more discriminative or more generalizing, according to the value it takes.

4.4 Classification Based on D-SBGPLVM

In this section we illustrate how to compute the latent representation of the data points belonging to a new sequence. This will allow to classify any new sequenced according to the introduced D-SBGPLVM model. Let $\mathbf{Y}_*$ be the data-space representation of a new sequence and $\mathbf{X}_*$ the corresponding latent-space representation. The new sequence's centroid in latent-space can be estimated orders of magnitude faster than $\mathbf{X}_*$ by making use of Eq. (9) introduced in Sect. 4.2. Thus, the coordinates of the test sequence centroid, in each dimension of the latent space are given by:

$$\forall\, q:\ \mu_{*q} = g_q(\mathbf{Y}_*) = \sum_{s=1}^{S} a_{qs}\, k(\mathbf{Y}_*, \mathbf{Y}_s) \tag{16}$$

where μ_{*q} is the qth dimension coordinate of the centroid $\boldsymbol{\mu}_*$ of the test sequence. In this case, no minimization is required and the time, necessary for computing the coordinates of the centroid of the test sequence, is proportional to the time needed to compute the kernel values.

At this point, any multi-class classification method can be employed, in order to perform classification. As the latent-space has a dimensionality much smaller than the original data-space, it is expected that classification is more robustly performed in the latent representation of the sequences. Moreover, the proposed method provides a concise way to classify sequences as a whole, as the model treats them explicitly as individual entities.

5 Results

The ability of the Discriminative Sequence Back-Constrained GP-LVM model to provide a latent-space representation suitable for efficient and robust classification of sequences, is examined in this section.

Evaluation on the HDM05 "Cuts" *Dataset* [16]. Part of the "Cuts" sequences, contained in the HDM05 dataset, has been used for evaluating the model we propose, in comparison to other methods which can be used for sequence classification. This dataset includes the following actions: *Clapping hands*-5 repetitions (17 sequences); *Hopping on right leg*-3 reps. (12 seqs.); *Kick with right foot in front*-2 reps. (15 seqs.); *Running on place*-4 steps (15 seqs.); *Throwing high with right hand while standing* (14 seqs.); *Walking starting with right foot*-4 steps (16 seqs.).

The sequences are sampled at a frequency of 120 frames per second. For this dataset, sequences are already accurately segmented, in order to contain a single action with the same number of repetitions.

The results of the proposed method are compared with the classification results, obtained by directly using the DTW distances of the sequences in the data-space, as well as using the highest class-conditional densities obtained by the Variational Gaussian Process Dynamical Systems (V-GPDS) method [6]. All results are taken by Cross-Validation. Each experiment is performed by keeping all action sequences of one of the five subjects as test sequences and by using the sequences of the other four subjects as training instances. Finally, the results are averaged over the five individual experiments.

Table 1 gives the accuracy rate achieved with each of these three methods for each action as well as in average. Regarding the results obtained by the proposed method, relative features are used and the dimensionality of the latent-space space is fixed to four. Moreover, for the back-constraints the kernel proposed in [2] is used and the initial positions of the latent points are obtained by using the Local Linear Embedding algorithm [20]. Finally, classification in latent-space is performed by SVMs using the RBF kernel function. Figure 1 shows the corresponding confusion matrix obtained by using the D-SBGPLVM model.

One can see from the results provided in Table 1 that our method gives the best results, both for each individual type of action, except for Hop, as well as in average. We observe that the classification accuracy is relatively high for the DTW distance alone. This depends also on the fact that this dataset is specifically constructed in such a way, that actions of the same kind can be aligned with a very small cost. This is possible as they are defined at a high detail level regarding their execution and they

Table 1 Comparison of the classification results for the HDM05 "Cuts" dataset

	DTW (%)	V-GPDS (%)	D-SBGPLVM (%)
Clap	70.6	16.7	88.2
Hop	100	66.7	83.3
Kick	40.0	33.3	53.3
Run	66.7	33.3	80.0
Throw	64.3	50.0	78.6
Walk	100	83.3	100
Average	73.0	47.2	80.9

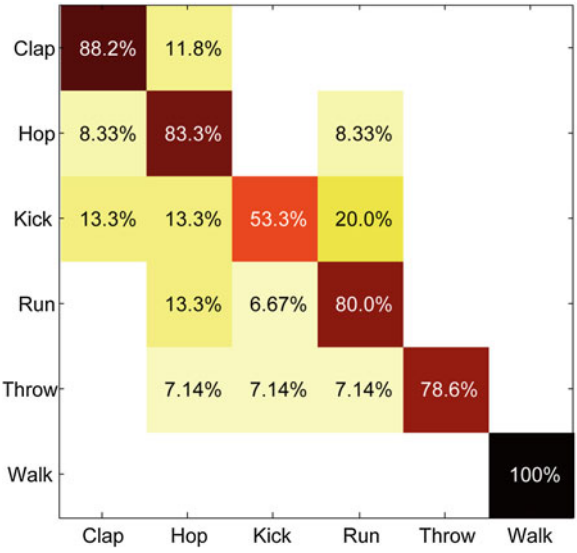


Fig. 1 Confusion matrix by using D-SBGPLVM model in combination with SVM on the HDM05 "Cuts" dataset. Average accuracy: 80.9 %

have been also accurately segmented manually. Regarding classification of human actions using the V-GPDS model, it is necessary to train a different model for each individual type of action. After a model has been trained for each type of action, it is possible to compute the class conditional densities for the new sequence.

Considering that the analogous model of V-GPDS, which does not consider time dynamics introduced in [25], provides good classification results (e.g. on the USPS Handwritten Digits Dataset) we expected higher classification rates for the adapted V-GPDS model. Searching the cause of this issue, we have noticed that models for certain actions tend to provide quite high conditional densities most of the time. Further investigation is needed in this direction, as the experiments performed using V-GPDS were not sufficient to derive safe conclusions and possibly a more suitable adaptation of the model for classification purposes is needed.

In the case of D-SBGPLVM, the model is trained by optimising the latent coordinates of the sequences and the hyper-parameters of the model by using all training sequences. By the optimisation process, we recover also the parameters of the kernel based regression, which forms the inverse mapping from the data-space to the latent-space. We provide some examples of bi-dimensional latent-spaces recovered by training the model using sequences of the HDM05 "Cuts" dataset in Fig. 2. In these figures, each color corresponds to a different class of action, crosses are the latent representations for each individual data point, triangles correspond to the centroids of the training sequences and finally the squares correspond to the estimated position of the testing sequences centroids computed using the back-constraints. In Fig. 2 the recovered latent-spaces are shown for three different types of representa-

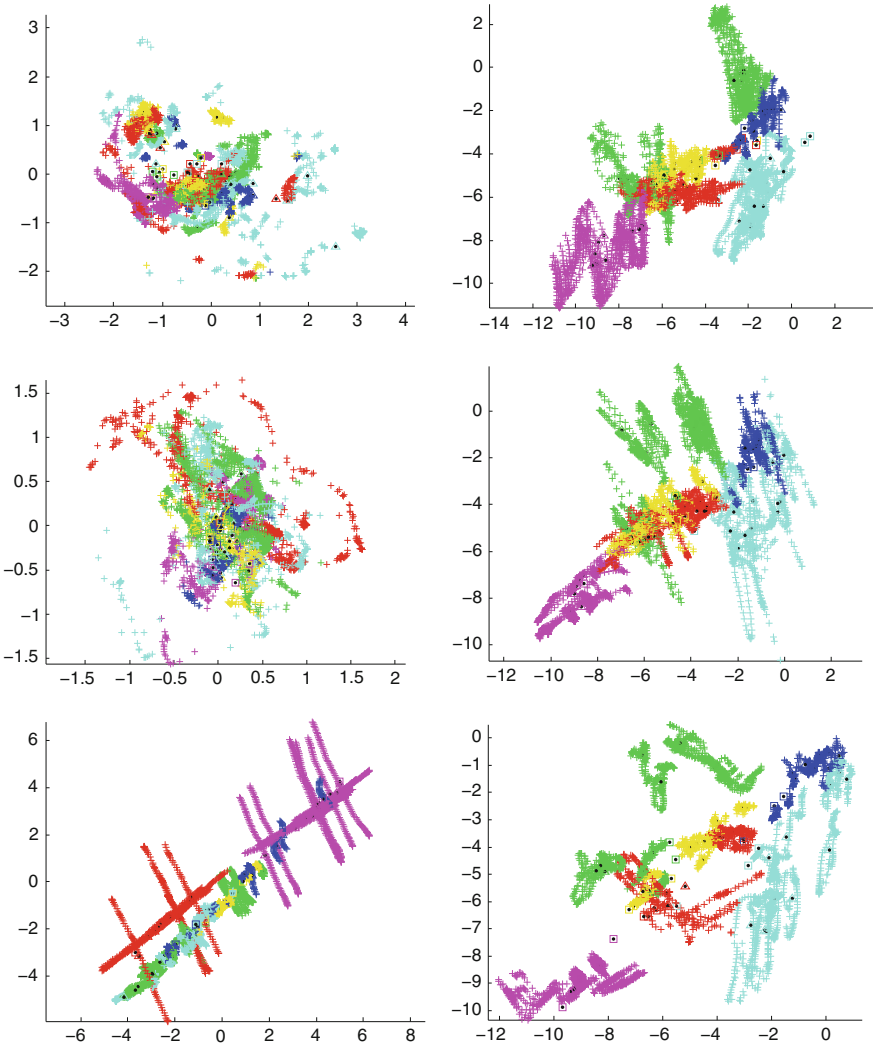


Fig. 2 *Left Column* Latent-space representation by PPCA initalization and considering Euler Angles (*Top*), Unit-Quaternion (*Middle*) and 3D Point Cloud (*Bottom*) representations *Right Column* latent-space representation considering relative features representation and PPCA (*Top*), LLE (*Middle*) and ISOMAP (*Bottom*) initalization

tions considered for the sequences and by using Probabilistic PCA in order to retrieve initial values for the latent points. In the case of Euler Angles and Unit-Quaternions, one can notice that different sequences are placed on top of each other and thus we expect classification rates to be low.

Our interpretation is that this mainly depends on the high non-linearity of the data-space and the fact the PPCA, being a linear dimensionality reduction technique, is not

able to provide suitable initial values for the latent points. As our model is non-linear and it is optimized by using a gradient based algorithm, it is susceptible to local minima. However, in the case of 3D point cloud representation, the data-space does not show excessive non-linearity and even PPCA initialization seems to be sufficient to recover a better structure for the latent-space.

The case of Relative Features (as in [14], but without discretization based on some threshold) is examined also in Fig. 2. Relative features include for example the distance between two specified joints, the distance of a joint with respect to the plane defined by three other, the angle between two successive joints etc. Here we can better observe the impact of the initialization technique on the resulting structure of the latent-space. It is evident that the use of more sophisticated non-linear dimensionality reduction techniques to obtain the initial values, helps recovering a better structure of the common latent-space.

Evaluation on actions of the CMU Dataset [4]. Seven actions from the CMU dataset have been also considered for evaluating the model we propose. This dataset includes the following actions: *Walking* (15 sequences); *Running* (15 seqs.); *Jumping* (15 seqs.); *Sitting-Standing* (7 seqs.); *Throwing-Tossing* (15 seqs.); *Boxing* (9 seqs.); *Dancing* (9 seqs.).

Each of these actions is performed from a different actor. Moreover, the actions have not been hand-picked and their label only relies on the default labelling provided by the publishers of the dataset. Finally, motion sequences have not been manually segmented. We perform classification instead by just considering the first two seconds of each sequence. For these reasons, we can see that this dataset represents a more challenging and realistic instance of the action recognition problem. Five-fold cross-validation has been used here for obtaining the final classification results.

The classification accuracy achieved by the proposed method, compared with the results of DTW distances and V-GPDS method, are provided in Table 2. Here, Euler angles are considered as features provided to the D-SBGPLVM, while the rest of the setting is the same with the one described for the "Cuts" experiments. In Fig. 3 we provide the corresponding confusion matrix and the overall classification rate, when the D-SBGPLVM model is used.

Table 2 Comparison of the classification results for the actions taken from CMU dataset

	DTW (%)	V-GPDS (%)	D-SBGPLVM (%)
Walk	80.0	40.0	66.7
Run	60.0	40.0	66.7
Jump	86.7	40.0	73.3
Throw-Toss	80.0	40.0	80.0
Sit-Stand	46.7	40.0	80.0
Box	100	20.0	80.0
Dance	26.7	80.0	73.3
Average	63.5	42.9	72.9

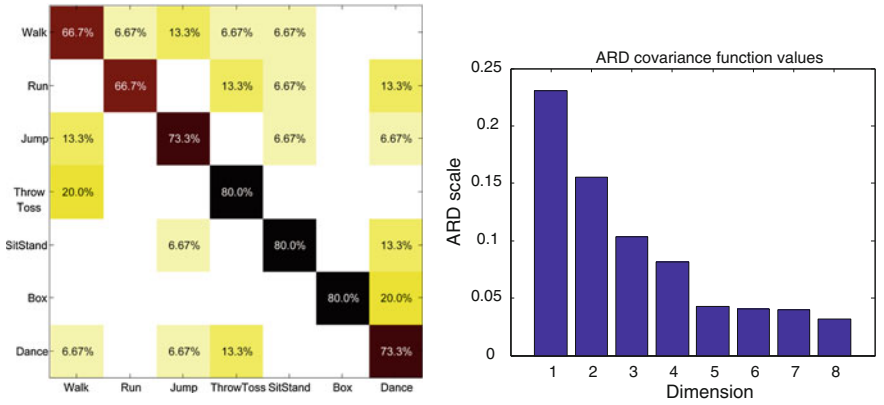


Fig. 3 *Left* Confusion matrix by using D-SBGPLVM model in combination with SVM on the CMU dataset. Average accuracy: 72.9 % *Right* sorted ARD covariance function values obtained after training the model for the same dataset using the RBF-ARD kernel. Average accuracy: 74.1 %

We can observe here, that the results for the "CMU" dataset are analogous to the ones corresponding to the "Cuts" dataset. We expect that the lower rate achieved in general by all algorithms mainly depend on the particular difficulties which characterise this dataset, as mentioned above. Considering these difficulties, one can see that the proposed model gives satisfying classification results. This also demonstrates the generalization capabilities of the proposed probabilistic model, which based on this characteristic leads to an overall accuracy that exceeds the accuracy achieved by the other two methods considered here.

The same experiments were also performed by considering the recently proposed 'path kernel' [3] providing equivalent results. The classification rate was slightly lower but this may be related to the particular selection of the parameters of the kernel. Moreover, we performed trials using the automatic relevance determination (ARD) squared exponential kernel as in [6, 25]. In this case, considering eight dimensions for the latent space, we obtained a classification rate of 74.1 % for the CMU dataset. What is important to note here are the values of the relative importance of each dimension after training the model, shown in Fig. 3. One can see here that most of the information for the actions is embedded in a four dimensional sub-manifold. This result is in accordance with the ones reported in [17].

6 Conclusions

In this paper, we have introduced a novel GP-LVM variant in order to recover the structure of a lower dimensional manifold for a set of sequences of different action types. We have shown that the model, according to our approach, attains increased classification accuracy by working in the low dimensional latent-space instead of

the original data-space. By exploiting the inverse mapping, from the data-space to the latent-space, our approach is able to infer the class of a new sequence within a few seconds (Matlab implementation tested on the following system: 2.2 GHz quad-core AMD Phenom, 4 GB RAM). This provides a crucial advantage with respect to other GP-LVM models which require several minutes to complete this task, having to deal with a new optimization to obtain the latent-space representation of the new data instances. We have further shown that the proposed D-SBGPLVM model attains classification rates equivalent to the current state-of-the-art when combined with a standard classifier, as for example SVM, for classification in the latent-space.

Within the directions of our future work, we further consider the combination of the proposed method with some pose recovery algorithm. In this way, it would be possible to train the model by using action sequences taken from a MoCap dataset and classify sequences recovered from videos by means of the pose recovery algorithm. This would make action recognition from 2D video sequences also possible. Finally, we are currently considering automated ways for the segmentation of motion sequences to sub-sequences of individual actions without prior knowledge of the actions preformed. This step is important for allowing the processing of sequences containing multiple actions with the method described in this work.

Acknowledgments This paper describes research done under the EU-FP7 ICT 247870 NIFTi project.

References

1. Aggarwal, J.K., Cai, Q.: Human motion analysis: a review. Comput. Vis. Image Underst. **73**(3), 428–440 (1999)
2. Bahlmann, C., Haasdonk, B., Burkhardt, H.: On-line handwriting recognition with support vector machines-a kernel approach. In: International Workshop on Frontiers in Handwriting Recognition, pp. 49–54 (2002)
3. Baisero, A., Pokorny, F.T., Kragic, D., Ek, C.H.: The path kernel. In: Proceedings of the International Conference on Pattern Recognition Applications and Methods (2013)
4. CMU: Carnegie-mellon mocap database, http://mocap.cs.cmu.edu/ (2003)
5. Cuturi, M., Vert, J.P., Birkenes, O., Matsui, T.: A kernel for time series based on global alignments. Comput. Res. Repos. (2006)
6. Damianou, A.C., Titsias, M.K., Lawrence, N.D.: Variational gaussian process dynamical systems. In: Neural Information Processing Systems Conference, pp. 2510–2518 (2011)
7. Gong, D., Medioni, G.: Dynamic manifold warping for view invariant action recognition. In: International Conference on Computer Vision (2011)
8. Härdle, W., Simar, W.: Applied Multivariate Statistical Analysis. Springer, New York (2003)
9. Lawrence, N.D.: Gaussian process latent variable models for visualisation of high dimensional data. In: Neural Information Processing Systems Conference (2003)
10. Lawrence, N.D., Candela, J.Q.: Local distance preservation in the gp-lvm through back constraints. In: International Conference on Machine learning, pp. 513–520 (2006)
11. Li, Y., Fermüller, C., Aloimonos, Y., Ji, H.: Learning shift-invariant sparse representation of actions. In: International Conference on Computer Vision and Pattern Recognition, pp. 2630–2637 (2010)
12. Moeslund, T.B., Hilton, A., Krüger, V.: A survey of advances in vision-based human motion capture and analysis. Comput. Vis. Image Underst. **104**(2–3), 90–126 (2006)

13. Mordohai, P., Medioni, G.G.: Dimensionality estimation, manifold learning and function approximation using tensor voting. J. Mach. Learn. Res. **11**, 411–450 (2010)
14. Müller, M.: Information Retrieval for Music and Motion. Springer, Heidelberg (2007)
15. Müller, M., Röder, T., Clausen, M.: Efficient content-based retrieval of motion capture data. In: SIGGRAPH, pp. 677–685 (2005)
16. Muller, M., Roder, T., Clausen, M., Eberhardt, B., Krüger, B., Weber, A.: Documentation mocap database hdm05. Technical report CG-2007-2, Universität Bonn (2007)
17. Ntouskos, V., Papadakis, P., Pirri, F.: A comprehensive analysis of human motion capture data for action recognition. In: Proceedings of the International Conference on Computer Vision Theory and Applications, pp. 647–652 (2012)
18. Poggio, T.: Early vision: from computational structure to algorithms and parallel hardware. Comput. Vis. Graph. Image Process. **31**(2), 139–155 (1985)
19. Rasmussen, C., Williams, C.: Gaussian processes for machine learning. Adaptive Computation and Machine Learning. MIT, Cambridge (2006)
20. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. Science **290**(5500), 2323–2326 (2000)
21. Sheikh, Y., Sheikh, M., Shah, M.: Exploring the space of a human action. Int. Conf. Comput. Vis. **1**, 144–149 (2005)
22. Shimodaira, H., Noma, K., Nakai, M., Sagayama, S.: Dynamic time-alignment kernel in support vector machine. Neural Inf. Process. Syst. Conf. **2**, 921–928 (2001)
23. Taylor, G.W., Hinton, G.E., Roweis, S.T.: Modeling human motion using binary latent variables. In: Neural Information Processing Systems Conference, pp. 1345–1352 (2006)
24. Tenenbaum, J.B., Silva, V.D., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science (2000)
25. Titsias, M.K., Lawrence, N.D.: Bayesian gaussian process latent variable model. J. Mach. Learn. Res. Proc. Track **9**, 844–851 (2010)
26. Turaga, P.K., Chellappa, R., Subrahmanian, V.S., Udrea, O.: Machine recognition of human activities: a survey. IEEE Trans. Circuits Syst. Video Technol. **18**(11), 1473–1488 (2008)
27. Urtasun, R., Darrell, T.: Discriminative gaussian process latent variable model for classification. In: International Conference on Machine Learning, pp. 927–934 (2007)
28. Urtasun, R., Fleet, D.J., Fua, P.: 3d people tracking with gaussian process dynamical models. In: International Conference on Computer Vision and Pattern Recognition, pp. 238–245 (2006)
29. Urtasun, R., Fleet, D.J., Geiger, A., Popovic, J., Darrell, T., Lawrence, N.D.: Topologically-constrained latent variable models. In: International Conference on Machine Learning, pp. 1080–1087 (2008)
30. Waltisberg, D., Yao, A., Gall, J., Van Gool, L.: Variations of a hough-voting action recognition system. In: International conference on Pattern Recognition, pp. 306–312 (2010)
31. Wang, J.M., Fleet, D.J., Hertzmann, A.: Gaussian process dynamical models. Neural Inf. Proc. Syst. Conf. **18**, 1441–1448 (2006)
32. Yao, A., Gall, J., Fanelli, G., Gool, L.V.: Does human action recognition benefit from pose estimation? In: British Machine Vision Conference, pp. 67.1–67.11 (2011)
33. Yao, A., Gall, J., Gool, L.J.V.: A hough transform-based voting framework for action recognition. In: International Conference on Computer Vision and Pattern Recognition, pp. 2061–2068 (2010)
34. Zhang, X., Fan, G.: Joint gait-pose manifold for video-based human motion estimation. In: European Conference on Computer Vision, pp. 47–54 (2011)

Graph Cut Based Segmentation of Predefined Shapes: Applications to Biological Imaging

Emmanuel Soubies, Pierre Weiss and Xavier Descombes

Abstract We propose an algorithm to segment 2D ellipses or 3D ellipsoids. This problem is of fundamental importance in various applications of cell biology. The algorithm consists of minimizing a *contrast invariant* energy defined on sets of non overlapping ellipsoids. This highly non convex problem is solved by combining a stochastic approach based on marked point processes and a graph-cut algorithm that selects the best admissible configuration. In order to accelerate the computing times, we delineate fast algorithms to assess whether two ellispoids intersect or not and various heuristics to improve the convergence rate.

Keywords Nuclei segmentation · 2D and 3D images · Graph-cuts · Marked point processes · Ellipses and ellipsoids · Multiple objects detection · Multiple birth and cut · Bio-imaging

1 Introduction

Cell or nuclei segmentation in 2D and 3D is a major challenge in bio-medical imaging. New microscopes provide images at higher resolutions, deeper into biological tissues, leading to an increasing need for automatic cell delineation. This task may be easy in certain imaging modalities where images are well resolved and contrasted, but it remains mostly unresolved in emerging fluorescent microscopes dedicated to live imaging such as confocal, bi-photon, or selective plane illumination microscopes. These modalities suffer from multiple degradations such as light attenuation in the sample, photo bleaching, heavy noise and spatially varying blur that make the segmentation task hard even for human experts.

E. Soubies (✉) · X. Descombes
INRIA/I3S/IBV, MORPHEME Team, Sophia-Antipolis, France
e-mail: esoubies@gmail.com

X. Descombes
e-mail: xavier.descombes@inria.fr

P. Weiss
ITAV-USR3505, Université de Toulouse, CNRS, Toulouse, France
e-mail: pierre.armand.weiss@gmail.com

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_10

Our aim in this work is to propose a segmentation algorithm robust to such situations. Since images are heavily deteriorated, standard methods aiming at finding contours based on a sole regularity assumption such as active contours or Mumford-Shah derivatives fail for the segmentation. This observation led us to introduce strong shape priors: cells are modelled as ellipses or ellipsoids that should fit the image contents. Unfortunately, adding geometrical constraints makes the optimization problems highly non convex and appeal for the development of new global optimization methods.

Following recent works [5, 6, 9], we use randomized algorithms that allow to escape from local minima. These algorithms are based on marked point processes. The Marked Point Process (MPP) approach [1, 7] consists in estimating a configuration of geometric objects (in our case ellipses or ellipsoids) whose number, location and shape are unknown. It has proved to be very efficient in numerous image analysis applications as it allows the combination of radiometric information with strong geometrics constraints on the objects but also at a global scale. Defined by a density against the Poisson process measure, its main advantage is to consider a random number of objects and can be considered as an extension of the Markov Random Field approach. A review of this approach and its applications can be found in [5].

The objects are defined on a state space $\chi = I \times M$ by their location and their marks (i.e. geometric attributes). The associated marked point process X is a random variable whose realisations are random configurations of objects. Considering a Gibbs process, the modeling consists of an energy construction. Similarly to the Bayesian framework, this energy can be written as the sum of a data term and a prior. In this paper we consider a pairwise interactions prior that forbids intersections between objects. Once the model defined, the solution is obtained by minimizing the energy. This energy being highly non-convex requires stochastic dynamics, such as MCMC methods, to be minimized. The Reversible Jump MCMC embeded in a simulated annealing framework is a natural candidate for this task [10]. However, in case of simple constraints such as non overlap, the recently proposed multiple birth and death algorithm is preferable [6]. To avoid the fastidious calibration of annealing parameters, we propose to revisit the combination of the multiple births principle with the graph cut paradigm proposed by [9].

The paper is organized as follows. We formalize the segmentation problem as a minimization problem in Sect. 2. Section 3 begins by a global algorithm description and is followed by a precise description of each algorithm step. We finish by presenting numerical results in Sect. 4.

2 Problem Statement

Figure 1 contains typical examples of images encountered in biology. It is readily seen from these images that most nuclei contours can be well approximated by ellipses or ellipsoids, at least at a coarse scale. Moreover these nuclei cannot overlap due to obvious physical considerations. We thus formulate our segmentation problem as

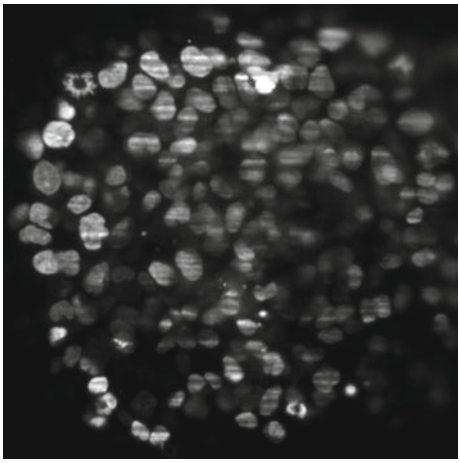


Fig. 1 Example of a SPIM image (*Multicellular tumor spheroid*)

that of finding a set of non overlapping ellipsoids that fit the image contents. We formalize this statement in the latter.

Let $\mathcal{C}_n,\ n \in \mathbb{N}$ denote the set of configurations containing n objects that do not overlap. An element $\mathbf{x} \in \mathcal{C}_n$ is a set of n non overlapping objects. Since the number of nuclei in the configuration is unknown, we aim both at finding this number n^* and the best configuration $\mathbf{x} \in \mathcal{C}_{n^*}$ with respect to a certain data fidelity term $f(\mathbf{x})$. Our optimization problem can thus be formulated as follows. Let

$$g(n) = \min_{\mathbf{x} \in \mathcal{C}_n} f(\mathbf{x})$$

denote the minimum value of f in the set $\mathcal{C}_n$. We wish to find both

$$n^* = \arg\min_{n \in \mathbb{N}} g(n)$$

and

$$\mathbf{x}^* = \arg\min_{\mathbf{x} \in \mathcal{C}_{n^*}} f(\mathbf{x}).$$

By convention, we assume that $\mathcal{C}_0 = \emptyset$ and that $\min_{\mathbf{x} \in \mathcal{C}_0} f(\mathbf{x}) = 0$. The data term f should thus be negative for configurations that are likely to represent the nuclei parameters and positive otherwise. We detail how the ellipses are parametrized and the construction of such a function in the following paragraphs.

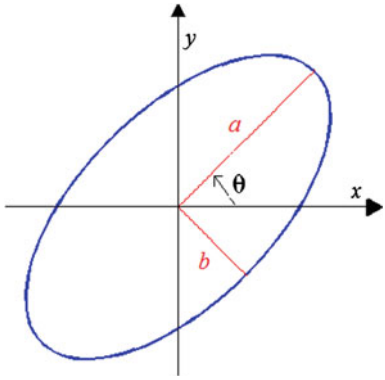


Fig. 2 Parameters of the ellipse

2.1 Object Modelling

In two dimensions, ellipses are parameterized using five parameters (see Fig. 2):

- $(x, y) \in \Omega$: center coordinates which should belong to the image domain Ω.
- $\theta \in [0, 2\pi[$: angle with the horizontal direction.
- $0 < \lambda_- < b < a < \lambda_+$: describe the ellipses minor and major axes size. λ_- and λ_+ are user defined parameters that describe the nuclei maximal size and ellipticity.

In three dimensions, nuclei are parameterized using nine parameters:

- $(x, y, z) \in \Omega$: center coordinates.
- $\phi, \theta, \psi \in [0, 2\pi[^3$: Euler angles to define the ellipsoids orientations.
- $0 < \lambda_- < c < b < a < \lambda_+$: axes lengths.

Overall, it can be seen that objects belong to a state space χ defined as a parallelepiped:

$$\chi = \Omega \times [0, 2\pi[\times [\lambda_-, \lambda_+]^2 \tag{1}$$

in 2D and

$$\chi = \Omega \times [0, 2\pi[^3 \times [\lambda_-, \lambda_+]^3 \tag{2}$$

in 3D.

In this paper, the objects are denoted ω and their boundary is denoted $\partial\omega$.

2.1.1 Data Term

Let $u : \Omega \to \mathbb{R}$ denote a grayscale image. In order to define the data term $f(\mathbf{x})$, we associate an elementary energy $U_d(\omega, u)$ to each element $\omega \in \mathbf{x}$ and set:

$$f(\mathbf{x}) = \sum_{\omega \in \mathbf{x}} U_d(\omega, u). \tag{3}$$

The function $U_d(\omega, u) \in [-1, 1]$ should be negative if the object ω is well positioned on the image and positive otherwise.

In fluorescence microscopy, nuclei are usually characterized by bright region surrounded by a dark background since they are stained or genetically modified in order to express a fluorescent protein. Unfortunately, their radiometry is not constant due to local bleaching or light attenuation in the deepest layers. We thus need to construct an energy that is *contrast invariant*, meaning that local modifications of the radiometry shall not affect the energy. Such an energy can be constructed easily by considering the normal to the image level lines $\frac{\nabla u}{|\nabla u|}$ where ∇u denotes the usual gradient in $\mathbb{R}^d$ and $|\nabla u|$ denotes the gradient magnitude in the standard Euclidean norm. This tool is well known to be contrast invariant. Let us define an energy U for a given object ω as:

$$U(\omega) = \frac{1}{|\partial\omega|} \int_{\partial\omega} \langle \frac{\nabla u(x)}{\sqrt{|\nabla u(x)|^2 + \epsilon^2}}, n(x) \rangle dx \tag{4}$$

where $\langle \cdot, \cdot \rangle$ denotes the standard scalar product, $|\partial\omega|$ denotes the length of the object boundary, $n(x)$ denotes the outward normal to ω at location $x \in \partial\omega$ and ϵ is a regularization parameter that discard faint transitions. The behavior of this energy is illustrated on Fig. 3. Overall, it does what is expected, but as can be seen on the illustration (b) and (d) in Fig. 3, badly located ellipses might have a negative energy and be kept in the final configuration. It is thus necessary to modify U in order to promote well located objects only. A simple way to do so consists in setting:

$$U_d(\omega, u) = \psi(U(\omega), s)$$

where $s \in]-1, 0]$ is an acceptance threshold for the objects and

$$\psi(\alpha, s) = \min(\frac{1}{s+1}\alpha - \frac{s}{s+1}, 1).$$

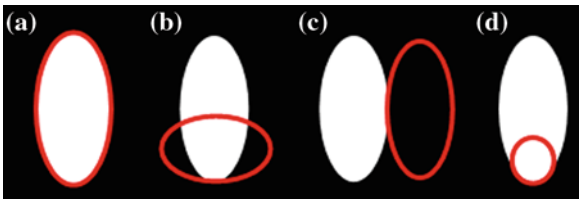


Fig. 3 Behavior of the energy given by Eq. (4). The corresponding energy values are: U = −1 (**a**). U = −0.1 (**b**). U = −0.2 (**c**). U = −0.5 (**d**)

Other data terms based on the contrast between the object interior and the background as presented by [9] (in dimension 2) could also be used but present two drawbacks: first they require to compute an integral over the interior of the domain while the proposed approach consist in computing a boundary integral which is faster. Second, such measures might be inaccurate in the case of very dense media, where the background can be difficult to extract. Finally our measure is contrast invariant, which is central for the targeted applications.

3 Multiple Birth and Cut Algorithm

The Multiple Birth and Cut algorithm (MBC) has been proposed by [9] for counting flamingos in a colony. In this section, we describe the different steps of the MBC algorithm (Algorithm 1).

The main idea consists in generating two random configurations of non-overlapping objects $\mathbf{x}$ and $\mathbf{x}'$ (birth step) and then keep the subset of objects in $\mathbf{x} \cup \mathbf{x}'$ that minimizes f (cut step). This process is iterated and decreases f at each iteration. The cut step can be performed efficiently using a Graph Cut algorithm [4, 11]. We describe this algorithm more formally below:

Algorithm 1: Multiple Birth & Cut algorithm.

Require: N
1: Generate a configuration $\mathbf{x}_{[0]}$ with Algorithm 2
2: $n \longleftarrow 0$
3: **while** (Not converged) **do**
4: Generate of a new configuration $\mathbf{x}'$ using Algorithm 2.
5: $\mathbf{x}_{[n+1]} \longleftarrow Cut(\mathbf{x}_n \cup \mathbf{x}')$
6: $n \longleftarrow n + 1$
7: **end while**

Interestingly, this algorithm contains only one parameter N (the number of objects generated in a configuration). We observed experimentally that this parameter might affect slightly the speed of convergence but not the segmentation accuracy. This algorithm is thus much easier to tune than more standard RJMCMC based dynamics.

3.1 Birth Step

A new configuration $\mathbf{x}'$ of non-overlapping objects is generated. Note that only objects which are in the same configuration have to respect the non-overlapping constraint, but two objects in different configurations can intersect as can be seen on Fig. 4.

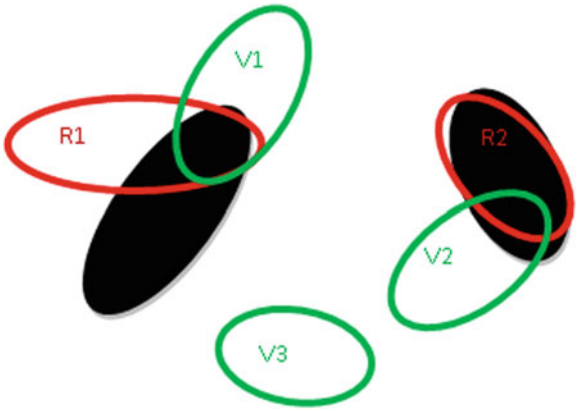


Fig. 4 Two configurations on an image (the *black ellipses* are the object to detect)

The birth step is detailed in Algorithm 2. The fourth step of this algorithm can be efficiently implemented using a lookup table and the fast intersection algorithm proposed in the latter.

Algorithm 2: Birth step.

Require: N, n_{max}.
1: Set $k = 0, n = 0, \mathbf{x}' = \emptyset$.
2: **while** $k < N$ and $n < n_{max}$ **do**
3: Construct an object ω' by generating a random vector uniformly in χ.
4: If ω' intersects an object in $\mathbf{x}'$, set $n = n + 1$ and go back to 3.
5: Otherwise set $\mathbf{x}' = \mathbf{x}' \cup \{\omega'\}, k = k + 1, n = 0$ and go back to 3.
6: **end while**

3.2 Cut Step

This step consists in selecting the best configuration of non-overlapping objects in $(\mathbf{x}_{[n]} \cup \mathbf{x}')$. To perform this optimization, a weighted graph is constructed. The nodes of this graph are the objects ω of the two configurations $\mathbf{x}_{[n]}$ and $\mathbf{x}'$. This graph also possesses two special nodes, the source '**s**' and the sink '**t**'. The weights should belong to $[0, 1] \cup \{+\infty\}$ and a weight equal to 1 should be associated to a well positioned object. It is thus necessary to redefine the data term $U_d(\omega, u)$ by:

$$W(\omega) = (1 - U_d(\omega, u))/2.$$

which will be equal to 1 for $U_d(\omega, u) = -1$ and to 0 for $U_d(\omega, u) = 1$ what is expected.

3.2.1 Graph Construction

Each object of the configuration $(\mathbf{x}_{[n]} \cup \mathbf{x}')$ is linked to the source and the sink. The difference between the objects $\omega_i \in \mathbf{x}_{[n]}$ and the objects $\omega_j \in \mathbf{x}'$ is that the objects $\omega_i \in \mathbf{x}_{[n]}$ are linked to the source with a weight equal to the data energy $W(\omega)$ and to the sink with a weight equal to $1 - W(\omega)$, while it is the reverse for the objects $\omega_j \in \mathbf{x}'$.

The weights associated to edges linking two objects are non zero only when two objects intersect. If $\omega_1 \in \mathbf{x}_{[n]}$ (current configuration) intersects with $\omega_2 \in \mathbf{x}'$ (new configuration), the link from ω_1 to ω_2 is set to ∞ and the link from ω_2 to ω_1 is set to zero.[1] This ensures that the cut step generates an admissible configuration (with no overlapping objects). Figure 5 summarises the graph construction of the configurations on Fig. 4. The nuclei to detect are represented by black ellipses.

3.2.2 Cut

Once the graph is constructed, we perform a cut that consists in partitioning the vertices into two disjoint subsets. One contains the source and the other the sink. The cut realized is the one with minimal cost (the one minimizing the sum of the weights of the removed edges).

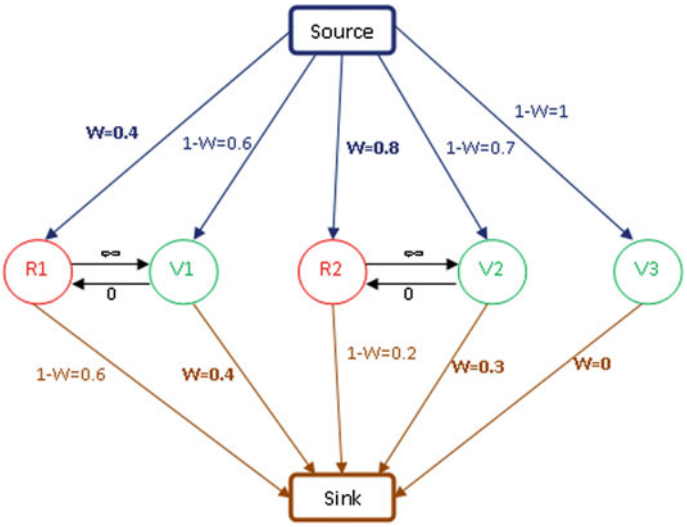


Fig. 5 Graph corresponding to the Fig. 4

[1] When two objects intersect the link affected by a weight of ∞ is always the link from the object of the current configuration to the object of the new configuration.

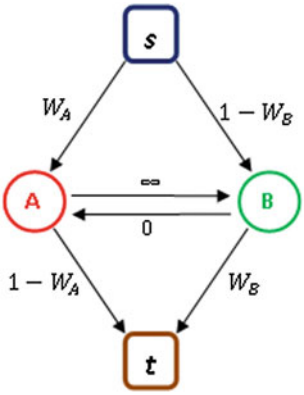


Fig. 6 Graph associated to two overlapping objects

After the cut step, if $\omega_i \in \mathbf{x}_{[n]}$ is in the sub-graph containing the source, we keep it, otherwise we remove it. On the contrary the objects $\omega_j \in \mathbf{x}'$ are kept only if they belong to the sub-graph that contains the sink. This difference of interpretation between the two configurations combined with the different weights to the source and the sink, ensure that in case where an object of $\mathbf{x}_{[n]}$ and an object of $\mathbf{x}'$ intersect, only one can be kept.

Let $\omega_A \in \mathbf{x}_{[n]}$ and $\omega_B \in \mathbf{x}'$ be two overlapping objects. Figure 6 presents the associated graph and Fig. 7 shows the four possible cuts of this graph.

As the cut step consists in finding the cut with minimal cost, the situation presented in 7(d) can not occur since the cost is equal to ∞. Furthermore, case 7(c) can occur and then, for two overlapping objects, either one is kept (7(a) and 7(b)) or both are removed (7(c)).

Remark 1 In 7(c) and 7(d) the vertices are well partitioned into two disjoint subsets since there is no subset of edges in the resulting graph which allows to go from the source '**s**' to the sink '**t**'.

The cut step is implemented using the graph-cut code developed by Boykov and Kolmogorov in [3, 4, 11].

3.3 A Fast Determination of Ellipses Intersection

One of the proposed algorithm bottleneck is the fast determination of whether two ellipsoids intersect or not. In this section, we present a fast algorithm to answer that question and prove theoretically that only a few arithmetic operations suffice to provide the answer with a low error rate.

Let ω be an ellipse or an ellipsoid. It can be defined using a quadratic function Q_ω as $\omega = \{x \in \mathbb{R}^d, Q_\omega(x) = 1\}$. The quadratic function Q can be defined by:

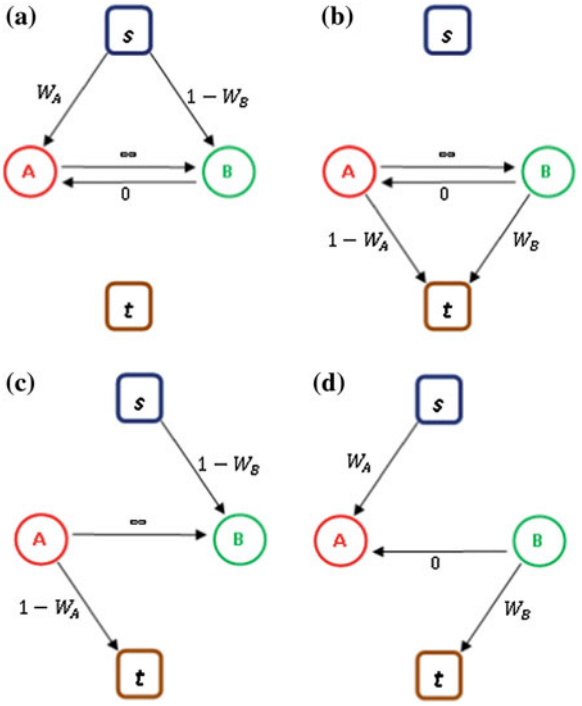


Fig. 7 Four possible cuts of the graph corresponding to the Fig. 6. Keep w_A, remove w_B (**a**). $Cost = 1 - W_A + W_B$. Keep w_B, remove w_A. $Cost = 1 - W_B + W_A$ (**b**). Remove w_A and w_B. $Cost = W_A + W_B$ (**c**). Keep w_A and w_B. $Cost = \infty$ (**d**)

$$Q_\omega(x) = \langle A(x - c), (x - c) \rangle \tag{5}$$

where c denotes the object center and A is positive definite matrix defined by:

$$A = P^{-1} D P = P^T D P.$$

where P is a rotation matrix and D is a positive diagonal matrix. In 2D, P is defined by:

$$P = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}$$

and

$$D = \begin{pmatrix} \frac{1}{a^2} & 0 \\ 0 & \frac{1}{b^2} \end{pmatrix}.$$

In 3D, the notation become cumbersome and we leave them to the reader.

Let ω_1 and ω_2 be two ellipses or ellipsoids. In order to know whether they intersect or not, we can find the minimal level set of Q_{ω_2} which intersects the boundary of ω_1. If this level set is associated to a value greater than 1, the ellipses are separated, otherwise they overlap. This idea can be formulated as the following minimization problem:

$$\min_{x\in\mathbb{R}^d,\, Q_{\omega_1}(x)\leq 1} Q_{\omega_2}(x) \tag{6}$$

This problem consists of minimizing a quadratic function over convex set. Projected descent methods can thus be used. Unfortunately, there exists no closed form solution to the problem of projection of a point on an ellipse. We thus need to simplify the constraint set:

$$\begin{aligned}
&\min_{Q_{\omega_1}(x)\leq 1} Q_{\omega_2}(x) \\
&\quad = \min_{\langle A_1(x-c_1),(x-c_1)\rangle\leq 1} \langle A_2(x-c_2),(x-c_2)\rangle \\
&\quad = \min_{\langle \sqrt{A_1}(x-c_1),\sqrt{A_1}(x-c_1)\rangle\leq 1} \langle A_2(x-c_2),(x-c_2)\rangle \\
&\quad = \min_{\|y-\sqrt{A_1}c_1\|_2^2\leq 1} \langle A_2(A_1^{-\frac{1}{2}}y-c_2),(A_1^{-\frac{1}{2}}y-c_2)\rangle.
\end{aligned}$$

where $y = \sqrt{A_1}x$. In this reformulation, the constraint set $Y = \{y \in \mathbb{R}^d, \|y - \sqrt{A_1}c_1\|_2^2 \leq 1\}$ is a simple l^2-ball and the function $F(y) = \langle A_2(A_1^{-\frac{1}{2}}y - c_2),(A_1^{-\frac{1}{2}}y - c_2)\rangle$ is a strongly convex differentiable function. We can thus use a projected gradient descent that writes:

Algorithm 3: Detection of overlapping ellipsoids.

Require: Q_{ω_1}, Q_{ω_2}, $\epsilon > 0$.
1: Set $k = 0$, $y_0 = \frac{c_1+c_2}{2}$.
2: Set $\mu = \frac{b_1^2}{a_2^2}$, $L = \frac{a_1^2}{b_2^2}$.
3: Set $\tau = \frac{2}{\mu+L}$.
4: **while** $\|y_{k+1} - y_k\| \geq \epsilon$ **do**
5: $\quad y_{k+\frac{1}{2}} = y_k - \tau\nabla F(y_k)$.
6: $\quad y_{k+1} = \Pi_Y\left(y_{k+\frac{1}{2}}\right)$.
7: $\quad k = k + 1$.
8: **end while**
9: If $F(y_k) >= 1$ return 0 (the ellipsoids do not intersect with high probability).
10: If $F(y_k) < 1$ return 1 (the ellipsoids intersect).

Let y^* denote the solution of the above problem. The previous algorithm comes with the following guarantees:

Theorem 1 After k iterations, y_k satisfies:

$$F(y_k) - F(y^*) \leq \frac{\mu}{2}\|y_0 - y^*\|_2^2 \left(\frac{Q_F - 1}{Q_F + 1}\right)^{2k}$$

$$\|y_k - y^*\|_2^2 \leq \|y_0 - y^*\|_2^2 \left(\frac{Q_F - 1}{Q_F + 1}\right)^{2k}$$

where

$$Q_F = \frac{a_1^2}{b_2^2}\frac{a_2^2}{b_1^2} \leq \frac{\lambda_+^4}{\lambda_-^4}.$$

Proof The Hessian of F is $H_F(y) = 2A_1^{-\frac{1}{2}} A_2 A_1^{-\frac{1}{2}}$. Since A_1 and A_2 are products of orthogonal and diagonal matrices ($A = P^T D P$), the eigenvalues of $H_F(y)$ can be easily bounded:

$$\lambda_{min}[H_F(y)] \geq \frac{b_1^2}{a_2^2} \qquad \lambda_{max}[H_F(y)] \leq \frac{a_1^2}{b_2^2}$$

The function F is thus μ-strongly convex with $\mu \geq \frac{b_1^2}{a_2^2}$ and its gradient is L-Lipschitz with $L \leq \frac{a_1^2}{b_2^2}$. Using standard convergence theorems in convex analysis [2], we obtain the announced result. □

The conditioning number Q_F depends solely on the ratio between the major axis and the minor axis sizes and not on the dimension d. This algorithm will thus be as efficient in 3D as in 2D. For two circles the ratio Q_F is equal to $\frac{a}{b} = 1$ and the algorithm provides the exact answer after one iteration. For elliptic ratios of 2, $Q_f = 16$ and in the worst case, after 18 iterations, the algorithm returns a point y^k that is 100 times closer to the intersection than y_0. We also tested an accelerated algorithm by [12], where the convergence rate is of order $\left(\frac{\sqrt{Q_F}-1}{\sqrt{Q_F}+1}\right)^{2k}$ but it did not improve the computing times.

In our problems the ratio between a and b is always less than 2 and the algorithm usually converges in just a few iterations (2–10 depending on the problem).

3.4 Acceleration by Local Perturbations

When the objects variability is important, the state space size increases and affects the convergence speed of the MBC algorithm. This problem is particulary important in 3D since ellipsoids are defined by 9 parameters instead of 5 for the 2 dimensional case.

In order to improve the convergence speed, [8] proposed to insert a *selection phase* in the birth step. This selection phase consists in generating a dense configuration of objects at similar locations and to keep the best ones using Belief Propagation in order to form the new configuration.

In this paper, we propose another heuristic in order to increase the convergence speed. We propose to alternate between two different kinds of birth steps. The first one is that proposed in algorithm 2. The second one consists in perturbating locally the current configuration. This principle mimics the proposition kernels used in RJMCMC algorithms [13]. The idea behind this modification is that after a while, most objects are close to their real location and that local perturbations may allow much faster convergence than fully randomized generation. This algorithm is described in details in Algorithm 4.

Algorithm 4: MBC algorithm with local perturbations.

Require: N
Generate a configuration $\mathbf{x}_{[0]}$ using Algorithm 2.
$n \longleftarrow 0$
while (Not converged) **do**
  Generate a uniformly distributed random number $r \in [0, 1]$.
  if $r <$ **p then**
    Generate a new configuration $\mathbf{x}'$ using Algorithm 2.
  else
    Generate a new configuration $\mathbf{x}'$ using Algorithm 5.
  end if
  $\mathbf{x}_{[n+1]} \longleftarrow Cut(\mathbf{x}_n \cup \mathbf{x}')$
  $n \longleftarrow n + 1$
end while

3.4.1 Local Perturbations

A given object ω in $\mathbf{x}_{[n]}$ is described by a set of parameters $\lambda \in \chi$ (see Eqs. 1 and 2). We generate the new object ω' by setting its parameters $\lambda' = \lambda + z$ where z is the realization of a random vector Z distributed uniformly in χ_ϵ where:

$$\chi_\epsilon = [-\delta_{xy}, \delta_{xy}]^2 \times [-\delta_{ab}, \delta_{ab}]^2 \times [0, 2\pi[$$

in 2D and

$$\chi_\epsilon = [-\delta_{xyz}, \delta_{xyz}]^3 \times [-\delta_{abc}, \delta_{abc}]^3 \times [0, 2\pi[^3$$

in 3D.

The value of the different δ describes the perturbation extent. We observed that small values accelerates the convergence speed.

Algorithm 5: Birth step with local perturbation.

Require: $\mathbf{x}_{[n]}$.
1: **while** $k < size(\mathbf{x}_{[n]})$ **do**
2: Construct an object ω' by local perturbation of $\omega_k \in \mathbf{x}[n]$.
3: If ω' intersects an object in $\mathbf{x}'$, set $k = k + 1$ and go back to 2.
4: Otherwise set $\mathbf{x}' = \mathbf{x}' \cup \{\omega'\}$, $k = k + 1$ and go back to 2.
5: **end while**

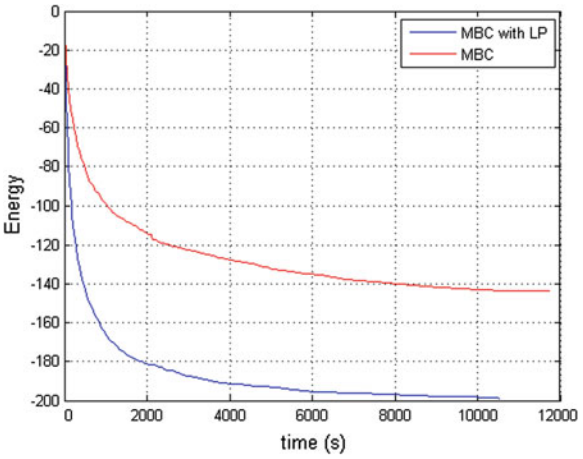


Fig. 8 Comparison of the MBC and MBC with LP algorithms

3.4.2 Comparison of the Convergence Speed

We have tested this method in order to compare the speed of convergence of the MBC algorithm and the MBC algorithm with local perturbation. Figure 8 presents the energy evolution with respect to time for both MBC and MBC with local perturbations (denoted MBC with LP) on the same image (the 3D nuclei of Drosophila embryo). The segmentation result is presented on Fig. 13 (image size $700 \times 350 \times 100$). These results show that the MBC with LP algorithm strongly improve the MBC algorithm.

4 Results

In this section, we present some practical results in 2D and 3D. Figure 9 shows the segmentation result on a Drosophila embryo obtained using SPIM imaging. This is a rather easy case, since nuclei shapes vary little. The images are impaired by various defects: blur, stripes and attenuation. Despite this relatively poor image quality, the segmentation results are almost perfect. The computing time is 5 min using a c++ implementation. The image size is 700×350.

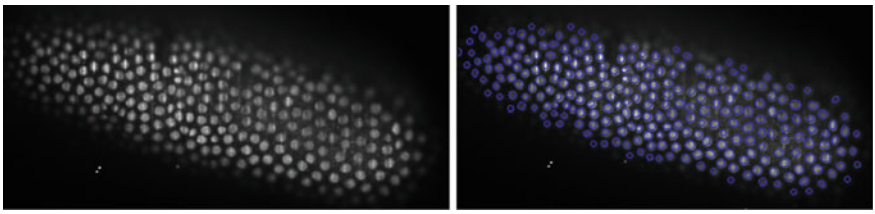


Fig. 9 *Left* Drosophila embryo. *Right* result of the proposed 2D segmentation algorithm

Figure 10 presents a more difficult case, where the image is highly deteriorated. Nuclei cannot be identified in the image center. Moreover, nuclei variability is important meaning that the state space size χ is large. Some nuclei are in mitosis (see e.g. top-left). In spite of these difficulties, the MBC algorithm provides acceptable results. They would allow to make statistics on the cell location and orientation, which is a major problem in biology. The computing times for this example is 30 min.

Nuclei segmentation is a major open problem with a large number of other applications. In Fig. 11, we attempt to detect the spermatozoon heads. The foreseen application is tracking in order to understand their collective motion. Figure 12 presents a multicellular spheroid, an in vitro model mimicking microtumor region organization, surrounded by a circle of high aspect ratio pillars made in a soft material by advanced microfabrication processes. The aim of this experiment is to determine the displacement of the pillars induced by the spheroid dynamics. To address this question, the precise detection of the contours of the top of the pillars is required for this quantitative measurement.

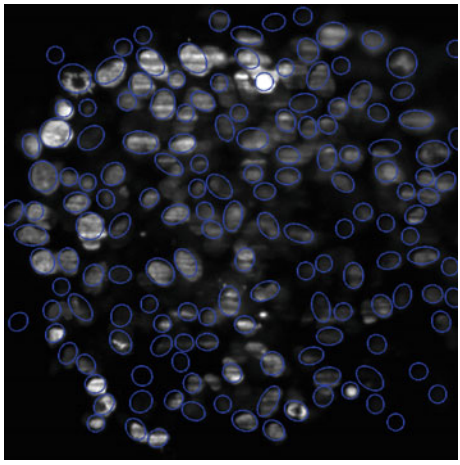


Fig. 10 2D segmentation of a multicellular tumor spheroid (Fig. 1)

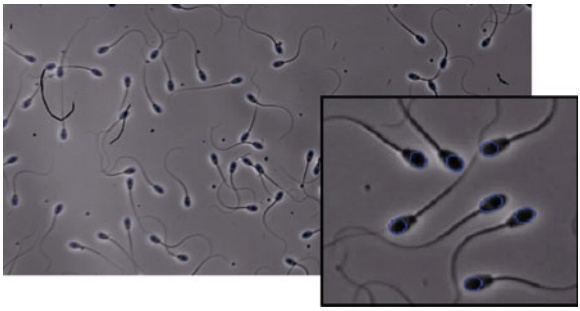


Fig. 11 Segmentation of a spermatozoon colony (5 min). Image size: 2,000 × 1,024

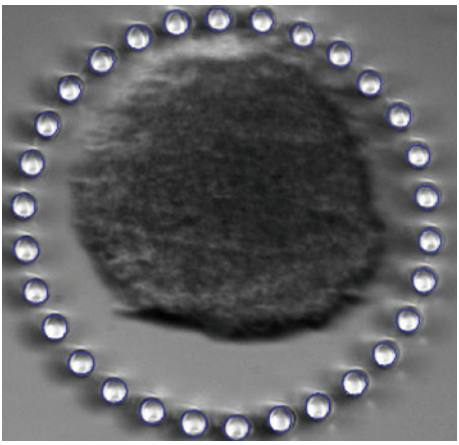


Fig. 12 Micro pillars detection (<1 min). Image size: 840 × 800

3D results are presented in Figs. 13 and 14. For the Drosophila embryo, the segmentation is very close to what a human expert would do. The computing times are 2 hours and the image size is 700 × 350 × 100. The curves in Fig. 8 correspond to this image.

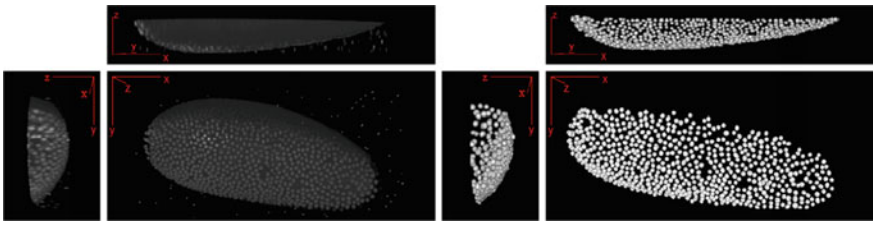


Fig. 13 *Left* 3D Drosophila embryo nuclei. *Right* Result of the proposed 3D segmentation algorithm

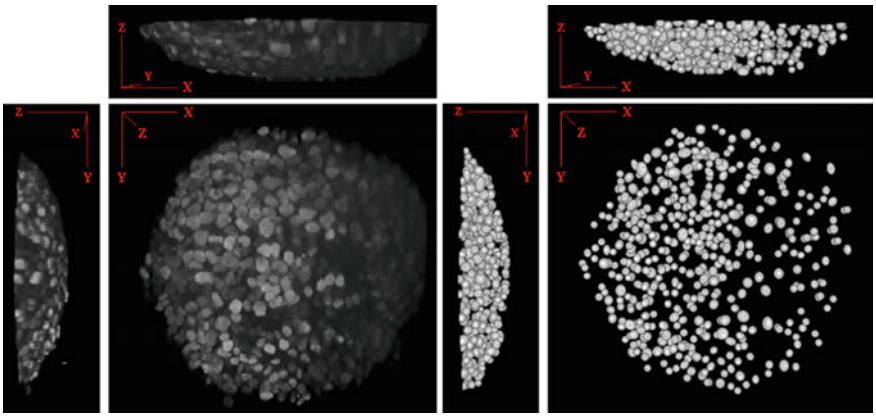


Fig. 14 *Left* 3D multicellular tumor spheroid. *Right* Result of the proposed 3D segmentation algorithm

The spheroid segmentation presented in Fig. 14 is less precise than the previous ones due to an important cell variability and to the fact that the images are extremly blurry in the Z direction. For that case, image restoration algorithms and the design of new energies robust to strong perturbations seem important.

5 Conclusions

We proposed a detection algorithm capable of identifying sets of non overlapping ellipses or ellipsoids. Interestingly, this algorithm contains only parameters that are related to physical properties of the underlying objects (e.g. nuclei variability in size and ellipticity) and is thus easy to apply for any person working in fields such as biological imaging. We presented the wide applicability of this algorithm for 2D and 3D images. Even in hard cases with contrast loss and high noise, the algorithm manages to find most nuclei due to contrast invariant energies.

Future work will include a quantitative evaluation of the algorithm efficiency with gold standards. We are also investigating the possibility to encode more complex interactions between objects to handle cases where the normal to the image level lines do not provide sufficient information for ellipsoid fitting.

Acknowledgments This work was partially funded by the Mission pour l'interdisci- plinarité from CNRS, Région Midi Pyrénées, PEPII CASPA3D, ANR SPHIM3D and ANR MOTIMO. The authors wish to thank F. Malgouyres and J. Fehrenbach for interesting discussions. They also thank V. Lobjois, C. Emery, J. Thomazeau, P. Escande and B. Ducommun for their help in this project. They thank L. Aoun and C. Vieu for providing images and interesting discussions regarding micro pillars detection. They thank all the ITAV staff for their warm welcome in a biology laboratory.

References

1. Baddeley, A., Van Lieshout, M.: Stochastic geometry models in high-level vision. J. Appl. Stat. **20**(5–6), 231–256 (1993)
2. Bertsekas, D.: Nonlinear programming (1999)
3. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans. Pattern Anal. Mach. Intell. **26**(9), 1124–1137 (2004)
4. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **23**, 1222–1239 (2001)
5. Descombes, X.: Stochastic Geometry for Image Analysis. Wiley/Iste, x. descombes edition, London (2011)
6. Descombes, X., Minlos, R., Zhizhina, E.: Object extraction using a stochastic birth-and-death dynamics in continuum. J. Math. Imaging Vis. **33**(3), 347–359 (2009)
7. Dong, G., Acton, S.: Detection of rolling leukocytes by marked point processes. J. Electron. Imaging **16**, 033013 (2007)
8. Gamal-Eldin, A., Descombes, X., Charpiat, G., Zerubia, J.: A fast multiple birth and cut algorithm using belief propagation. In: 18th IEEE International Conference on Image Processing (ICIP), pp. 2813–2816. IEEE (2011)
9. Gamal Eldin, A., Descombes, X., Charpiat, G., Zerubia, J., et al.: Multiple birth and cut algorithm for multiple object detection. J. Multimed. Process. Technol.(2012)
10. Green, P.J.: Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. Biometrika **82**, 711–732 (1995)
11. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **26**, 65–81 (2004)
12. Nesterov, Y.: Introductory lectures on convex optimization: A basic course, vol. 87. Springer, Berlin (2004)
13. Perrin, G., Descombes, X., Zerubia, J.: A marked point process model for tree crown extraction in plantations. In: IEEE International Conference on Image Processing, ICIP 2005, vol. 1, pp. I–661. IEEE (2005)

Artificial Neural Network Modeling of Relative Humidity and Air Temperature Spatial and Temporal Distributions Over Complex Terrains

Kostas Philippopoulos, Despina Deligiorgi and Georgios Kouroupetroglou

Abstract In this work we present a methodological approach of applying Artificial Neural Networks (ANN) for modeling of both the air temperature (AT) and relative humidity (RH) spatial and temporal distributions over complex terrains. A number of implementation issues are discussed, along with their relative advantages and limitations. Moreover, after the introduction of a set of metrics, the accuracy of the evaluation of ANN based spatial and time series AT and RH modeling in the case of a specific region is examined, by applying a number of alternative feed forward ANN topologies. The Levenberg-Marquardt back propagation algorithm was used for the ANNs training in the temporal forecasting of AT and RH, with the optimum architecture being the one that minimizes the Mean Absolute Error on the validation set. The Radial Basis Function and the Multilayer Perceptrons non-linear Feed Forward ANNs schemes are compared for the spatial estimation of AT and RH. We found that the spatial and temporal AT and RH variability over complex terrains can be modeled efficiently by ANNs.

Keywords Artificial neural networks · Relative humidity modeling · Air temperature modeling · Spatial interpolation · Time-series forecasting

1 Introduction

Air temperature (AT) and relative humidity (RH) measurements in high resolution time series are available only at limited stations because meteorological data are generally recorded at specific locations and derived from different meteorological

K. Philippopoulos · D. Deligiorgi (✉)
Division of Environmental Physics and Meteorology, Department of Physics, National and Kapodistrian University of Athens, Athens, Greece
e-mail: despo@phys.uoa.gr

G. Kouroupetroglou
Division of Communication and Signal Processing, Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, Athens, Greece
e-mail: koupe@di.uoa.gr

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*, Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_11

stations with non-identical characteristics. Spatial interpolation approaches essentially transfer available information in the form of data from a number of adjacent irregular sites to the estimated sites. Thus, spatial interpolation methods are frequently used to estimate values of AT and RH data in locations where they are not measured. Various methods have been developed with the purpose to compare the performance of different traditional spatial interpolation methods for air temperature data [1, 2]. Accurate ambient temperature estimates are important not only in spatial but also in temporal scales. Air temperature time series forecasting is one of the most significant aspects in environmental research and in climate impact studies. Time series forecasts are valuable in the renewable energy industry, in agriculture for estimating potential hazards, and within an urban context, in air quality studies for assessing the risk of adverse health effects in the general population.

During the last few decades, there has been a substantial increase in the interest on Artificial Neural Networks (ANN). ANNs have been successfully adopted in solving complex problems in many fields. Essentially, ANNs provide a methodological approach in solving various types of nonlinear problems that are difficult to deal with using traditional techniques. Often, a geophysical phenomenon exhibits temporal and spatial variability, and is suffering from issues of nonlinearity, conflicting spatial and temporal scale and uncertainty in parameter estimation [3]. ANNs have been proved [4] to be flexible models that have the capability to learn the underlying relationships between the inputs and outputs of a process, without needing the explicit knowledge of how these variables are related.

Recently, numerous applications of ANNs to estimate air temperature data have been presented, e.g. in areas with sparse network of meteorological stations [5, 6], for the prediction of hourly [7], daily [8] and year-round air temperature [9] or room temperature [10], as well as for simulating the Heat Island [11].

In this work first we briefly present the theoretical background of ANN methodologies applicable to the field of AT and RH time series and spatial modeling. Next, we focus on implementation issues and on evaluating the accuracy of the aforementioned methodologies using a set of metrics in the case of a specific region with complex terrain at Chania, Crete Island, Greece. A number of alternative Feedforward ANN topologies are applied in order to assess the spatial and temporal RH and AT prediction capabilities in different time horizons.

2 ANN-Based Prediction Modeling

Artificial Neurons are Process Element (PE) that attempt to simulate in a simplistic way the structure and function of the real physical biological neurons. A PE in its basic form can be modelled as a nonlinear element that first sums its weighted inputs $x_1, x_2, x_3, \ldots, x_n$ (coming either from original data, or from the output of other neurons in a neural network) and then passes the result through an activation function Ψ (or transfer function) according to the formula:

$$y_j = \Psi\left(\sum_{i=1}^{n} x_i w_{ji} + \theta_j\right) \quad (1)$$

where y_j is the output of the artificial neuron, θ_j is an external threshold (or bias value) and w_{ji} are the weight of the respective input x_i which determines the strength of the connection from the previous PE's to the corresponding input of the current PE. Depending on the application, various non-linear or linear activation functions Ψ have been introduced [12, 13] like the: signum function (or hard limiter), sigmoid limiter, quadratic function, saturation limiter, absolute value function, Gaussian and hyperbolic tangent functions. Artificial Neural Networks (ANN) are signal or information processing systems constituted by an assembly of a large number of simple Processing Elements, as they have been described above. The PE of an ANN are interconnected by direct links called connections and cooperate to perform a Parallel Distributed Processing in order to solve a specific computational task, such as pattern classification, function approximation, clustering (or categorization), prediction (or forecasting or estimation), optimization and control. One of the main strength of ANNs is their capability to adapt themselves by modifying the interaction between their PE. Another important feature of ANNs is their ability to automatically learn from a given set of representative examples.

The architectures of ANNs can be classified into two main topologies: (a) Feed-forward multilayer networks (FFANN) in which feedback connections are not allowed and (b) Feedback recurrent networks (FBANN) in which loops exist. FFANNs are characterized mainly as static and memory-less systems that usually produce a response to an input quickly [14]. Most FFANNs can be trained using a wide variety of efficient conventional numerical methods. FBANNs are dynamic systems. In some of them, each time an input is presented, the ANN must iterate for a potentially long time before it produces a response. Usually, they are more difficult to train FBANNs compared to FFANNs.

FFANNs have been found to be very effective and powerful in prediction, forecasting or estimation problems [15]. Multilayer perceptrons (MLPs) and radial basis function (RBF) topologies are the two most commonly-used types of FFANNs. Essentially, their main difference is the way in which the hidden PEs combine values coming from preceding layers: MLPs use inner products, while RBF constitutes a multidimensional function which depends on the distance $r = \|x - c\|$ between the input vector x and the center c (where $\|\cdot\|$ denotes a vector norm) [28]. As a consequence, the training approaches between MLPs and RBF based FFANN is not the same, although most training methods for MLPs can also be applied to RBF ANNs. In RBF FFANNs the connections of the hidden layer are not weighted and the hidden nodes are PEs with a RBF, however the output layer performs a simple weighted summation of its inputs, like in the case of MLPs. One simple approach to approximate a nonlinear function is to represent it as a linear combination of a number of fixed nonlinear RBFs $\{z_i(x)\}$, according to (2):

$$\Phi(x) = \sum_{i=1}^{l} z_i(x)\, w_i \tag{2}$$

Typical choices for RBFs $z_i = F(\|x - c\|)$ are: piecewise linear approximations, Gaussian function, cubic approximation, multiquadratic function and thin plate splines.

A MLP FFANN can have more than one hidden layer. But theoretical research has shown that a single hidden layer is sufficient in that kind of topologies to approximate any complex nonlinear function [16, 17].

There are two main learning approaches in ANNs: (i) supervised, in which the correct results are known and they are provided to the network during the training process, so that the weights of the PEs are adjusted in order its output to match the target value and (ii) unsupervised, in which the ANN performs a kind of data compression, looking for correlation patterns between them and by applying clustering approaches. Moreover, hybrid learning (i.e. a combination of the supervised and unsupervised methodologies) has been applied in ANNs. Numerous learning algorithms have been introduced for the above learning approaches [14].

The introduction of the back propagation learning algorithm [18] to obtain the weight of a multilayer MLP could be regarded as one of the most significant breakthroughs for training ANNs. The objective of the training is to minimize the training mean square error E_{mse} of the ANN output compared to the required output for all the training patterns:

$$E_{mse} = \sum_{k=1}^{p} E_k = \frac{1}{2N} \sum_{j=Y} \sum_{k=1}^{p} \left(y_i - d_{kj}\right)^2 \tag{3}$$

where: E_k is the partial network error, p is the number of the available patterns and Y the set of the output PEs. The new configuration in time $t > 0$ is calculated as follows:

$$w_{ji}(k) = w_{ji}(k-1) - \alpha \frac{\partial E}{\partial w_{ji}} + \beta[w_{ji}(k-1) - w_{ji}(k-2)] \tag{4}$$

To speed up the training process, the faster Levenberg-Marquardt Back propagation Algorithm has been introduced [19]. It is fast and has stable convergence and it is suitable for training ANN in small-and medium-sized problems. The new configuration of the weights in the k+1 step is calculated as follows:

$$w(k+1) = w(k) - \left(J^T J + \lambda I\right)^{-1} J^T \varepsilon(k) \tag{5}$$

The Jacobian matrix for a single PS can be written as follows:

$$J = \begin{bmatrix} \frac{\partial \varepsilon_1}{\partial w_1} & \cdots & \frac{\partial \varepsilon_1}{\partial w_n} & \frac{\partial \varepsilon_1}{\partial w_0} \\ \vdots & & \vdots & \vdots \\ \frac{\partial \varepsilon_p}{\partial w_1} & \cdots & \frac{\partial \varepsilon_p}{\partial w_n} & \frac{\partial \varepsilon_p}{\partial w_0} \end{bmatrix} = \begin{bmatrix} x_{1_1} & \cdots & x_{n_1} & 1 \\ \vdots & & \vdots & \vdots \\ x_{1_p} & \cdots & x_{n_p} & 1 \end{bmatrix} \tag{6}$$

where: w is the vector of the weights, w_0 is the bias of the PE and ε is the error vector, i.e. the difference between the actual and the required value of the ANN output for the individual pattern. The parameter λ is modified based on the development of the error function E.

3 Application of ANN in Relative Humidity and Air Temperature Estimation

The present work aims to quantify the ability of ANNs to estimate and model the temporal and spatial AT and RH variabilities at a coastal environment. We focus on implementation issues and on evaluating the accuracy of the aforementioned methodologies in the case of a specific region with complex terrain. A number of alternative ANN topologies are applied in order to assess the spatial and time series AT and RH prediction capabilities in different time scales.

Moreover, this work presents an attempt to develop an extensive model performance evaluation procedure for the estimation of the RH and the AT using ANNs. This procedure incorporates a variety of correlation and difference statistical measures. In detail, the correlation coefficient (R), the coefficient of determination (R^2), the mean bias error (MBE), the mean absolute error (MAE), the root mean square error (RMSE) and the index of agreement (d) are calculated for the examined predictive schemes. The formulation and the applicability of such measures are extensively reported in [20, 21].

3.1 Area of Study

The study area is the Chania plain, located in the northwestern part of the island of Crete in Greece. The greater area is constricted by physical boundaries, which are the White Mountains on the south, the Aegean coastline on the northern and eastern part and the Akrotiri peninsula in the northeast of Chania city (Fig. 1). The topography of the region is complex due to the geophysical features of the region. The influence of the island of Crete on the wind field, especially during summer months and days where northerly etesian winds prevail, is proven to cause a leftward deflection and an upstream deceleration of the wind vector [22–24]. Moreover, the wind direction of the local field in the broader area of Chania city varies significantly due to the different topographical features [25].

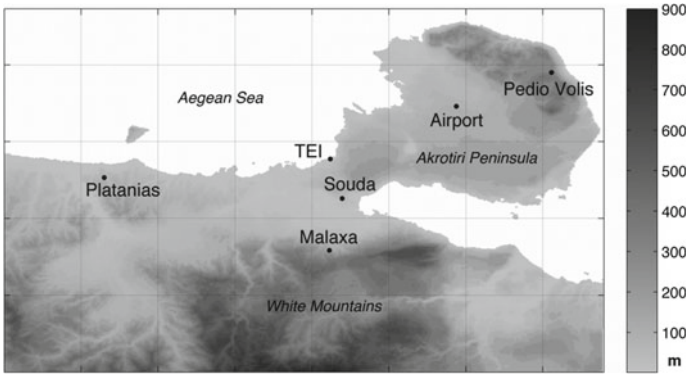


Fig. 1 Area of study and location of meteorological stations

Table 1 Geographical characteristics of the meteorological stations

Station Name	Latitude (°N)	Longitude (°W)	Elevation (m)	Characterization
Airport	24° 07′ 00″	35° 33′ 00″	140	Rural
TEI	35° 31′ 09″	24° 02′ 33″	38	Suburban—coastal
Souda	35° 30′ 30″	23° 54′ 40″	118	Suburban
Platanias	35° 29′ 46″	24° 03′ 00″	23	Rural—coastal
Malaxa	35° 27′ 57″	24° 02′ 33″	556	Rural
Pedio Volis	35° 34′ 11″	24° 10′ 20″	422	Rural

In this study, mean hourly AT and RH data are obtained from a network of six meteorological stations, namely Airport, Souda, Platanias, Malaxa, Pedio Volis and TEI (Fig. 1). The measurement sites cover the topography and land-use variability of the region (Table 1). The climatological station at the Airport is representative of the meteorological conditions that prevail at the Akrotiri peninsula and in this application it will be used as the reference station for examining the performance of the temporal and spatial pattern recognition approaches. TEI, Souda and Malaxa stations are situated along the perpendicular to the Aegean coastline north-south axis of the Chania basin, while the TEI and Platanias stations are representative of the coastal character of the basin. Moreover, TEI station is located at the east and in close proximity to the densely populated urban district of Chania city.

The topography induces significant spatial AT and RH variation. In detail, the inland stations at Souda and at the Airport exhibit the highest diurnal temperature ranges (7.75 and 6.56 °C respectively), while the spatial minimum is observed at Pedio Volis (2.32 °C), a finding that is attributed to the effect of altitude and the proximity of the site to the Aegean coastline. The highest daily maximum temperature values, averaged over the experimental period, are reported at the Airport (24 °C) and the lowest at Malaxa (19.46 °C).

3.2 Spatial Estimation of Air Temperature

Implementation. For the spatial estimation of air temperature the non-linear Feed Forward Artificial Neural Networks MLPANN and RBFANN are compared. The method aims to estimate the temperature at a target station, using AT observations as inputs from adjacent control stations.

The target station is located at Airport, while the concurrent AT observations from the remaining sites—control stations (Souda, Malaxa, Platanias, PedioVolis and TEI) are used as inputs in the MLPANN and RBFANN models.

The study period is from 19 July 2004 to 31 August 2006 and due to missing observations the input datasets consist of 12,416 simultaneous samples of hourly observations for each station. The 60 % of the available data (7,450 cases from 19 July 2004 at 23:00:00 to 1 Oct. 2005 at 09:00:00) was used for building and training the models (training set), the subsequent 20 % as the validation set (2,483 cases from 1 Oct. 2005 at 10:00:00 to 26 March 2006 at 11:00:00) and the remaining 20 % (2,483 cases from 26 March 2006 at 12:00:00 to 31 Aug. 2006 at 22:00:00) as the test set which is used to examine the performance of both the RBFANN and the MLPANN models. In MLPANNs the validation set is used for early stopping and to determine the optimum number of hidden layer neurons and in the RBFANNs to determine the optimum value of the spread parameter of the radial basis function. Large spread values result in a smooth function approximation that might not model the temperature variability adequately, while small spread values can lead to networks that might not generalize well. In our case the validation set is used for selecting the optimum value of the spread parameter, using the trial and calculating the error procedure by minimizing the MAE.

The optimum architecture for the MLPANN model is 5-17-1 (5 inputs, 17 hidden layers and 1 output neuron). The RBFANN used had five inputs and a radial basis hidden layer with 7,450 artificial neurons using Gaussian activation functions $\text{radbas}(n) = \exp(-n^2)$. The output layer had one PE with linear activation function.

Results. The model evaluation statistics for the Airport station for both MLPANN and RBFANN approaches are presented in Table 2. A general remark is that both models give accurate air temperature estimates with MAE values less than 0.9 °C and with very high d values and minimal biases. Furthermore the explained variance is 95.9 % for the RBFANN model and 96.3 % for the MLPANN scheme. The metrics indicate that MLPANN slightly outperforms the trained RBFANN network.

The comparison of the observed and the predicted air temperature values for both models are presented in Fig. 2 scatter plots and the respective residuals' distributions are given in Fig. 3. Limited data dispersion is observed for both models and in both cases the residuals are symmetrically distributed around 0 °C.

Moreover, a time series comparison between the observed and the predicted air temperature from the MLPANN and RBFANN models are presented in Fig. 4 for the period 10–23/8/2006. The predicted air temperature time series follows closely the observed values with no signs of systematic errors.

Table 2 ANN based model performance

	MLPANN	RBFANN
R	0.981	0.979
R^2	0.963	0.959
MBE (°C)	−0.008	0.034
MAE (°C)	0.819	0.871
RMSE (°C)	1.067	1.120
d	0.990	0.989

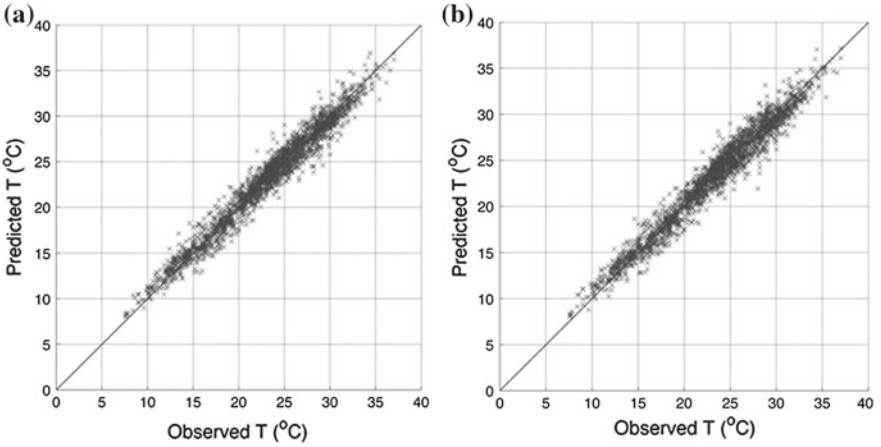


Fig. 2 Comparison of the observed and predicted air temperature values for the **a** MLPANN and **b** RBFANN schemes

The temperature estimation errors are further examined by calculating the MAE hourly values (Fig. 5). The analysis of both ANN models reveals two maxima, which are observed during the early morning warming period and during the late afternoon temperature decrease. The increase in the model errors can be attributed to the different heating and cooling rates between stations, a mechanism that is highly site specific and is greatly influenced by the local topography. For the remaining hours, both models are very accurate with errors less than 0.7 °C, a fact, which indicates the ability of the models to estimate with high accuracy the maximum, minimum and diurnal temperature range for the examined site.

3.3 Temporal Estimation of Air Temperature

Implementation. In the temporal forecasting of air temperature ANNs are used as function approximators aiming to estimate the AT in a location using the current and previous AT observations from the same site.

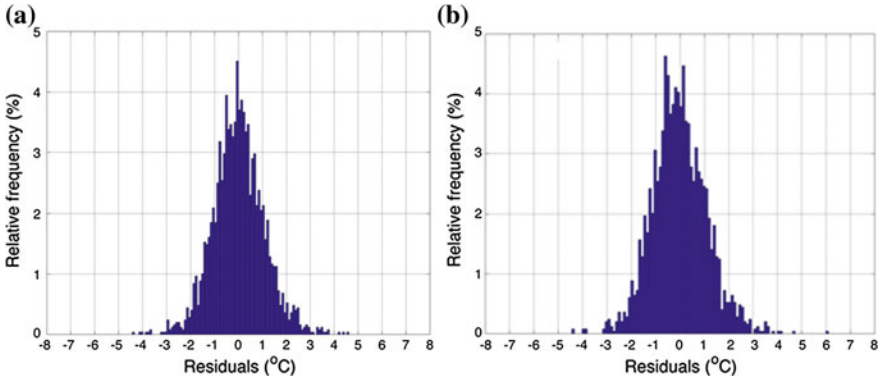


Fig. 3 The residuals' distribution for the **a** MLPANN and the **b** RBFANN models

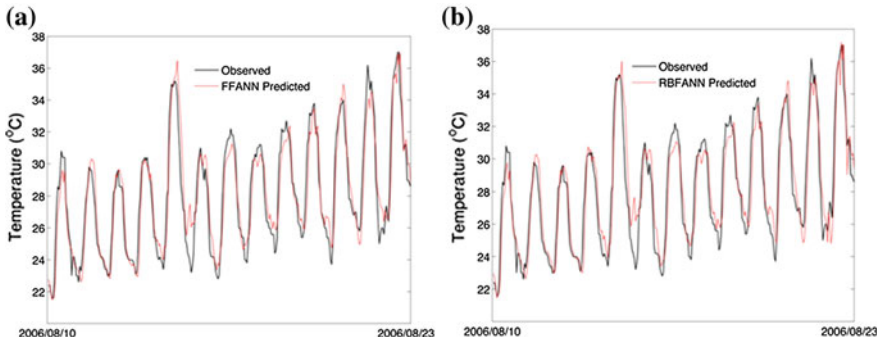


Fig. 4 Comparison of the observed and predicted air temperature time series for the **a** MLPANN and **b** RFBANN models

In this application the Feed-Forward Artificial Neural Network architecture with one hidden layer is selected for predicting the AT time series.

Separate ANNs are trained and tested in predicting the one hour (ANN-T1), two hours (ANN-T2) and three hours (ANN-T3) ahead air temperature at Airport station, based on the current and the five previous air temperature observations from the same site. Therefore, the input in each ANN is the air temperature at t, $t-1$, $t-2$, $t-3$, $t-4$ and $t-5$ and the output is the air temperature at: $t+1$ for the ANN-T1, $t+2$ for the ANN-T2 and $t+3$ for the ANN-T3.

The study period is from 19 July 2004 to 31 August 2006. In all cases, the first 60 % of the dataset is used for training the ANNs, the subsequent 20 % for validation and the remaining 20 % for testing, as was described for the case of spatial estimation of air temperature.

The optimum architecture (number of PEs in the hidden layer) is related to the complexity of the input and output mapping, along with the amount of noise and the size of the training data. A small number of PEs result to a non-optimum estimation

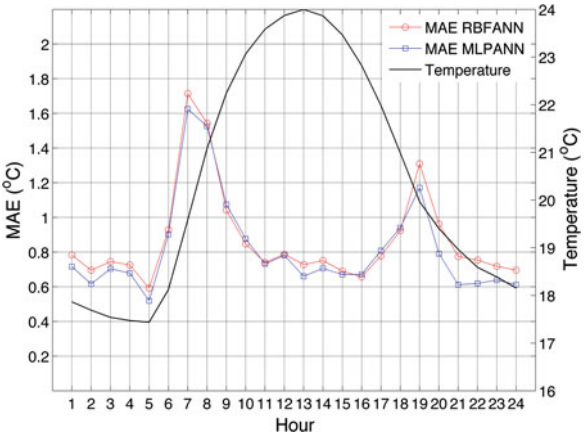


Fig. 5 Hourly MAE values and comparison with the hourly temperature evolution and the Airport station

Table 3 Optimum ANN architecture—number of PEs at the input, hidden and output layer

FFANN-T1	FFANN-T2	FFANN-T3
6 – 12 – 1	6 – 13 – 1	6 – 21 – 1

of the input-output relationship, while too many PEs result to overfitting and failure to generalize [26]. In this study the selection of the number of PEs in the hidden layer is based on a trial and error procedure and the performance is measured using the validation set. In each case, ANNs with a varying number from 5 to 25 PEs in the hidden layer were trained using the Levenberg-Marquardt backpropagation algorithm with the optimum architecture being the one that minimizes the Mean Absolute Error (MAE) on the validation set. A drawback of the backpropagation algorithm is its sensitivity to initial weights. During training, the algorithm can become trapped in local minima of the error function, preventing it from finding the optimum solution [27]. In this study and for eliminating this weakness, each network is trained multiple times (50 repetitions) with different initial weights. A hyperbolic tangent sigmoid transfer function $\text{tansig}(n) = 2/(1+\exp(-2n)) - 1$ was used as the activation function Ψ for the PEs of the hidden layer. In the output layers, PEs with a linear transfer function were used.

The optimum topologies of the selected ANNs that minimized the MAE on the validation set are presented in Table 3. In all cases, the architecture includes six PEs in the input layer and one PE in the output layer. The results indicate that the number of the neurons in the hidden layer is increased as the lag for forecasting the air temperature is increased.

Results. The model evaluation statistics for the Airport station are presented in Table 4 and the observed and ANN based predicted air temperature values are

Table 4 ANN based model performance

	FFANN-T1	FFANN-T2	FFANN-T3
R	0.988	0.967	0.942
R^2	0.977	0.935	0.887
MBE (°C)	−0.068	−0.225	−0.405
MAE (°C)	0.589	0.996	1.361
RMSE (°C)	0.844	1.427	1.904
d	0.994	0.983	0.968

compared in the scatter plots of Fig. 6. A general remark is that the ANNs performance is decreased with increasing the forecasting lag. In all cases the MAE is less than 1.4 °C and the explained variance decreases from 97.7 % for the ANN-T1 to 88.7 % for the ANN-T3 model.

The ANN-T1 model exhibits very good performance, as it is observed from the limited dispersion along the optimum agreement line of the one-hour air temperature (Fig. 6a). The data dispersion for the ANN-T2 (Fig. 6b) and for the ANN-T3 (Fig. 6c) scatter plots is increased and a small tendency of over-estimation of the low air temperature values along with an under-estimation of the high air temperature values is observed. This finding is furthermore established from the increased MBE for the ANN-T3 model (°C).

Regarding the residuals' distributions (Fig. 7), the errors for the ANN-T1 and for the ANN-T2 are approximately centered at 0 °C, while for the ANN-T3 model the maxima of the distribution is shifted to negative residual values, a fact which is attributed to the tendency of the ANN-T3 model to underestimate the air temperature values.

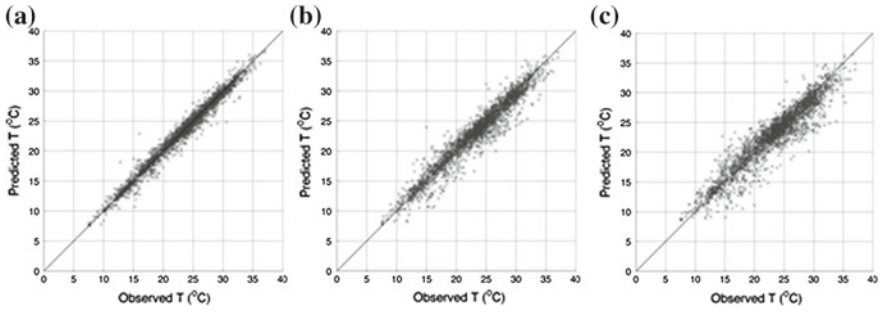


Fig. 6 Comparison of the observed and ANN based predicted air temperature values for the **a** one-hour, **b** two-hour and **c** three-hour ahead estimation

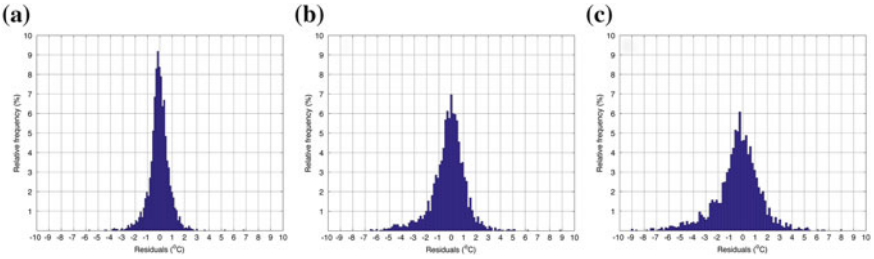


Fig. 7 Comparison of the residuals' distributions for the **a** FFANN-T1, **b** FFANN-T2 and **c** FFANN-T3 models

3.4 Spatial and Temporal Estimation of Relative Humidity

For the spatial and temporal estimation of relative humidity the FFANN models are employed. The implementation details are the same as in the case of air temperature presented above. In accordance with the spatial and temporal RH the FFANN models are used as function approximators of the relative humidity spatial and temporal variability in short temporal and spatial scales.

The optimum architecture for the spatial FFANN model is 5-26-1. For the temporal models, the number of the input-hidden-output neurons are presented in Table 5.

A general remark is that for both applications the number of the hidden layer neurons is higher for the RH estimation, compared to ambient temperature. This finding is attributed to the increased complexity of the RH input-output mapping, which is accomplished by the neural network transfer functions.

Regarding the RH spatial estimation, the model performance results are summarized in Table 6. The explained variance is greater than 72 % and the model according to the MBE statistic (−1.684 %), slightly underestimates the observed values. This minor tendency is also noted at the residuals distribution (Fig. 8a), where higher frequencies are associated with positive residual values. The comparisons of the observed versus the predicted RH values show some dispersion along the optimum agreement line (Fig. 8b). In the comparison of the observed and predicted RH time series (Fig. 9) some discrepancies mainly for the higher and lower values are observed.

Similarly with the temperature spatial estimation MAE hourly values (Fig. 5), the corresponding RH MAE hourly statistic values (Fig. 10) exhibit two maxima during

Table 5 FFANN optimum topology for the temporal estimation of RH

FFANN-T1	FFANN-T2	FFANN-T3
6-26-1	6-39-1	6-49-1

Table 6 Model evaluation statistics for the RH spatial FFANN model

R	R^2	MBE (%)	MAE (%)	RMSE (%)	d
0.852	0.726	−1.684	6.828	8.992	0.917

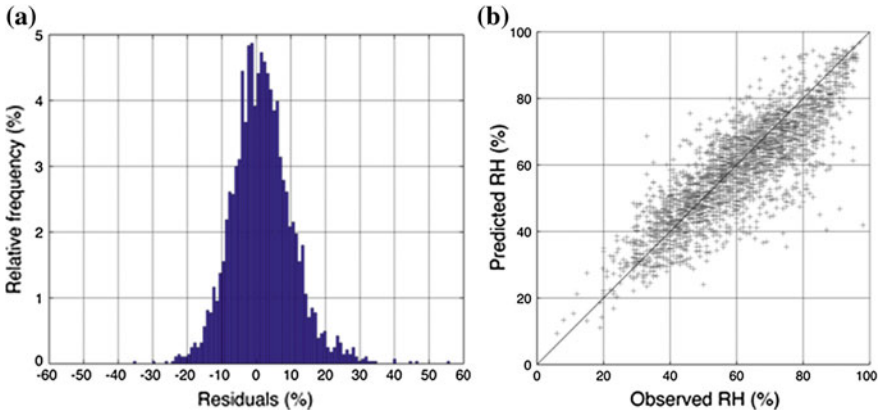


Fig. 8 **a** FFANN spatial model residuals distribution and **b** comparison of the observed and predicted RH values

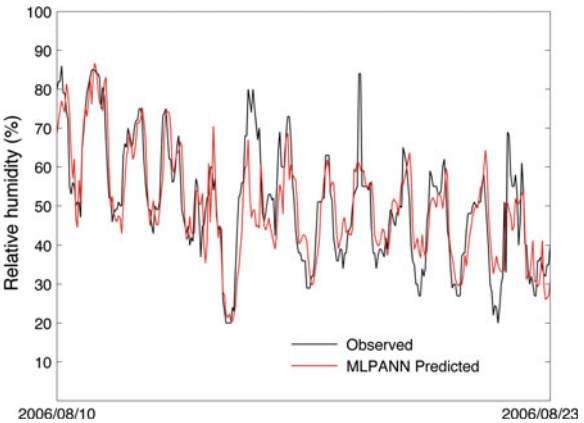


Fig. 9 Comparison of the observed and predicted RH time series

the morning and late in the afternoon. The lower MAE values (less than 6 %) are observed during midday where the daily minima of RH are observed.

Regarding the temporal estimation of RH, the overall performance statistical metrics are presented in Table 7. As expected in this case also the overall predictability of the FFANN models is decreased with increasing forecasting lag. In detail, in accordance with the scatter diagrams (Fig. 11) and the residual distributions (Fig. 12), the predictive ability of the FFANN models is high for the one-hour ahead estimation and decreases for the two and three-hour ahead RH estimations. The overall explained variance is decreased from 87 to 73.9 % and 60.4 % and the MAE values are increased from 4.3 to 6.5 % and 8.1 %. The index of agreement is higher than 0.9 for the one and two-hour ahead predictions and falls below this limit for the three hour ahead

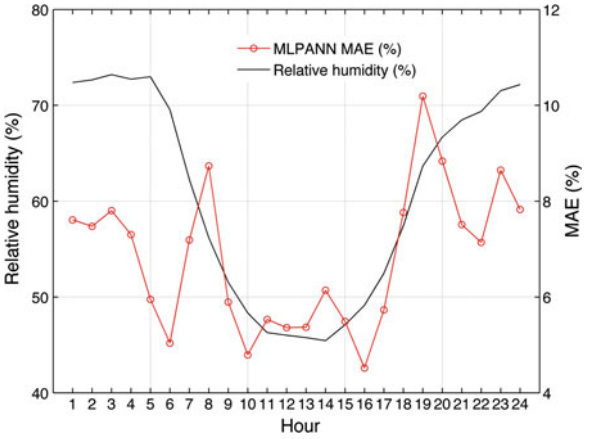


Fig. 10 Hourly MAE values in comparison with the diurnal RH evolution at the Airport site

Table 7 Model evaluation statistics for the temporal RH FFANN models

	FFANN-T1	FFANN-T2	FFANN-T3
R	0.933	0.860	0.777
R^2	0.870	0.739	0.604
MBE (%)	0.620	1.218	1.207
MAE (%)	4.280	6.497	8.092
RMSE (%)	6.135	8.685	10.659
d	0.965	0.921	0.869

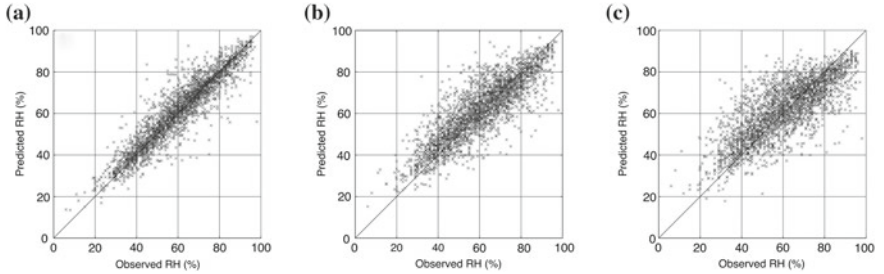


Fig. 11 Comparison of the observed and predicted RH values for the **a** FFANN-T1, **b** FFANN-T2 and **c** FFANN-T3 models

estimation. The FFANN-T2 and FFANN-T3 models show signs of overestimation (MBE statistic values greater than 1.2 %).

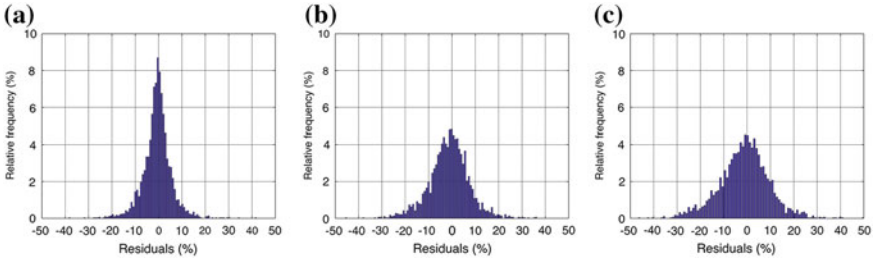


Fig. 12 Residual distributions for the **a** FFANN-T1, **b** FFANN-T2 and **c** FFANN-T3 models

4 Conclusions

The ability of neural networks to spatial estimate and predict short AT and RH values is studied extensively and is well established. We reviewed the theoretical background and the relative advantages and limitations of ANN methodologies applicable to the field of AT and RH time series and spatial modeling. Then, we have applied ANNs methodologies in the case of a specific region with complex terrain at Chania coastal region, Crete island, Greece. Details of the implementation issues are given along with the set of metrics for evaluating the accuracy of the methodology. A number of alternative feed-forward ANN topologies have been applied in order to assess the spatial and time series RH and AT prediction capabilities. For the one hour, two hours and three hours ahead AT and RH temporal forecasting at a specific site, ANNs were trained based on the current and the five previous AT and RH observations from the same site using the Levenberg-Marquardt back-propagation algorithm. The optimum architecture is the one that minimizes the Mean Absolute Error on the validation set. For the spatial estimation of AT at a target site the non-linear Radial Basis Function and Multilayer Perceptrons non-linear Feed Forward AANs schemes were compared. The underlying relative humidity and air temperature temporal and spatial variability is found to be modeled efficiently by the ANNs.

References

1. Price, D.T., McKenney, D.W., Nalder, I.A., Hutchinson, M.F., Kesteven, J.L.: A comparison of two statistical methods for spatial interpolation of Canadian monthly mean climate data. Agric. For. Meteorol. **101**(2–3), 81–94 (2000). doi:10.1016/S0168-1923(99)00169-0
2. Chai, H., Cheng, W., Zhou, C., Chen, X., Ma, X., Zhao, S.: Analysis and comparison of spatial interpolation methods for temperature data in Xinjiang Uygur autonomous region, China. Nat. Sci. **3**(12), 999–1010 (2011). doi:10.4236/ns.2011.312125
3. Deligiorgi, D., Philippopoulos, K.: Spatial interpolation methodologies in urban air pollution modeling: application for the greater area of metropolitan Athens, Greece. In: Nejadkoorki, F. (ed.) Advanced Air Pollution. InTech Publishers, Rijeka (2011). doi:10.5772/17734
4. Deligiorgi, D., Philippopoulos, K., Kouroupetroglou, G.: Artificial neural network based methodologies for the estimation of wind speed. In: Cavallaro, F.F. (ed.) Assessment and

Simulation Tools for Sustainable Energy Systems. Springer, Berlin (2013)
5. Snell, S., Gopal, S., Kaufmann, R.: Spatial Interpolation of surface air temperatures using artificial neural networks: evaluating their use for downscaling GCMs. J. Clim. **13**(5), 886–895 (2000). doi:10.1175/1520-0442(2000)013<0886:SIOSAT>2.0.CO;2
6. Chronopoulos, K., Tsiros, I., Dimopoulos, I., Alvertos, N.: An application of artificial neural network models to estimate air temperature data in areas with sparse network of meteorological stations. J. Environ. Sci. Health Part A: Tox./Hazard. Subst. Environ. Eng. **43**(14), 1752–1757 (2008). doi:10.1080/10934520802507621
7. Tasadduq, I., Rehman, S., Bubshait, K.: Application of neural networks for the prediction of hourly mean surface temperatures in Saudi Arabia. Renew. Energy **25**(4), 545–554 (2002). doi:10.1016/S0960-1481(01)00082-9
8. Dombayc, O., Golcu, M.: Daily means ambient temperature prediction using artificial neural network method: a case study of Turkey. Renew. Energy **34**(3), 1158–1161 (2009). doi:10.1016/j.renene.2008.07.007
9. Smith, B., Hoogenboom, G., McClendon, R.: Artificial neural networks for automated year-round temperature prediction. Comput. Electron. Agric. **68**(1), 52–61 (2009). doi:10.1016/j.compag.2009.04.003
10. Mustafaraj, G., Lowry, G., Chen, J.: Prediction of room temperature and relative humidity by autoregressive linear and nonlinear neural network models for an open office. Energy Build. **43**(6), 1452–1460 (2011). doi:10.1016/j.enbuild.2011.02.007
11. Mihalakakou, G., Flocas, H., Santamouris, M., Helmis, C.: Application of neural networks to the simulation of the heat island over Athens, Greece, using synoptic types as a predictor. J. Appl. Meteorol. **41**(5), 519–527 (2002). doi:10.1175/1520-0450(2002)041<519:AONNTT>2.0.CO;2
12. Fausett, L.V.: Fundamentals Neural Networks: Architecture, Algorithms, and Applications. Prentice-Hall Inc., New Jersey (1994)
13. Bishop, C.M.: Neural Networks For Pattern Recognition. Oxford University Press, Cambridge (1995)
14. Jain, A.K., Mao, J., Mohiuddin, K.M.: Artificial neural networks: a tutorial. Computer **29**(3), 31–44 (1996). doi:10.1109/2.485891
15. Zhang, G.P., Patuwo, E., Hu, M.: Forecasting with artificial neural networks: the state of the art. Int. J. Forecast. **14**(1), 35–62 (1998). doi:10.1016/S0169-2070(97)00044-7
16. Cybenco, G.: Approximation by superposition of a sigmoidal function. Math. Control Signals Syst. **2**(4), 303–314 (1989). doi:10.1007/BF02551274
17. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Netw. **2**(5), 359–366 (1989). doi:10.1016/0893-6080(89)90020-8
18. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**, 533–536 (1986). doi:10.1038/323533a0
19. Yu, H., Wilamowski, B.M.: Levenberg-Marquardt training. In: Wilamowski, B.M., Irwin, J.D. (eds.) Industrial Electronics Handbook, 2nd edn. CRC Press, Boca Raton (2011)
20. Fox, D.G.: Judging air quality model performance. Bull. Am. Meteorol. Soc. **62**(5), 599–609 (1981). doi:10.1175/1520-0477(1981)062<0599:JAQMP>2.0.CO;2
21. Willmott, C.J.: Some comments on the evaluation of model performance. Bull. Am. Meteorol. Soc. **63**(11), 1309–1313 (1982). doi:10.1175/1520-0477(1982)063<1309:SCOTEO>2.0.CO;2
22. Koletsis, I., Lagouvardos, K., Kotroni, V., Bartzokas, A.: The interaction of northern wind flow with the complex topography of Crete island-Part 1: observational study. Nat. Hazards Earth Syst. Sci. **9**, 1845–1855 (2009). doi:10.5194/nhess-9-1845-2009
23. Koletsis, I., Lagouvardos, K., Kotroni, V., Bartzokas, A.: The interaction of northern wind flow with the complex topography of Crete island-Part 2: numerical study. Nat. Hazards Earth Syst. Sci. **10**, 1115–1127 (2010). doi:10.5194/nhess-10-1115-2010
24. Kotroni, V., Lagouvardos, K., Lalas, D.: The effect of the island of Crete on the etesian winds over the Aegean sea. Q. J. R. Meteorol. Soc. **127**(576), 1917–1937 (2001). doi:10.1002/qj.49712757604

25. Deligiorgi, D., Kolokotsa, D., Papakostas, T., Mantou, E.: Analysis of the wind field at the broader area of Chania, Crete. In: 3rd IASME/WSEAS International Conference on Energy, Environment and Sustainable Development, pp. 270–275. Agios Nikolaos, Crete: World Scientific and Engineering Academy and Society Press (2007). Retrieved from:http://www.wseas.us/e-rary/conferences/2007creteeeesd/papers/562-194.pdf
26. Gardner, M.W., Dorling, S.R.: Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. Atmos. Environ. **32**(14–15), 2627–2636 (1998). doi:10.1016/S1352-2310(97)00447-0
27. Heaton, J.: Introduction to Neural Networks with Java. Heaton Research Inc, Chesterfield (2005)
28. Powel, M.J.D.: Radial basis functions for multivariable interpolation: a review. In: Mason, J.C., Cox, M.G. (eds.) Algorithms for Approximation. Clarendon Press, Oxford (1987)

Part II
Applications

Beyond SIFT for Image Categorization by Bag-of-Scenes Analysis

Sébastien Paris, Xanadu Halkias and Hervé Glotin

Abstract In this paper, we address the general problem of image/object categorization with a novel approach referred to as *Bag-of-Scenes* (BoS). Our approach is efficient for both low semantic applications, such as texture classification and higher semantic tasks such as natural scenes recognition. It is based on the widely used combination of (i) Sparse coding (Sc), (ii) Max-pooling and (iii) Spatial Pyramid Matching (SPM) techniques applied to histograms of multi-scale Local Binary/Ternary Patterns (LBP/LTP) as local features. This approach can be considered as a two-layer hierarchical architecture. The first layer encodes quickly the local spatial patch structure *via* histograms of LBP/LTP, while the second layer encodes the relationships between pre-analyzed LBP/LTP-scenes/objects. In order to provide comparative results, we also introduce an alternate 2-layer architecture. For this latter, the first layer is encoding directly the multi-scale Differential Vectors (DV) local patches instead of histograms of LBP/LTP. Our method outperforms SIFT-based approaches using Sc techniques and can be trained efficiently with a simple linear SVM. Our BoS method achieves 87.46 %, and 90.35 % of accuracy for Scene-15, UIUC-Sport datasets respectively.

Keywords Image categorization · Scenes categorization · Fine-grained visual categorization · Non-parametric local patterns · Multi-scale LBP/LTP · Dictionary learning · Sparse coding · LASSO · Max-pooling · SPM · Linear SVM

Granded by COGNILEGO ANR 2010-CORD-013 and PEPS RUPTURE Scale Swarm Vision.

S. Paris (✉)
DYNI Team, LSIS CNRS UMR 7296, Aix-Marseille University, Marseille, France
e-mail: sebastien.paris@lsis.org

X. Halkias · H. Glotin
DYNI Team, LSIS CNRS UMR 7296, Université Sud Toulon-Var, La Garde, France

H. Glotin
Institut Universitaire de France, Paris, France

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_12

1 Introduction

Image categorization consists of assigning a unique label with a generally high-level semantic value to an image. It has long been a challenging problem area in computer vision, biomonitoring and robotics and can be mainly viewed as belonging to the broader supervised classification framework. In scene categorization, the difficulty of the task can be partly explained by the high-dimensional input space of the images as well as the high-level semantic visual concepts that lead to large intra-class variation. Specifically, for object recognition the small aspect ratio (object's size vs image's size) can induce a high level of uninformative background pixels. A preliminary detection procedure is required to "hone-in" the object in a Region of Interest (ROI) [1, 2].

The *direct* framework in vision systems consists of extracting directly from the images meaningful features (using shape/texture/similarity/color information) in order to achieve the maximum generalization capacity during the classification stage. Examples of such popular features in computer vision and human cognition inspired models include GIST [3], HOG [4], Self-Similarity [5] and WLD [6].

Widely used in face detection [7, 8], face recognition [9, 10], texture classification [11, 12] and scene categorization [13–16], Local Binary Pattern (LBP) [17] and recent derivatives such as Local Ternary Pattern (LTP) [18], Gabor-LBP [19, 20], Local Gradient Pattern (LGP) [21] or Local Quantized Pattern (LQP) [22] are efficient local micro-patterns that define competitive features achieving state-of-the-art performances. LBP can be considered as a non-parametric local visual micro-pattern texture, encoding mainly contours and differential excitation information of the 8 neighbors surrounding a central pixel [23, 24]. This process represents a contractive mapping from $\mathbb{R}^9 \mapsto \mathbb{N}_{2^8} \subset \mathbb{N}^+$ for each local patch $p(\boldsymbol{x})$ centered in $\boldsymbol{x}$ ([25] provide a theoretical study of LBP).

The total number of different LBPs is relatively small and by construction is finite: from 256 up to 512 different patterns (if improved LBP is used). LTPs [26] have been extended from LBP as a parametric approximation of a ternary pattern. Instead of mapping $\mathbb{R}^9 \mapsto \mathbb{N}_{3^8} \subset \mathbb{N}^+$, they propose a split of the ternary pattern into two binary patterns followed by a concatenation of the two associated histograms. In [22], they generalize local patterns with the use of LQP by increasing neighborhood range, number of neighbors and pattern cardinality leading to map $\mathbb{R}^9 \mapsto \mathbb{N}_{k^N} \subset \mathbb{N}^+$. Histograms of LBPs (HLBP) (respectively HLTPs), which count the occurrence of each LBP (respectively LTP) in the scene, can easily capture general structures in the visual scene by integrating information in a ROI, while being less sensitive to local high frequency details. This property is import ant when the desire is to generalize visual concepts. As depicted in this work, it is advantageous to extend this analysis for several sizes of local ROIs using a spatial pyramid denoted by $\boldsymbol{\Lambda}$.

Recently, the alternative scheme of *Bag-of-Features* (BoF) has been employed in several computer vision tasks with wide success. It offers a deeper extraction of visual concepts and improves accuracy of computer vision systems. BoF image representation [27] and its SPM extension [28] share the same idea as HLBP: counting the

presence (or combination) of visual patterns in the scene. BoF contains at least three modules prior to the classification stage: (i) region selection for local feature extraction; (ii) codebook/dictionary generation and feature quantization; (iii) frequency histogram based image representation with SPM.

In general, SIFT/HOG patches [4, 29] are employed in the first module. These visual descriptors are then encoded, in an unsupervised manner, into a moderate sized dictionary using Vector Quantization (VQ) [28] or sparse coding [30]. Both SIFT patches or LBP computation followed by histograms can be seen as an encoder and a pooler process respectively [31]. In other words, computing Histograms of LBP/LTP can replace advantageously a first layer of encoder-pooler working on differential vectors. In [32], Wu et al. were first to introduce LBP (*via* CENTRIST) into a BoF framework coupled with the histogram intersection kernel (HIK). At least two disadvantages can be addressed against the BoF framework, mainly concerning the second stage. Firstly, the trained dictionaries don't have enough representative basis vectors for some (rare and detailed) local patches that are crucial for discriminative purposes. Secondly, during quantification/encoding a lot of important information can be lost [33].

In order to improve the encoding scheme, it has been shown that localized soft-assignment [34], local-constrained linear coding (LLC) [35], Fisher vectors (FV) [36, 37], orthogonal matching pursuit (OMP) [38] or Sparse coding (Sc) [14, 30] can easily be plugged into the BoF framework as a replacement for VQ. Moreover, pooling techniques coupled with SPM [28] can be effectively used as a replacement for the global histogram based image representation.

Our contributions in this paper are three-fold. We first re-introduce two multi-scale variants of the LBP operators and extend two novel multi-scale variants of the LTP operators [26]. Secondly, we propose to insert HLBP/HLTP into the Sc framework as a second analyzing layer and call this procedure *Fast Bag-of-Scenes* (FBoS). This new approach is efficient both for scene categorization and object recognition. The novel features can be trained efficiently with simple large-scale linear SVM solver such as *Pegasos* [39] or *LIBLINEAR* [40]. FBoS can be seen as a two layer Hierarchical BoF analysis: a first fast contractive low-dimension manifold encoder *via* HLBP/HLTP and a second inflating high-dimension encoder *via* Sc. Finally, in order to compare obtained results with FBoS, we introduce the multi-scale differential vectors as local features and use 2 cascading layers to encode these local features. We call this procedure *Exact Bag-of-Scenes* (EBoS). These local features represent the common starting point of both FBoS and EBoS procedures, one use a parametric encoder while for the second, DV encoding is trained from data.

2 Fast Bag-of-Scenes with Histogram of Multi-scale Local Patterns as the First Layer

For an image/patch $\boldsymbol{I}$ $(n_y \times n_x)$, we present two existing multi-scale versions of the LBP operator, denoted by the B operator and for its *improved* variant by the IB operator. We also introduce two novel multi-scale versions of the LTP, denoted by the T operator and for its *improved* variant by the IT operator.

2.1 Multi-scale LBP/ILBP

Basically, operator B encodes the relationship between a central block of $(s \times s)$ pixels located in (y_c, x_c) with its 8 neighboring blocks [41], whereas operator IB adds a ninth bit encoding a term homogeneous to the differential excitation (see left Fig. 1). Both can be considered as a non-parametric local texture encoder for scales. In order to capture information at different scales, the range analysis $s \in \mathcal{S}$, is typically set at $\mathcal{S} = [1, 2, 3, 4]$ for this paper, where $S = Card(\mathcal{S})$.

These two micro-codes are defined as follows[1]:

$$\begin{cases} B(y_c, x_c, s) & = \sum_{i=0}^{i=7} 2^i \mathbb{1}_{\{A_i \geq A_c\}} \\ IB(y_c, x_c, s) & = B(y_c, x_c, s) + 2^8 \mathbb{1}_{\left\{\sum_{i=0}^{7} A_i \geq 8Ac\right\}}. \end{cases} \tag{1}$$

For $\forall (y_c, x_c) \in \boldsymbol{R} \subset \boldsymbol{I}$, $B(y_c, x_c, s) \in \mathbb{N}_{2^8}$ and $IB(y_c, x_c, s) \in \mathbb{N}_{2^9}$ respectively.

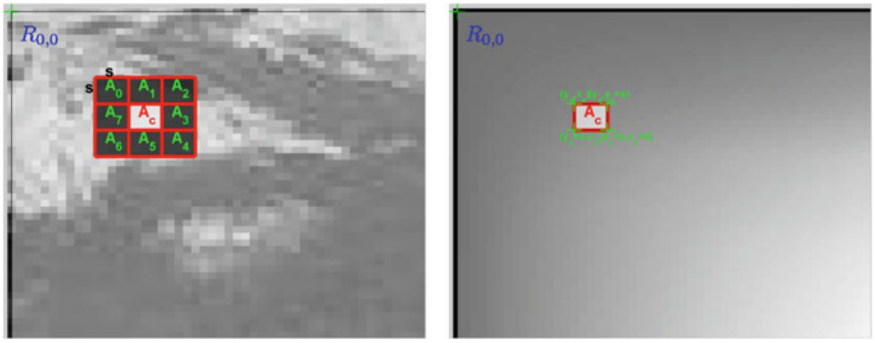


Fig. 1 *Left* $\boldsymbol{I}$ and $B(y_c, x_c, 4)$ overlaid. *Right* corresponding image integral $\boldsymbol{II}$ and the central block A_c. A_c can be efficiently computed with the 4 corner points

[1] $\mathbb{1}_{\{x\}} = 1$ if event x is true, 0 otherwise.

2.2 Multi-scale LTP/ILTP

We introduce the multi-scale version of LTP and its improved variant. The idea behind LTP is to extend the LBP for $k = 3$ with the help of a single threshold parameter $t \in \mathbb{N}_{2^8}$. With the same neighborhood configuration with $N = 8$ (see left Fig. 1), a direct extension would result to $3^8 = 6{,}561$ different patterns. In [26], they proposed to break the high dimensionality of the code by splitting the ternary code into two binary operators T_p and T_n such as:

$$\begin{cases} T_p(y_c, x_c, s; t) = \sum_{i=0}^{i=7} 2^i \mathbb{1}_{\{\frac{1}{s^2}(A_i - A_c) \geq t\}} \\ \\ T_n(y_c, x_c, s; t) = \sum_{i=0}^{i=7} 2^i \mathbb{1}_{\{\frac{1}{s^2}(A_i - A_c) \leq -t\}}. \end{cases} \tag{2}$$

The improved multi-scale LTP operators (denoted ILT_p and ILT_n) are derived similarly from MSLBP by:

$$\begin{cases} ILT_p(y_c, x_c, s; t) = T_p(y_c, x_c, s; t) + 2^8 \mathbb{1}_{\left\{\frac{1}{s^2}\left(\sum_{i=0}^{7} A_i - 8Ac\right) \geq t\right\}} \\ \\ ILT_n(y_c, x_c, s; t) = T_n(y_c, x_c, s; t) + 2^8 \mathbb{1}_{\left\{\frac{1}{s^2}\left(\sum_{i=0}^{7} A_i - 8Ac\right) \leq -t\right\}}. \end{cases} \tag{3}$$

Now, for $\forall (y_c, x_c) \in \boldsymbol{R} \subset \boldsymbol{I}$, both codes $\{T_p(y_c, x_c, s; t), T_n(y_c, x_c, s; t)\} \in \mathbb{N}_{2^8}$ while the improved version $\{ILT_p(y_c, x_c, s; t), ILT_n(y_c, x_c, s; t)\} \in \mathbb{N}_{2^9}$ respectively.

2.3 Integral Image for Fast Areas Computation

The different areas $\{A_i\}$ and A_c in Eqs. (1)–(3) can be computed efficiently using the image integral technique [42].

Let's define $\boldsymbol{II}$ the image integral of $\boldsymbol{I}$ by:

$$\boldsymbol{II}(y, x) \triangleq \sum_{y'=0}^{y'<y} \sum_{x'=0}^{x'<x} \boldsymbol{I}(y', x'). \tag{4}$$

Any square area $A(y, x, s) \in \boldsymbol{R}$ (see right Fig. 1) with upper-left corner located in (y, x) and side length s is the addition of only 4 values:

$$A(y,x,s) = \boldsymbol{II}(y+s,x+s) + \boldsymbol{II}(y,x) \\ -(\boldsymbol{II}(y,x+s) + \boldsymbol{II}(y+s,x)). \tag{5}$$

2.4 Histogram of Local Patterns

For all previously defined operators $op \in \{B, IB, T_p, T_n, IT_n, IT_p\}$, efficient features are obtained by counting occurrences of the jth visual LBP/LTP at scale s in a ROI $\boldsymbol{R} \subseteq \boldsymbol{I}$:

$$z_{op}(\boldsymbol{R}, j, s) = \sum_{(x_c,y_c)\in\boldsymbol{R}} \mathbb{1}_{\{op(y_c,x_c,s)=j\}},$$

where $j = 0, \ldots, b-1$ is the jth bin of the histogram and $b = \{256, 512, 256, 256, 512, 512\}$ for $op \in \{B, IB, T_p, T_n, IT_n, IT_p\}$ respectively.

Full histograms of LBP and its variant ILBP, denoted by $\boldsymbol{z}_B, \boldsymbol{z}_{IB}$, are computed by:

$$\boldsymbol{z}_{op}(\boldsymbol{R}, s) \triangleq \left[z_{op}(\boldsymbol{R}, 0, s), \ldots, z_{op}(\boldsymbol{R}, b-1, s)\right], \tag{6}$$

with a total size of patches $d = b = \{256, 512\}$ respectively.

For LTP, full histograms, denoted by $\boldsymbol{z}_T, \boldsymbol{z}_{IT}$ are defined by:

$$\boldsymbol{z}_{op}(\boldsymbol{R}, s) \triangleq \left[z_{op_p}(\boldsymbol{R}, 0, s), \ldots, z_{op_p}(\boldsymbol{R}, b-1, s), \ldots, \right. \\ \left., \ldots, z_{op_n}(\boldsymbol{R}, 0, s), \ldots, z_{op_n}(\boldsymbol{R}, b-1, s)\right], \tag{7}$$

with a total size of patches $d = 2 \cdot b = \{512, 1024\}$ respectively. To finalize the patch extraction stage, regardless of the type of histogram of local patterns used, a ℓ_2 clamped normalization procedure is performed on each histogram (clamp value = 0.2).

2.5 Sparse Coding on Patches of Multi-scale Local Patterns

Following the same framework as in [28, 30, 43, 44], we show here that the traditional BoF approach can be advantageously replaced by (i) Sc, (ii) max-pooling technique and (iii) a simple linear SVM as a classifier since the produced features are mostly linearly separable.

2.5.1 Patches of HB/HIB/HT/HIT

Here, we replace the collection of usual SIFT patches densely sampled on a grid by our HB/HIB/HT/HIT patches $\boldsymbol{z}$ seen previously. Specifically, F patches of size $(m \times m)$ associated with ROI's $\{\boldsymbol{O}_k\}$ (possibly overlapping) are extracted for $k = 0, \ldots, F-1$

and $\forall s \in \mathcal{S}$. For a faster computation for each scale s, the integral image $\boldsymbol{II}$ is first computed from $\boldsymbol{I}$.

For a complete dataset containing N images and $\forall s \in \mathcal{S}$, we obtain a collection of $P = T \cdot S$ patches $\boldsymbol{Z} \triangleq \{\boldsymbol{z}_i\}$, $i = 1, \ldots, P$, where $T = N \cdot F$. We define, the subset of patches $\boldsymbol{z}_i$ at scale s by $\boldsymbol{Z}(s) \subseteq \boldsymbol{Z}$ with T elements.

2.5.2 Sparse Coding Overview

In order to obtain highly discriminative visual features, a common procedure consists of encoding each patch $\boldsymbol{z}_i \in \boldsymbol{Z}(s)$ at scale s through an unsupervised trained dictionary $\boldsymbol{D} \triangleq [\boldsymbol{d}_1, \ldots, \boldsymbol{d}_K] \in \mathbb{R}^{b \times K}$, where K denotes the number of dictionary elements, and its corresponding weight vector $\boldsymbol{c}_i \in \mathbb{R}^K$. In the BoF framework, the vector $\boldsymbol{c}_i$ is assumed to have only one non-zero element:

$$\arg\min_{\boldsymbol{D},\boldsymbol{C}} \sum_{i=1}^{T} \|\boldsymbol{z}_i - \boldsymbol{D}\boldsymbol{c}_i\|_2^2 \quad s.t. \quad \|\boldsymbol{c}_i\|_{\ell_0} = 1, \tag{8}$$

where $\boldsymbol{C} \triangleq [\boldsymbol{c}_1, \ldots, \boldsymbol{c}_K]$ and $\| \bullet \|_{\ell_0}$ defines the pseudo zero-norm, where here only one element of $\boldsymbol{c}_i$ is non-zero. In Eq. (8), under these constraints, $(\boldsymbol{D}, \boldsymbol{C})$ can be optimized jointly by a K-means algorithm for example.

In the Sc approach, in order to (i) reduce the quantization error and (ii) to have a more accurate representation of the patches, each vector $\boldsymbol{x}_i$ is now expressed as a linear combination of a few vectors of the dictionary $\boldsymbol{D}$ and not only by a single one. Imposing the exact number of non-zero elements in $\boldsymbol{c}_i$ (sparsity level) involves a non-convex optimization [45]. In general, it is preferred to relax this constraint and to use instead an ℓ_1 penalty which also involves sparsity. The problem is then reformulated using the following equation:

$$\arg\min_{\boldsymbol{D},\boldsymbol{C}} \sum_{i=1}^{T} \|\boldsymbol{z}_i - \boldsymbol{D}\boldsymbol{c}_i\|_2^2 + \beta\|\boldsymbol{c}_i\|_{\ell_1} \quad s.t. \quad \|\boldsymbol{c}_i\|_{\ell_1} = 1, \tag{9}$$

where the sparsity is controlled by the parameter β. The last equation is not jointly convex in $(\boldsymbol{D}, \boldsymbol{C})$ and a common procedure consists of optimizing alternatively $\boldsymbol{D}$ given $\boldsymbol{C}$ by a block coordinate descent and then $\boldsymbol{C}$ given $\boldsymbol{D}$ by a LASSO procedure [46]. At the end of the process, for each scale $s \in \mathcal{S}$, a trained dictionary $\widehat{\boldsymbol{D}}(s)$ is obtained.

2.5.3 Spatial Pyramidal Matching and Max Pooling

For an image $\boldsymbol{I}$ and given a trained dictionary $\widehat{\boldsymbol{D}}(s)$ for a type of code at scale s, F sparse vectors $\{\boldsymbol{c}_k(s)\}$ are computed by a LASSO algorithm. The final efficient

descriptor $\boldsymbol{x}(s) \triangleq \left[x^0(s), \ldots, x^{K-1}(s)\right] \in \mathbb{R}^K$ is obtained by the following max-pooling procedure [30, 47]:

$$x^j(s) \triangleq \max_{k|\boldsymbol{O}_k \in \boldsymbol{R}} (|c_k^j(s)|), \;\; j = 0, \ldots, K-1, \tag{10}$$

where each element of $\boldsymbol{x}(s)$ represents the max-response of the absolute value of sparse codes belonging to the ROI $\boldsymbol{R}$. In order to improve accuracy, a spatial pyramidal matching procedure helps to perform a more robust local analysis. The spatial pyramid $\boldsymbol{\Lambda}$ has $V = \sum_{l=0}^{L-1} V_l$ ROIs $\{\boldsymbol{R}_{l,v}\}$ with $l = 0, \ldots, L-1$, $v = 0, \ldots, V_l - 1$ (see Fig. 2 for an example). The quantity $\boldsymbol{z}_{l,v}^j(s)$ for each ROI $\boldsymbol{R}_{l,v}$ is computed by:

$$x_{l,v}^j(s) \triangleq \max_{k|\boldsymbol{O}_k \in \boldsymbol{R}_{l,v}} (|c_k^j(s)|), \;\; j = 0, \ldots, K-1. \tag{11}$$

We define generally our SP matrix $\boldsymbol{\Lambda}$ with L levels such as $\boldsymbol{\Lambda} \triangleq \left[\boldsymbol{e}_y, \boldsymbol{e}_x, \boldsymbol{d}_y, \boldsymbol{d}_x, \boldsymbol{\lambda}\right]$, a matrix of size $(L \times 5)$. For a level $l \in \{0, \ldots, L-1\}$, the image $\boldsymbol{I}$, with size $(n_y \times n_x)$, is divided into potentially overlapping sub-windows $\boldsymbol{W}_{l,v}$ of size $(h_l \times w_l)$. All these windows are sharing the same associated weight λ_l. In our implementation, $h_l \triangleq \lfloor n_y.e_{y,l} \rfloor$ and $w_l \triangleq \lfloor n_x.e_{x,l} \rfloor$ where $e_{y,l}$, $e_{x,l}$ and λ_l are the l^{th} element of vectors $\boldsymbol{e}_y$, $\boldsymbol{e}_x$ and $\boldsymbol{\lambda}$ respectively. Sub-window shifts in $x-y$ axis are defined by integers $\delta_{y,l} \triangleq \lfloor n_y.d_{y,l} \rfloor$ and $\delta_{x,l} \triangleq \lfloor n_x.d_{x,l} \rfloor$ where $d_{y,l}$ and $d_{x,l}$ are elements of $\boldsymbol{d}_y$ and $\boldsymbol{d}_x$ respectively. Overlapping can be performed if $d_{y,l} \leq e_{y,l}$ and/or $d_{x,l} \leq e_{x,l}$. For example $\boldsymbol{\Lambda} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 1 \end{bmatrix}$ represents the two levels SP $(1 \times 1 + 2 \times 2)$. The total number of sub-windows is equal to $V = \sum_{l=0}^{L-1} V_l = \sum_{l=0}^{L-1} \lfloor \frac{(1-e_{y,l})}{d_{y,l}} + 1 \rfloor \cdot \lfloor \frac{(1-e_{x,l})}{d_{x,l}} + 1 \rfloor$.

We reinforce our model by an important normalization step, improving considerably accuracy, consists of the ℓ_2 normalization of all vectors $\{\boldsymbol{x}_{l,v}(s)\}$, $v =$

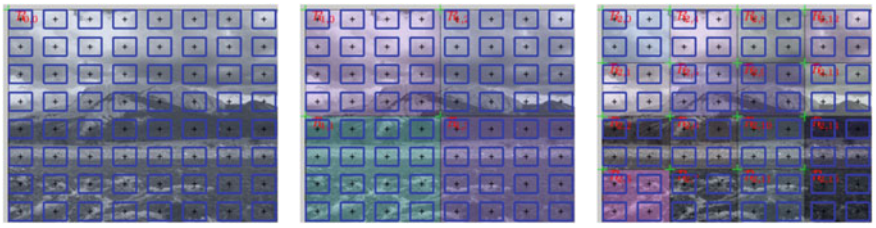


Fig. 2 Example of SPM $\boldsymbol{\Lambda}$ with $L = 3$, $F = 8 \times 8$ and $V = 1 + 4 + 16$. The F ROIs $\{\boldsymbol{O}_k\}$, $k = 0, \ldots, F-1$ associated with each patch $\boldsymbol{z}_k$ are represented by *blue squares*. Sparse codes $\boldsymbol{c}_k$ are computed for each ROI $\boldsymbol{O}_k$. *Upper-left corner* of each max-pooling window $\boldsymbol{R}_{l,v}$ taking {64, 16, 4} $\boldsymbol{c}_k$ is indicated with a *green cross*. *Left* $\boldsymbol{R}_{0,0} = \boldsymbol{I}$ for $l = 0$. *Middle* $\{\boldsymbol{R}_{1,v}\}$, $v = 0, \ldots, 3$ for $l = 1$. *Right* $\{\boldsymbol{R}_{2,v}\}$, $v = 0, \ldots, 15$ for $l = 2$

$0, \ldots, V_l - 1$, $s \in \mathcal{S}$, *i.e.* belonging to the same pyramidal layer l. This step is also very important and often hidden in the existing literature.

The final descriptor $\boldsymbol{x}(\boldsymbol{\Lambda})$ will be defined by the weighted concatenation of all the $\boldsymbol{x}_{l,v}(s)$ vectors, *i.e.* $\boldsymbol{x}(\boldsymbol{\Lambda}) \triangleq \{\lambda_l \boldsymbol{x}_{l,v}(s)\}$, $l = 0, \ldots, L-1$, $v = 0, \ldots, V_l - 1$ and $\forall s \in \mathcal{S}$. The total size of the feature vector $\boldsymbol{x}(\boldsymbol{\Lambda})$ is $f = K \cdot V \cdot S$, where typically in our simulations, we fixed $K = \{1024, 2048\}$, $V = \{10, 21, 26\}$ and $S = 4$. A final ℓ_2 clamped normalization step is performed on the full vector $\boldsymbol{x}(\boldsymbol{\Lambda})$.

3 Exact Bag-of-Scenes with Differential Vectors as Local Features

In order to compare our proposed method using HLBP/HLTP as fast first encoder-pooler layer, we also introduce a two layer architecture where the first layer will encode specifically our new DV patches (see Fig. 3). The idea is to learn directly from data how to encode differential values used in the LBP/LTP formulation.

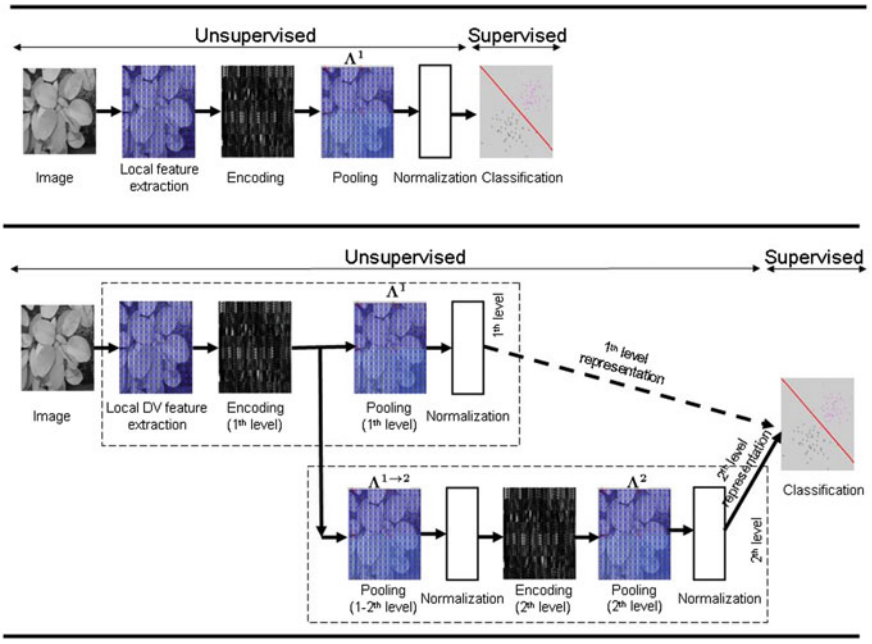


Fig. 3 *Top* classic flat *features extraction-coding-pooling* framework. *Bottom* our proposed 2-layer architecture

3.1 Multi-range Differential Vectors

For the first layer, in order to obtain some invariance to monotonic changes, we introduce (see Fig. 4) a multi-range differential vector $\boldsymbol{z}^1(s;\boldsymbol{R}) \in \mathbb{R}^d$ at scale s and centered in (y_c, x_c). $\boldsymbol{z}^1(s;\boldsymbol{R})$ is computed by subtracting areas A_i^s (of the square block pixels B_i^s) with the central area A_c^s (associated with block B_c^s). B_i^s blocks, belonging to some concentric holed squares surrounding B_c^s, are parameterized by the block's length vector $\boldsymbol{S} = [s_1, \ldots, s_{\mathcal{S}}]$ while the square lattice is parameterized by the border's width vector $\boldsymbol{U} = [u_1, \ldots, u_{\mathcal{U}}]$. DV at scale $s \in \boldsymbol{S}$ are defined by concatenation of all local differential values for all square's range:

$$\boldsymbol{z}^1(s;\boldsymbol{U}) \triangleq \bigcup_{r_i \in \boldsymbol{U}} \left\{ \bigcup_{j \setminus B_j^s \subset H(u_i,s)} \left\{ A_j^s - A_c^s \right\} \right\}, \tag{12}$$

where $H(u, s)$ is a holed square centered in (y_c, x_c) of border width s and eccentricity $u \cdot s$. The total DV length is equal to $d = \sum_{u_i \in \boldsymbol{U}} (4 \cdot (2 \cdot (u_i - 1) + 1) + 4)$. In Fig. 4, an example of this square topology is represented for $\boldsymbol{S} = [4]$ and $\boldsymbol{U} = [1, 2]$ leading to $d = 8 + 16 = 24$. For the special case where $\boldsymbol{U} = [1]$ and $\boldsymbol{S} = [1]$, we retrieve the 8 differential values computed in LBP. These relatively small DV are densely sampled on a regular grid with the same centers whatever the scale $s \in \boldsymbol{S}$. Typically, we sample $F = 10^5$ DV per scale. In order to compute efficiently the $\boldsymbol{z}^1$'s vectors (all A_i^s areas), we used the integral image technique [42].

Fig. 4 Differential vectors with a neighborhood composed of two concentric holed squares

3.2 A Two-Layer Architecture

Similarly to deep architectures, we use a two-layer stacked architecture. The first layer encodes the DV mini-patches while a more robust global representation is obtained by the second. Specifically, we chose the couple SC-MP as *encoder-pooler* for both layers.

In the SC approach, each vector z_t^j, $j \in \{1, 2\}$ is now expressed as a linear combination of a few vectors of the dictionary $\boldsymbol{D}^j$ of size $(d \times K^j)$. The associated sparse modeling problem, *i.e.* the offline estimation of $\boldsymbol{\theta}^j = \boldsymbol{D}^j$ given a random collection of local feature patches, is formulated using the following equation:

$$\arg\min_{\boldsymbol{D}^j, \boldsymbol{C}^j} \sum_{t=1}^{T} \|z_t^j - \boldsymbol{D}^j \boldsymbol{c}_t^j\|_2^2 + \beta \|\boldsymbol{c}_t^j\|_{\ell_1} \quad s.t. \quad \|\boldsymbol{c}_t^j\|_{\ell_1} = 1, \tag{13}$$

where the sparsity in controlled by the parameter β. At the end of the process, for each scale $s \in \boldsymbol{S}$, a trained dictionary $\widehat{\boldsymbol{D}}^j(s)$ is obtained. Given this trained dictionary $\widehat{\boldsymbol{D}}^j(s)$, T sparse codes $\boldsymbol{C}^j = \{\boldsymbol{c}_t^j(s)\}$, $t = 1, \ldots, T$ are computed by a LASSO procedure.

As depicted in Fig. 3, after the first encoder, we pooled the selected sparse codes set $\boldsymbol{C}^1$ over each window $\boldsymbol{W}^{1\to 2} \in \boldsymbol{\Lambda}^{1\to 2}$, where $\boldsymbol{\Lambda}^{1\to 2}$ designs the spatial pyramid (SP) configuration matrix from the first to the second layer.

For the current layer j, the current scale s and for the kth codeword, two following quantities are computed by MP:

$$\begin{cases} x^{k,j}(s) & \triangleq \max\limits_{t | z_t^j \subset \boldsymbol{W}^j} (|c_t^{k,j}(s)|), \ j = \{1, 2\} \\ z^{k,j+1}(s) & \triangleq \max\limits_{t | z_t^j \subset \boldsymbol{W}^{j\to j+1}} (|c_t^{k,j}(s)|), \ j = 1, \end{cases} \tag{14}$$

where $x^{k,j}(s)$ and $z^{k,j+1}(s)$ defines the output of the jth layer and the input of the j^{t+1} layer respectively. For the second pooling stage, input z_t^2, $t = 1, \ldots, V^{1\to 2}$ will be assumed to be centered in the middle of $\boldsymbol{W}_t^{1\to 2}$ and moreover, we will also assume that for each $\boldsymbol{W}_p^2$, $p = 1, \ldots, V^2$ at least exist a t such $\boldsymbol{W}_t^{1\to 2} \subset \boldsymbol{W}_p^2$.

The final global feature is constructed by concatenating all $\{x^{k,2}(s)\}$ into a unique vector of length $f^2 = \mathcal{S} \cdot V^2 \cdot K^2$. In order to have a multi-level analysis, the first layer global representation can be also concatenated with the second layer one.

4 Experimental Results

We test our two BoS frameworks on Scene-15 [28], UIUC-Sport [48]. For each dataset, we assume available a training data set $\{\boldsymbol{x}_i(\boldsymbol{\Lambda}), y_i\}_{i=1}^N$, where $\boldsymbol{x}_i(\boldsymbol{\Lambda}) \in \mathbb{R}^f$ is one of our previously defined features and $y_i \in \{1, \ldots, M\}$, where M is the number

of classes. As in [30, 43], we will use a simple large-scale linear SVM such as LIBLINEAR [40] with the 1-vs-all multi-class strategy.

For all datasets, we used SIFT patches with block size (16×16) pixels and (26×26) pixels for our HB/HIB/HT/HIT respectively. For HT/HIT patches, we fix $t = 1$. For SIFT/HB/HIB/HT/HIT, we extract $F = 35 \cdot 35 = 1{,}225$ patches per scale while for DV patches, we extracted $F = 120 \cdot 120 = 14{,}400$ patches per scale and $\boldsymbol{S} = [1, 2, 3]$, $\boldsymbol{U} = [1, 2]$. We fixed $\boldsymbol{\Lambda}^{1\rightarrow 2} = \left[\frac{1}{9}\ \frac{1}{9}\ \frac{1}{27}\ \frac{1}{27}\ 1\right]$. For both dictionary learning and sparse codes computation, we fix $\beta = 0.2$ and $N_{ite} = 50$ iterations to train dictionaries. We use our own modified version of the SPAMS toolbox [45]. Finally, we performed a 10 cross-validation to compute the average overall accuracy and its standard deviation using the LIBLINEAR solver and fixing parameter $C = 15$.

4.1 Scene-15 Dataset

The Scene-15 dataset contains a total of 4,485 images in grey color assigned to $M = 15$ categories. The number of images in each category ranges from 200 to 400. 100 images per class are used to train, the rest for testing.

For HT/HIT patches, we select 225,000 patches to train dictionaries and pooling is performed with $\boldsymbol{\Lambda} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{6} & \frac{1}{6} & 1 \end{bmatrix}$. For DV patches, we select also 225,000 patches to train dictionaries for the first layer and 75,000 to train dictionaries for the second layer. Second pooling is performed with $\boldsymbol{\Lambda}^2 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{6} & \frac{1}{6} & 1 \end{bmatrix}$.

In Fig. 5, we plot accuracy versus the number of words K in the dictionary training. With our particular choice of $\boldsymbol{\Lambda}$ and for one unique scale, we retrieved results comparable to [30], *i.e.* 80.28 % versus 81.24 % for our implementation. Regardless the number of scale used and the type of patch, our BoS framework outperforms the SIFT-ScSPM approach. In Table 1, we compare our results with the state-of-the-art for this dataset (with $S = 4$ scales). The best performance is actually obtained with the SIFT-LScSPM involving a more sophisticated dictionary training through the Laplacian sparse coding. The latter is very time and memory consuming[2] but allows us an increase with normal SIFT patch from 80.28 % $\pm$ 0.93 with simple Sc to 89.75 % $\pm$ 0.5 with LSc. The second best result is obtained with spatial FV followed by the kernel descriptors. For FV, they reduced SIFT to 64 dimension (total size equal to $K(1 + 2 \cdot d) = 12800$) and used a multi-class logistic regression. It is also worth noting that KDES-EKM uses a concatenation of 3 descriptors coupled with an efficient feature mapping (KDES-A+LSVM got 81.9 % $\pm$ 0.60 for a fair comparison). However, our results with a HIT patch and a simple linear SVM are very close (86.53 %) while requiring sparse coding only for the second layer. Our EBoS with DV patches is improving results by almost 1 % but with more computational efforts (for training/encoding both layers).

[2] LSc requiers to store sparse codes of the template set, *i.e*, a sparse matrix ($K \times N_{template}$).

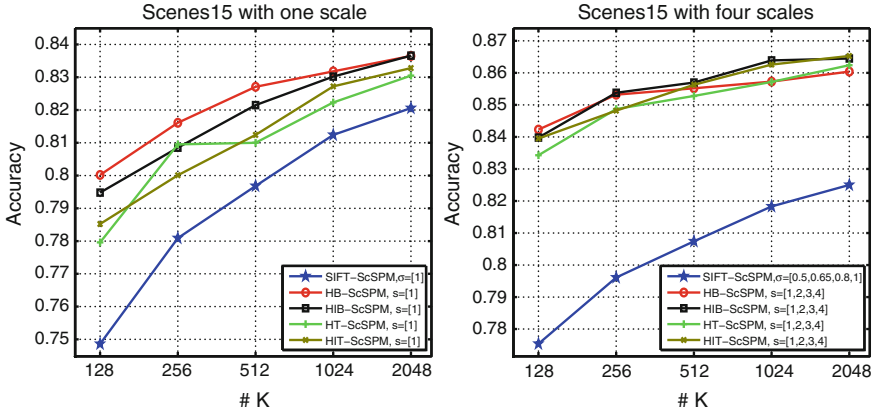


Fig. 5 Results for Scenes 15. *Left* one scale are used for all kind of patches. *Right* four scales are used for all kind of patches

Table 1 Recognition rate (and standard deviation) for Scene-15 dataset

Algorithms	Accuracy ± Std
SIFT-ScSPM ($K = 1024$) [30]	80.28 % ± 0.93
SIFT-ScSPM ($K = 1024$, our implementation)	81.24 % ± 0.73
SIFT-MidLevel ($K = 2048$) [43]	84.20 % ± 0.30
SIFT-LScSPM ($K = 1024$) [14]	**89.75% ± 0.50**
KDES-EKM ($K = 1000$) [49]	*86.70%*
PCASIFT-SFV ($K = 100$) [37]	**88.20%**
SIFT-DITC ($K = 1000$) [50]	85.4 %
HB-ScSPM ($K = 2048$, our work)	86.04 %±0.36
HIB-ScSPM ($K = 2048$, our work)	86.45 %±0.44
HT-ScSPM ($K = 2048$, our work)	86.24 %±0.43
HIT-ScSPM ($K = 2048$, our work)	**86.53% ± 0.37**
DV-Sc2SPM2 ($K^1 = 2048$, $K^2 = 2048$, our work)	**87.46% ± 0.57**

4.2 UIUC-Sport Dataset

The UIUC-sport dataset contains a total of 1,579 images assigned to $M = 8$ categories. 60 images per class are used to train, 70 for testing. For this dataset, we define $\boldsymbol{\Lambda} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 1 \end{bmatrix}$. Color (R, G, B) information channels are used, sampling patches and training dictionaries on each of them. For HT/HIT patches, we fix $t = 5$. We select 240,000 patches to train dictionaries. For DV patches, we select also 225,000

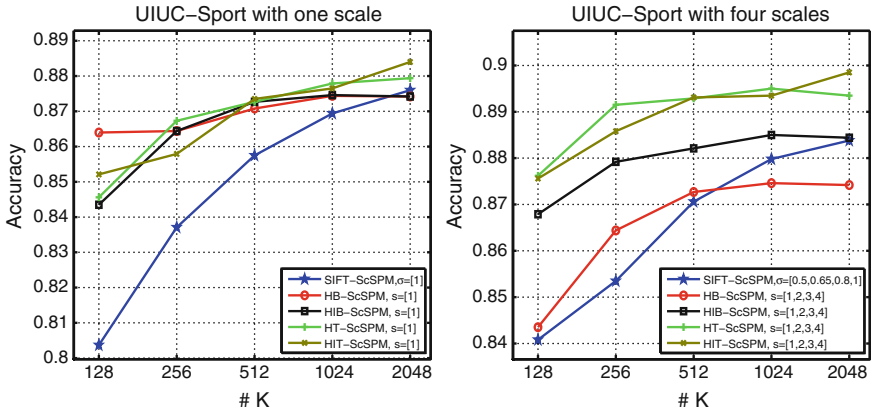


Fig. 6 Results for UIUC-Sport. *Left* one scale are used for all kind of patches. *Right* four scales are used for all kind of patches

Table 2 Recognition rate (and standard deviation) for UIUC-Sport dataset

Algorithms	Accuracy ± Std
SIFT-ScSPM ($K = 1024$) [30]	82.70 % ± 1.50
SIFT-ScSPM ($K = 1024$, our implementation)	87.98 % ± 1.08
SIFT-LScSPM ($K = 1024$) [14]	85.30 % ± 0.31
SIFT-HOMP ($K = 2 \times 1024$) [38]	85.70 % ± 1.30
HB-ScSPM ($K = 2048$, our work)	87.42 %±1.27
HIB-ScSPM ($K = 2048$, our work)	88.44 %±1.25
HT-ScSPM ($K = 2048$, our work)	89.35 %±1.42
HIT-ScSPM ($K = 2048$, our work)	**89.85**% ± **1.28**
DV-Sc2SPM2 ($K^1 = 2048, K^2 = 2048$, our work)	**90.38**% ± **1.05**

patches to train dictionaries for the first layer and 75,000 to train dictionaries for the second layer. Second pooling is performed with $\boldsymbol{\Lambda}^2 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 1 \end{bmatrix}$.

In Fig. 6, we plot accuracy versus. K. Notice, that our implementation of SIFT-ScSPM outperforms results from [30]. Our choice of $\boldsymbol{\Lambda}$, color information used in training and our specific normalization procedure may explain these improved results. We can also notice, especially for a small dictionary size, that our BoS framework is far superior to SIFT-ScSPM. In Table 2, we compare our results with the state-of-the-art (with $S = 4$ scales). To our best of knowledge, Both FBoS and EBoS frameworks, with HIT patches or DV patches, are obtaining the state-of-the-art performances with **89.85 / 90.38** % respectively of overall accuracy.

5 Conclusions and Perspectives

We have presented in this article FBoS and EBoS architectures. For FBos, HB/HIB /HT/HIT patches are used as a fast local textures encoder and Sc as scenes encoder. This first hand-crafted layer can advantageously replace complex hierarchical feature extractors such as Deep Belief Networks and the patch extraction phase is even faster than the SIFT one, attributed mainly to the integral image technique. Achieved performances outperform state-of-the-art results with a simple linear SVM for object recognition tasks. For EBos, a first layer is trained to encode the dataset specific differential vectors. Some relative performance improvements are obtained compared to FBoS with EBoS.

As potential future work, many perspectives can be investigated. For example, experimenting with LSc [14] or FV [37] should improve the encoding part of the pipeline, while supervised pooling techniques [51] will surely also improve results.

References

1. Bosch, A., Zisserman, A., Munoz, X.: Image classification using random forests and ferns. In: ICCV'07 (2007)
2. Larios, N., Lin, J., Zhang, M., Lytle, D., Moldenke, A., Shapiro, L., Dietterich, T.: Stacked spatial-pyramid kernel: an object-class recognition method to combine scores from random trees. In: WACV'11 (2011)
3. Oliva, A., Torralba, A.: Modeling the shape of the scene: a holistic representation of the spatial envelope. Int. J. Comput. Vis. **42**, 145–175 (2001)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR'05 (2005)
5. Deselaers, T., Ferrari, V.: Global and efficient self-similarity for object classification and detection. In: CVPR'10 (2010)
6. Chen, J., Shan, S., He, C., Zhao, G., Pietikainen, M., Chen, X., Gao, W.: Wld: a robust local image descriptor. IEEE Trans. PAMI **32**(9), 1705–1720 (2010)
7. Fröba, B., Ernst, A.: Face detection with the modified census transform. In: FGR'04 (2004)
8. Wu, J., Geyer, C., Rehg, J.M.: Real-time human detection using contour cues. In: ICRA'11 (2011)
9. Marcel, S., Rodriguez, Y., Heusch, G.: On the recent use of local binary patterns for face authentication. Int. J. Image Video Process. Spec. Issue Facial Image Process. 1–9 (2007)
10. Zhang, L., Chu, R., Xiang, S., Liao, S., Li, S.Z.: Face detection based on multi-block lbp representation. In: ICB'07 (2007)
11. Sadat, R.M.N., Teng, S.W., Lu, G., Hasan, S.F.: Texture classification using multimodal invariant local binary pattern. In: WACV'11 (2011)
12. Bianconi, F., González, E., Fernández, A., Saetta, S.A.: Automatic classification of granite tiles through colour and texture features. Expert Syst. Appl. **39**(12), 11212–11218 (2012)
13. Wu, J., Rehg, J.M.: Where am i: place instance and category recognition using spatialpact. In: CVPR'2008 (2008)
14. Gao, S., Tsang, I.W.-H., Chia, L.-T., Zhao, P.: Local features are not lonely Laplacian sparse coding for image classification. In: CVPR'10 (2010)
15. Paris, S., Glotin, H.: Pyramidal multi-level features for the robot vision@icpr 2010 challenge. In: ICPR'10 (2010)

16. Zhang, B., Gao, Y., Zhao, S., Liu, J.: Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor. IEEE Trans. Image Proc. **19**(2), 533–544 (2010)
17. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. PAMI **24**(7), 971–987 (2002)
18. Zheng, Y., Shen, C., Hartley, R.I., Huang, X.: Effective pedestrian detection using center-symmetric local binary/trinary patterns. In: CoRR, vol. abs/1009.0892 (2010)
19. Zhang, W., Shan, S., Qing, L., Chen, X., Gao, W.: Are gabor phases really useless for face recognition? Pattern Anal. Appl. **12**(3), 301–307 (2009)
20. Lee, H., Chung, Y., Kim, J., Park, D.: Face image retrieval using sparse representation classifier with gabor-lbp histogram. In: WISA'10 (2010)
21. Jun, B., Kim, D.: Robust face detection using local gradient patterns and evidence accumulation. Pattern Recognit. **45**, 3304–3316 (2012)
22. Hussain, S.U., Triggs, W.: Visual recognition using local quantized patterns. In: CVPR'12 (2012)
23. Heikkilä, M., Pietikäinen, M., Schmid, C.: Description of interest regions with center-symmetric local binary patterns. In: CVGIP'06 (2006)
24. Huang, D., Shan, C., Ardabilian, M., Wang, Y., Chen, L.: Local binary patterns and its application to facial image analysis: a survey. IEEE Trans. Syst. Man Cybern. Part C: Appl. Rev. **41**, 1–17 (2011)
25. Bianconi, F., Fernández, A.: On the occurrence probability of local binary patterns: a theoretical study. J. Math. Imaging Vis. **40**(3), 259–268 (2011)
26. Tan, X., Triggs, B.: Enhanced local texture feature sets for face recognition under difficult lighting conditions. Trans. Image Proc. **19**, 1635–1650 (2010)
27. Willamowski, J., Arregui, D., Csurka, G., Dance, C.R., Fan, L.: Categorizing nine visual classes using local appearance descriptors. In: ICPR'04 (2004)
28. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: CVPR'06 (2006)
29. Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV'99 (2009)
30. Yang, J., Yu, K., Gong, Y., Huang, T.S.: Linear spatial pyramid matching using sparse coding for image classification. In: CVPR'09 (2009)
31. Sermanet, P., Chintala, S., LeCun, Y.: Convolutional neural networks applied to house numbers digit classification. In: ICPR'12 (2012)
32. Wu, J., Rehg, J.: Beyond the euclidean distance: creating effective visual codebooks using the histogram intersection kernel. In: ICCV'09 (2009)
33. Boiman, O., Shechtman, E., Irani, M.: In defense of nearest-neighbor based image classification. In: CVPR'08 (2008)
34. Avila, S.E.F., Thome, N., Cord, M., Valle, E., de Albuquerque Araújo, A.: Bossa: extended bow formalism for image classification. In: ICIP'11 (2011)
35. Oliveira, G.L., Nascimento, E.R., Viera, A.W., Campos, M.F.M.: Sparse spatial coding: a novel approach for efficient and accurate object recognition. In: ICRA'12 (2012)
36. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: ECCV'10 (2010)
37. Krapac, J., Verbeek, J., Jurie, F.: Modeling spatial layout with fisher vectors for image categorization. In: ICCV'11 (2011)
38. Bo, L., Ren, X., Fox, D.: Hierarchical matching pursuit for image classification: architecture and fast algorithms. In: NIPS'11, pp. 2115–2123 (2011)
39. Shalev-Shwartz, S., Singer, Y., Srebro, N., Cotter, A.: Pegasos: primal estimated sub-gradient solver for svm (2007)
40. Hsieh, C., Chang, K., Lin, C., Keerthi, S.: A dual coordinate descent method for large-scale linear svm (2008)
41. Liao, S., Zhu, X., Lei, Z., Zhang, L., Li, S.Z.: Learning multi-scale block local binary patterns for face recognition. In: ICB (2007)
42. Viola, P., Jones, M.: Robust real-time face detection. Int. J. Comput. Vis. **57**, 137–154 (2004)

43. Boureau, Y., Bach, F., LeCun, Y., Ponce, J.: Learning mid-level features for recognition. In: CVPR'10 (2010)
44. Chatfield, K., Lempitsky, V., Vedaldi, A., Zisserman, A.: The devil is in the details: an evaluation of recent feature encoding methods. In: BMVC (2011)
45. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: ICML'09 (2009)
46. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. Ser. B **58**, 267–288 (1996)
47. Boureau, Y., Ponce, J., LeCun, Y.: A theoretical analysis of feature pooling in vision algorithms. In: ICML'10 (2010)
48. Li, L.: What, where and who? Classifying event by scene and object recognition. In: CVPR'07 (2007)
49. Bo, L., Ren, X., Fox, D.: Kernel descriptors for visual recognition. In: NIPS'10 (2010)
50. Elfiky, N.M., Khan, F.S., van de Weijer, J., Gonzàlez, J.: Discriminative compact pyramids for object and scene recognition. Pattern Recognit. **45**(4), 1627–1636 (2012)
51. Jia, Y., Huang, C., Darrell, T.: Beyond spatial pyramids: receptive field learning for pooled image features. In: NIPS'11 (2011)

Unsupervised Learning of Semantics of Object Detections for Scene Categorization

Grégoire Mesnil, Salah Rifai, Antoine Bordes, Xavier Glorot, Yoshua Bengio and Pascal Vincent

Abstract Classifying scenes (e.g. into "street", "home" or "leisure") is an important but complicated task nowadays, because images come with variability, ambiguity, and a wide range of illumination or scale conditions. Standard approaches build an intermediate representation of the global image and learn classifiers on it. Recently, it has been proposed to depict an image as an aggregation of its contained objects: the representation on which classifiers are trained is composed of many heterogeneous feature vectors derived from various object detectors. In this paper, we propose to study different approaches to efficiently learn contextual semantics out of these object detections. We use the features provided by Object-Bank [24] (177 different object detectors producing 252 attributes each), and show on several benchmarks for scene categorization that careful combinations, taking into account the structure of the data, allows to greatly improve over original results (from +5 to +11 %) while drastically reducing the dimensionality of the representation by 97 % (from 44,604 to 1,000). We also show that the uncertainty relative to object detectors hampers the use of external semantic knowledge to improve detectors combination, unlike our unsupervised learning approach.

Keywords Unsupervised learning · Transfer learning · Deep learning · Scene categorization · Object detection

1 Introduction

Automatic scene categorization is crucial for many applications such as content-based image indexing [37] or image understanding. This is defined as the task of assigning images to predefined categories ("office", "sailing", "mountain", etc.).

G. Mesnil (✉) · S. Rifai · X. Glorot · Y. Bengio · P. Vincent
LISA, Université de Montréal, Québec, Canada
e-mail: gregoire.mesnil@gmail.com

G. Mesnil
LITIS, Université de Rouen, Rouen, France

A. Bordes
CNRS - Heudiasyc UMR 7253, Université de Technologie de Compiègne, Compiègne, France

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_13

Classifying scene is complicated because of the large variability of quality, subject and conditions of natural images which lead to many ambiguities w.r.t. the corresponding scene label.

Standard methods build an intermediate representation before classifying scenes by considering the image as a whole [10, 28, 38, 40]. In particular, many such approaches rely on power spectral information, such as magnitude of spatial frequencies [28] or local texture descriptors [10]. They have shown to perform well in cases where there are large numbers of scene categories.

Another line of work conveys promising potential in scene categorization. First applied to object recognition [9], attribute-based methods have now proved to be effective for dealing with complex scenes. These models define high-level representations by combining semantic lower-level elements, e.g., detection of object parts. A precursor of this tendency for scenes was an adaptation of pLSA [15] to deal with "visual words" proposed by [5]. An extension of this idea consists in modeling an image based on its content i.e., its objects [7, 24]. Hence, the Object-Bank (OB) project [25] aims at building high-dimensional over-complete representations of scenes (of dimension 44,604) by combining the outputs of many object detectors (177) taken at various poses, scales and positions in the original image (leading to 252 attributes per detector). Experimental results indicate that this approach is effective since simple classifiers such as Support Vector Machines trained on their representations achieve state-of-the-art performance. However, this approach suffers from two flaws: (1) curse of dimensionality (very large number of features) and (2) individual object detectors have a poor precision (30 % at most). To solve (1), the original paper proposes to use structured norms and group sparsity to make best use of the large input. Our work studies new ways to combine the very rich information provided by these multiple detectors, dealing with the uncertainty of the detections. A method designed to select and combine the most informative attributes would be able to carefully manage redundancy, noise and structure in the data, leading to better scene categorization performance.

Hence, in the following, we propose a sequential 2-steps strategy for combining the feature representations provided by the OB object detectors on which the linear SVM classifier is destined to be trained for categorizing scenes. The first step adapts Principal Components Analysis (PCA) to this particular setting: we show that it is crucial to take into account the structure of the data in order for PCA to perform well. The second one is based on *Deep Learning*. Deep Learning has emerged recently (see [3] for a review) and is based on neural network algorithms able to discover data representations in an unsupervised fashion [2, 14, 18, 19, 32]. We propose to use this ability to combine multiple detector features. Hence, we present a model trained using Contractive Auto-Encoders [33, 34], which have already proved to be efficient on many image tasks and has contributed to winning a transfer learning challenge [26].

We validate the quality of our models in an extensive set of experiments in which several setups of the sequential feature extraction process are evaluated on benchmarks for scene classification [21, 23, 31, 41]. We show that our best results substantially outperform the original methods developed on top of OB features, while

producing representations of much lower dimension. The performance gap is usually large, indicating that advanced combinations are highly beneficial. We show that our method based on dimensionality reduction followed by deep learning offers a flexibility which makes it able to benefit from semi-supervised and transfer learning.

2 Scene Categorization with Object-Bank

Let us begin by introducing the approach of the OB project [24]. First, the 177 most useful (or frequent) objects were selected from popular image datasets such as LabelMe [35], ImageNet [6] and Flickr. For each of these 177 objects, a specific detector, existing in the literature [11, 16], was trained. Every detector is composed of 2 *root filters* depending on the pose, each one coming with its own deformable pattern of parts, e.g., there is one root filter for the front-view of a bike and one for the side-view. These $354 = 177 \times 2$ part-based filters (each composed by a root and its parts) are used to produce features of natural images. For a given image, a filter is convolved at 6 different scales. At each scale, the max-response among $21 = 1 + 4 + 16$ positions (whole image, quadrants, quadrants within each quadrant) is kept, producing a response map of dimension $126 = 6 \times 21$. All 2×177 maps are finally concatenated to produce an over-complete representation $x \in \mathbb{R}^{44,604}$ of the original image.

In the original OB paper [24], classifiers for scene categorization are learned directly on these feature vectors of dimension 44,604. More precisely, C classifiers (Linear SVM or Logistic Regression) are trained in a 1-versus-all setting in order to predict the correct scene category $y_{\text{category}}(x)$ among C different categories. Various strategies using structured sparsity with combinations of ℓ_1/ℓ_2 norms have been proposed to handle the very large input.

3 Unsupervised Feature Learning

The approach of OB for the task of scene categorization, based on specific object detectors, is appealing since it works well in practice. This suggests that a scene is better recognized by first identifying basic objects and then exploiting the underlying semantics in the dependencies between the corresponding detectors.

However, it appears that none of the individual object detectors reaches a recognition precision of more than 30 %. Hence, one may question whether the ideal view that inspired this approach (and expressed above) is indeed the reason of OB's success. Alternatively, one may hypothesize that the 44,604 OB features are more useful for scene categorization because they represent high level statistical properties of images than because they precisely report the absence/presence of objects—see Fig. 1. OB tried structured sparsity to handle this feature selection but there may be other ways—simpler or not.

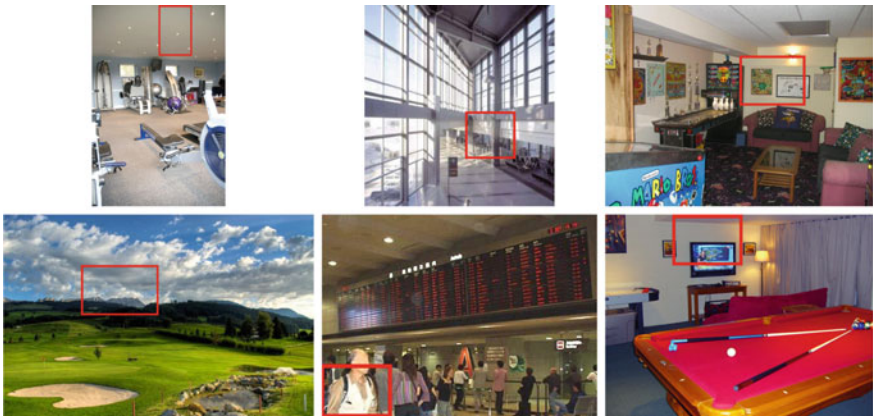


Fig. 1 *Left* Cloud *Middle* Man *Right* Television. *Top* False Detections *Bottom* True Detections. Images from **SUN** [41] for which we compute the OB representation and display the bounding box around the average position of various objects detectors. For instance, the *television* detector can be viewed either as a *television* detector or a *rectangle* shape detector i.e. high-order statistical properties of the image

This paper investigates several ways of learning higher-level features **on top of** the high dimensional representation provided by OB, expecting that capturing further structure may improve categorization performance. Our approach employs *unsupervised feature learning/extraction* algorithms, i.e. generic feature extraction methods which were not developed specifically for images. We will consider both standard Principal Component Analysis and Contractive Auto-Encoders [33, 34]. The latter is a recent machine learning method which has proved to be a robust feature extraction tool.

3.1 Principal Component Analysis

Principal Component Analysis (PCA) [17, 30] is the most prevalent technique for linear dimensionality reduction. A PCA with k components finds the k orthonormal directions of projection in input space that retain most of the *variance* of the training data. These correspond to the eigenvectors associated with the leading eigenvalues of the training data's covariance matrix. Principal components are ordered, so that the first corresponds to the direction along which the data varies the most (largest eigenvalue), etc…

Since we will consider an auto-encoder variant (presented next), we should mention here a well-known result: a linear auto-encoder with k hidden units, trained to minimize squared reconstruction error, will learn projection directions that span the same *subspace* as a k component PCA [1]. However the regularized non-linear

auto-encoder variant that we consider below is capable of extracting qualitatively different, and usually more useful, nonlinear features.

3.2 Contractive Auto-Encoders

Contractive Auto-Encoders (CAEs) [33, 34] are among the latest developments in a line of machine learning research on nonlinear feature learning methods, that started with the success of Restricted Boltzmann Machines [14] for pre-training deep networks, and was followed by other variants of auto-encoders such as sparse [13, 19, 32] and denoising auto-encoders [39]. It was selected here mainly due to its practical ease of use and recent empirical successes.

Unlike PCA that decomposes the input space into leading *global* directions of variations, the CAE learns features that capture local directions of variation (in some regions of input space). This is achieved by penalizing the norm of the Jacobian of a latent representation $h(x)$ with respect to its input x at training samples. In [34], authors show that the resulting features provide a local coordinate system for a low dimensional manifold of the input space. This corresponds to an atlas of charts, each corresponding to a different region in input space, associated with a different set of active latent features. One can think about this as being similar to a mixture of PCAs, each computed on a different set of training samples that were grouped together using a similarity criterion (and corresponding to a different input region), but without using an independent parametrization for each component of the mixture, i.e., allowing to generalize across the charts, and away from the training examples.

In the following, we summarize the formulation of the CAE as a regularized extension of a basic Auto-Encoder (AE). In our experiments, the parametrization of this AE consists in a non-linear encoder or latent representation h of m hidden units with a linear decoder or reconstruction g towards an input space of dimension d.

Formally, the latent variables are parametrized by:

$$h(x) = s(Wx + b_h), \tag{1}$$

where s is the element-wise logistic sigmoid $s(z) = \frac{1}{1+e^{-z}}$, $W \in \mathcal{M}_{m\times d}(\mathbb{R})$ and $b_h \in \mathbb{R}^m$ are the parameters to be learned during training. Conversely, the units of the decoder are linear projections of $h(x)$ back into the input space:

$$g(h(x)) = W^T h(x). \tag{2}$$

Using mean squared error as the reconstruction objective and the L2-norm of the Jacobian of h with respect to x as regularization, training is carried out by minimizing the following criterion by stochastic gradient descent:

$$\mathcal{J}_{\mathrm{CAE}}(\Theta) = \sum_{x\in\mathcal{D}} \|x - g(h(x))\|^2 + \lambda \sum_{i=1}^{m}\sum_{j=1}^{d} \left|\frac{\partial h_i}{\partial x_j}(x)\right|^2, \tag{3}$$

where $\Theta = \{W, b_h\}$, $\mathcal{D} = \{x^{(i)}\}_{i=1,\ldots,n}$ corresponds to a set of n training samples $x \in \mathbb{R}^d$ and λ is a hyper-parameter controlling the level of contraction of h. A notable difference between CAEs and PCA is that features extracted by CAEs are non-linear w.r.t. the inputs, so that multiple layers of CAEs can be usefully composed (stacked), whereas stacking linear PCAs is pointless.

4 Extracting Better Features with Advanced Combination Strategies

In this work, we study two different sub-structures of OB. We consider the *pose* response defined by the output of only one part-based filter at all positions and scales, and the *object* response which is the concatenation of all *pose* responses associated to an object. Combination strategies are depicted in Fig. 2.

4.1 Simplistic Strategies: Mean and Max Pooling

The idea of pooling responses at different locations or poses has been successfully used in Convolutional Neural Networks such as LeNet-5 [22] and other visual processing [36] architectures inspired by the visual cortex.

Here, we pool the 252 responses of each object detector into one component (using the mean or max operator) leading to a representation of size $177 = 44{,}604/252$. It corresponds to the mean/max over the object responses at different scales and locations. One may view the object max responses as features encoding absence/presence of objects while discarding all the information about the detector's positions.

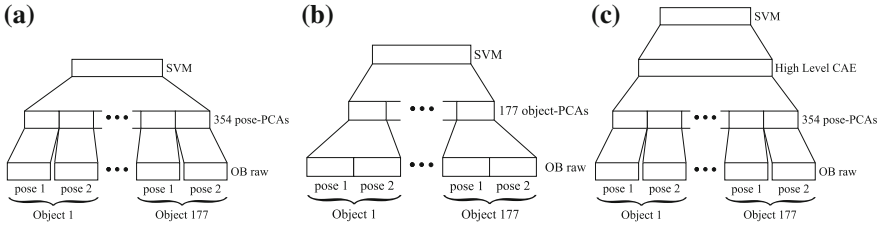


Fig. 2 Different Combination Strategies (**a**) and (**b**) *pose* and *object* PCAs (**c**) high-level CAE: *pose*-PCA as dimensionality reduction technique in the first layer and a CAE stacked on top. We denote it high-level because it can learn *context information* i.e. plausible joint appearance of different objects

4.2 Combination Strategies with PCA

PCA is a standard method for extracting features from high dimensional input, so it is a good starting point. However, as we find in our experiments, exploiting the particular structure of the data, e.g., according to poses, scales, and locations, can yield to improved results.

Whole PCA. An ordinary PCA is trained on the raw output of OB ($x \in \mathbb{R}^{44,604}$) without looking for any structure. Given the high-dimensionality of OB's representation, we used the Randomized PCA algorithm of the scikits toolbox.[1]

Pose-PCA. Each of the two *poses* associated with each *object* detector is considered independently. This results in $354 = 2 \times 177$ different PCAs, which are trained on *pose* outputs ($x \in \mathbb{R}^{126}$)—see Fig. 2.

Object-PCA. Only each *object* response ($x \in \mathbb{R}^{252}$) is considered separately, therefore 177 PCAs are trained in total. It allows the model to capture variations among all *pose* responses at various scales and positions—see Fig. 2.

Note that, in all cases, whitening the PCA (i.e. dividing each eigenvector's response by the corresponding squared root eigenvalue) performs very poorly. For post-processing, the PCA outputs $\tilde{x}$ are always normalized: $\tilde{x} \leftarrow (\tilde{x} - \mu)/\sigma$ according to mean μ and the deviation σ of the whole, per *object* or per *pose* PCA outputs. Thereby, we ensure contributions from all *objects* or *poses* to be in the same range. The number of components in all cases has been selected according to the classification accuracy estimated by 5-fold cross-validation.

4.3 Improving upon PCA with CAE

Due to hardware limitations and high-dimensional input, we could not train a CAE on the whole OB output ("whole CAE"). However, we address this problem with the sequential feature extraction steps below.

To overcome the tractability problem that forbids a CAE to be trained on the whole OB output, we preprocess it by using the *pose*-PCAs as a dimensionality reduction method. We keep only the 5 first components of each *pose*. Given this low-dimensional representation (of dimension $1,770$), we are able to train a CAE—see Fig. 2. The CAE has a global view of all object detectors and can thus learn to capture *context information*, defined by the joint appearance of combinations of various objects. Moreover, instead of using an SVM on top of the learned representations, we can use a Multi-Layer Perceptron whose weights would be initialized by those of this CAE. This setting is where the CAE has shown to perform best in practice [33].

[1] Available from http://scikits.appspot.com/.

5 Experiments

5.1 Datasets

We evaluate our approach on 3 scene datasets, cluttered indoor images (MIT Indoor Scene), natural scenes (15-Scenes), and event/activity images (UIUC-Sports). Images from a large scale scene recognition dataset (SUN-397 database) have also been used for unsupervised learning.

- **MIT Indoor** is composed of 67 categories and, following [24, 31], we used 80 images from each category for training and 20 for testing.
- **15-Scenes** is a dataset of 15 natural scene classes. According to [21], we used 100 images per class for training and the rest for testing.
- **UIUC-Sports** contains 8 event classes. We randomly chose 70 / 60 images for our training / test set respectively, following the setting of [23, 24].
- **SUN-397** contains a full variety of 397 well sampled scene categories (100 samples per class) composed of 108,754 images in total.

5.2 Tasks

We consider 3 different tasks to evaluate and compare the considered combination strategies. In particular, various supervision settings for learning the CAE are explored. Indeed, a great advantage of this kind of method is that it can make use of vast quantities of unlabeled examples to improve its representations. We thus illustrate this by proposing experiments in which the CAE has been trained in supervised or in semi-supervised way and also in a transfer context.

MIT Indoor (*plain*). Only the official training set of the MIT Indoor scene dataset (5,360 images) is used for unsupervised feature learning. Each representation is evaluated by training a linear SVM on top of the learned features.

MIT + SUN (*semi-supervised*). This task, like the previous one, uses the official train/test split of the MIT Indoor scene dataset for its supervised training and evaluation of scene categorization performance. For the initial unsupervised feature extraction however, we augmented the MIT Indoor training set with the whole dataset of images from SUN-397 (108,754 images). This yields a total of 123,034 images for unsupervised feature learning and corresponds to a *semi-supervised* setting. Our motivation for adding scene images from SUN, besides increasing the number of training samples, is that on MIT Indoor, which contains only indoor scenes, OB detectors specialized on outdoor objects would likely be mostly inactive (as a sailboat detector applied on indoor scenes) and irrelevant, introducing an harmful noise in the unsupervised feature learning. As SUN is composed of a wide range of indoor and outdoor scene images, its addition to MIT Indoor ensures that each detector

meaningfully covers its whole range of activity (having a "balanced" number of positives/negatives detections through the training set) and the feature extraction methods can be efficiently trained to capture it.

One may object that training on additional images does not provide a fair comparison w.r.t. the original OB method. Nevertheless, we recall that (1) the supervised classifiers do not benefit from these additional examples and (2) object detectors which are the core of OB representations (and all detector-based approaches) have also obviously been trained on *additional* images.

UIUC-Sports and 15-Scenes (*transfer*). We would also like to evaluate the discriminative power of the various representations learned on the MIT + SUN dataset, but on new scene images and categories that were *not* part of the MIT + SUN dataset. This might be useful in case other researchers would like to use our compact representation on a different set of images. Using the representation output by the feature extractors learned with MIT+SUN, we train and evaluate classifiers for scene categorization on images from UIUC-Sports and 15-Scenes (not used during unsupervised training). This corresponds to a *transfer learning* setting for the feature extractors.

5.3 SVMs on Features Learned with Each Strategy

In order to evaluate the quality of the features generated by each strategy, a linear SVM is trained on the features extracted by each combination method. We used LibLinear [8] as SVM solver and chose the best C according to 5-fold cross-validation scheme. We compare accuracies obtained by features provided by all considered combination methods against the original OB performances [24]. Results obtained with SVM classifiers on all MIT-related tasks are displayed in Table 1 and those concerning UIUC and 15-scenes in Table 2.

The simplistic strategy *object* mean-pooling performs surprisingly well on all datasets and tasks whereas *object* max-pooling obtained the worst results. It suggests that taking the mean response of an object detector across various scales and positions is actually meaningful compared to consider presence/absence of objects as max-pooling does.

On MIT and MIT + SUN, *object* or *pose* PCAs reach almost the same range of performance slightly above the current state-of-the-art performances [29], except for whole-PCA which performs poorly: one must consider the structure of OB to combine features efficiently. In the experiments, keeping the 10 (resp. 15) first principal components gave us the best results for pose-PCA (resp. object-PCA).

Besides, Table 3 shows that both PCAs and PCA + CAE allow a huge reduction of the dimension of the OB feature representation.

Results obtained for the UIUC-Sports and 15-Scenes transfer learning tasks are displayed in Table 2. Representations learned on MIT + SUN generalize quite well and can be easily used for other datasets even if images from those datasets have not been seen at all during unsupervised learning.

Table 1 MIT Indoor

	MIT (*plain*) (%)	MIT+SUN (*semi-supervised*) (%)
object-MAX + SVM	24.3	–
object-MEAN + SVM	41.0	–
whole-PCA + SVM	40.2	–
object-PCA + SVM	42.6	46.1
pose-PCA + SVM	40.1	46.0
pose-PCA + MLP	42.9	46.3
pose-PCA + CAE (MLP)	**44.0**	**49.1**
Object Bank + SVM	37.6	–
Object Bank + rbf-SVM	37.7	–
DPM + Gist + SP	43.1	–
Improvement w.r.t. Object Bank	+6.4	+11.5

Results are reported on the official split [31] for all combination strategies described in Sect. 4. Only the unsupervised feature learning strategies (PCA and CAE based) *can* benefit from the addition of unlabeled scenes from SUN. Object Bank + SVM refers to the original system [24] and DPM + Gist + SP [29] corresponds to the state-of-the-art method on MIT Indoor

Table 2 UIUC Sports and 15-Scenes

	UIUC-Sports (%)	15-SCENES (%)
object-MAX + SVM	67.23 ± 1.29	71.08 ± 0.57
object-MEAN + SVM	81.88 ± 1.16	83.17 ± 0.53
object-PCA + SVM	83.90 ± 1.67	85.58 ± 0.48
pose-PCA + SVM	83.81 ± 2.22	85.69 ± 0.39
pose-PCA + MLP	84.29 ± 2.23	84.93 ± 0.39
pose-PCA + CAE (MLP)	$\mathbf{85.13 \pm 1.07}$	$\mathbf{86.44 \pm 0.21}$
Object Bank + SVM	78.90	80.98
Object Bank + rbf-SVM	78.56 ± 1.50	83.71 ± 0.64
Improvement w.r.t. OB	+6.23	+5.46

Results are reported for 10 random splits and compared to the original OB results [24]—Object Bank + SVM—on one single split

Table 3 Dimensionality reduction

Object-Bank	Pooling	*whole*-PCA	*object*-PCA	*pose*-PCA	*pose*-PCA+CAE
44,604	177	1,300	2,655	1,770	1,000

Dimension of representations obtained on MIT Indoor. The *pose*-PCA + CAE produces a compact and powerful combination

5.4 Deep Learning with Fine Tuning

Previous work [20] on Deep Learning generally showed that the features learned through unsupervised learning could be improved upon by fine-tuning them through a supervised training stage. In this stage (which follows the unsupervised pre-training stage), the features and the classifier on top of them are together considered to be a supervised neural network, a Multi-Layer Perception (MLP) whose hidden layer is the output of the trained features. Hence we apply this strategy to the *pose* PCA + CAE architecture, keeping the PCA transformation fixed but fine-tuning the CAE and the MLP altogether. These results are given at the bottom of Tables 1 and 2. The MLP are trained with early stopping on a validation set (taken from the original training set) for 50 epochs.

This yields 44.0 % test accuracy on plain MIT and 49.1 % on MIT + SUN: this allows to obtain state-of-the-art performance, with or without semi-supervised training of the CAEs, even if these additional examples are highly beneficial. As a check, we also evaluate the effect of the unsupervised pre-training stage by completely skipping it and only training a regular supervised MLP of 1,000 hidden units on top of the PCA output, yielding a worse test accuracy of 42.9 % on MIT and 46.3 % on MIT+SUN. This improvement with fine-tuning on labeled data is a great advantage for CAE compared to PCA. Fine-tuning is also beneficial on UIUC-Sports and 15-Scenes. On both datasets, this leads to an improvement of +6 and +5 % w.r.t the original system.

Finally, we trained a non-linear SVM (with rbf kernel) to verify whether this gap in performances was simply due to the replacement of a linear classifier (SVM) by a non-linear one (MLP) or to the detectors' outputs combination. The poor results of the rbf-SVM (see Tables 1 and 2) suggests that the careful combination strategies are essential to reach good performance (Table 4).

Table 4 *Context semantics*

Context Semantics learned by the CAE
Sailboat, rock, tree, coral, blind
Roller coaster, building, rail, keyboard, bridge
Sailboat, autobus, bus stop, truck, ship
Curtain, bookshelf, door, closet, rack
Soil, seashore, rock, mountain, duck
Attire, horse, bride, groom, bouquet
Bookshelf, curtain, faucet, screen, cabinet
Desktop computer, printer, wireless, computer screen

Names of the detectors corresponding to the highest weights of 8 hidden units of the CAE. These hidden units will fire when those objects will be detected altogether

5.5 Use of External Semantic Information for Re-ranking

WordNet's [27] semantic structure provides an easy way to measure word similarities. We assume that closely related objects detectors (according to WordNet) should fire together and could be grouped in order to build semantically meaningful features. E.g. by grouping the output of *ship*, *sea* and *sun* into a single feature, the combination's output might be useful for classifying the "sailing" scene category.

In our experiments, we used the lesk distance in WordNet to extract the neighbors of each detector's name. Some examples are depicted in Table 5. Afterwards, given the score $s(x) \in \mathbb{R}^{177}$ obtained with the mean-pooling strategy from the original OB representation $x \in \mathbb{R}^{44,604}$, we performed the following Re-Ranking operation:

$$s_i'(x) = \sum_{j=1}^{177} s_j(x)\gamma^{R(i,j)} \quad \text{for} \quad i = 1, \ldots, 177 \tag{4}$$

where $\gamma \in [0, 1]$ is a decay hyper-parameter tuned on a validation set. $R(i, j)$ corresponds to the rank of the object j among the neighbors of object i according to the lesk metric ($R(i, i) = 0$). Results are presented in Table 6. The relatively small improvement brought by WordNet illustrates the fact that the poor intrinsic quality of the object detectors prevents any use of external semantic resource to improve their combination.

Table 5 *WordNet semantics* Names of the detectors and their top-ranked neighbors according to the lesk distance computed from WordNet

Rank	Bus	Lion	Laptop
1.	Car	Tree	Baggage
2.	Ship	Dog	Desktop computer
3.	Truck	Bird	Computer
4.	Aircraft	Horse	Bed
5.	Train	Computer	Door

Table 6 *Re-Ranking* Results are reported on the official split [31]

object-MEAN+SVM	MIT (*plain*)
w/o Re-Ranking	41.03 %
with Re-Ranking	41.52 %

Object-mean+SVM refers to the mean-pooling strategy with and w/o the Re-Ranking transformation

6 Discussion

In this work, we add one or more levels of trained representations on top of the layer of object and part detectors (OB features) that have constituted the basis of very promising trend of approach for scene classification [24]. These higher-level representations are mostly trained in an unsupervised way, following the trend of so-called Deep Learning [3, 14, 18], but can be fine-tuned using the supervised detection objective.

These learned representations capture statistical dependencies in the co-occurrence of detections the object detectors from [24]. In fact, one can see in Table 4 plausible contexts of joint appearance of several objects learned by the CAE. These detectors, which can be quite imperfect when seen as actual detectors, contain a lot of information when combined altogether. However, the uncertainty of detectors makes it hard to combine using external semantic sources such as WordNet. As reported in Table 6, we observe a slight improvement (+0.5 %) using our Re-Ranking strategy and lesk words' similarities. The extraction of those *context semantics* with unsupervised feature-learning algorithms has empirically shown better performances but these semantics are inherent to the detectors outputs and can not be easily combined with any known predefined semantic system such as the one defined in WordNet.

In particular, we find that Contractive Auto-Encoder [33, 34] can substantially improve performance on top of *pose* PCAs as a way to extract non-linear dependencies between these lower-level OB detectors (especially when fine-tuned). They also improve greatly upon the use of the detectors as inputs to an SVM or a logistic regression (which were, with structured regularization, the original methods used by OB).

This trained post-processing allows us to reach the state-of-the-art on MIT Indoor and UIUC (85.13 % against 85.30 % obtained by LScSPM [12]) while being competitive on 15-scenes (86.44 % also versus 89.70 % LScSPM). On these last two datasets, we reach the best performance for methods only relying on object/part detectors. Compared to other kinds of methods, we are limited by the accuracy of those detectors (only trained on HOG features), whereas competitive methods can make use of other descriptors such as SIFT [12], known to achieve excellent performance in image recognition.

Besides its good accuracies, it is worth noting that the feature representation obtained by the *pose* PCA + CAE is also very compact, allowing a 97 % reduction compared to the original data (see Table 3). Handling a dense input of dimension 44,604 is not a common thing. By providing this compact representation, we think that researchers will be able to use the rich information provided by OB in the same way they use low-level image descriptors such as SIFT.

As future work, we are planning other ways of combining OB features e.g. considering the output of all detectors at a given scale and position and combine them afterwards in a hierarchical manner. This would be a kind of dual view of the OB features. Other plausible departures could take into account the topology (e.g. spatial structure) of the pattern of detections, rather than treat the response at each location

and scale as an attribute and the set of attributes as unordered. This could be done in the same spirit as in Convolutional Networks [22], aggregating the responses for various objects detectors/locations/scales in a way that takes explicitly into account the object category, location and scale of each response, similarly to the way filter outputs at neighboring locations are pooled in each layer of a Convolutional Network.

Acknowledgments We would like to thank Gloria Zen for her helpful comments. This work was supported by NSERC, CIFAR, the Canada Research Chairs, Compute Canada and by the French ANR Project ASAP ANR-09-EMER-001. Codes for the experiments have been implemented using Theano [4] Machine Learning library.

References

1. Baldi, P., Hornik, K.: Neural networks and principal component analysis: learning from examples without local minima. Neural Netw. **2**, 53–58 (1989)
2. Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy layer-wise training of deep networks. Adv. Neural Inf. Proc. Sys. **19**, 153–160 (2007)
3. Bengio, Y.: Learning deep architectures for AI. Found. Trends Mach. Learn. **2**(1), 1–127 (2009). Also published as a book. Now Publishers, 2009
4. Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., Bengio, Y.: Theano: a CPU and GPU math expression compiler. In: Proceedings of the Python for Scientific Computing Conference (SciPy), June 2010. Oral Presentation
5. Bosch, A., Zisserman, A., Muñoz, X.: Scene classification via plsa. In: In Proceedings of the ECCV, pp. 517–530 (2006)
6. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: CVPR09 (2009)
7. Espinace, P., Kollar, T., Soto, A., Roy, N.: Indoor scene recognition through object detection. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Anchorage, AK (2010)
8. Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., Lin, C.-J.: Liblinear: a library for large linear classification. J. Mach. Learn. Res. **9**, 1871–1874 (2008)
9. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1778–1785 (2009)
10. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)—Volume 2—Volume 02, CVPR'05, pp. 524–531. IEEE Computer Society (2005)
11. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discrimitatively trained, multiscale, deformable part model. In: CVPR (2008)
12. Gao, S., Tsang, I., Chia, L., Zhao, P.: Local features are not lonely laplacian sparse coding for image classification. In: IEEE Conference on Computer Vision and Pattern Recognition (2010)
13. Goodfellow, I., Le, Q., Saxe, A., Ng, A.: Measuring invariances in deep networks. In: NIPS'09, pp. 646–654 (2009)
14. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. Neural Comput. **18**, 1527–1554 (2006)
15. Hofmann, T.: Unsupervised learning by probabilistic latent semantic analysis. Mach. Learn. **42**, 177–196 (2001)
16. Hoiem, D., Efros, A.A., Hebert, M.: Automatic photo pop-up. SIGGRAPH **24**(3), 577584 (2005)

17. Hotelling, H.: Analysis of a complex of statistical variables into principal components. J. Educ. Psychol. **24**, 417–441, 498–520 (1933)
18. Jarrett, K., Kavukcuoglu, K., Ranzato, M., LeCun, Y.: What is the best multi-stage architecture for object recognition?. In: Proceedings of the International Conference on Computer Vision (ICCV'09), pp. 2146–2153. IEEE (2009)
19. Kavukcuoglu, K., Ranzato, M., Fergus, R., LeCun, Y.: Learning invariant features through topographic filter maps. In: Proceedings of the CVPR'09, pp. 1605–1612. IEEE (2009)
20. Larochelle, H., Bengio, Y., Louradour, J., Lamblin, P.: Exploring strategies for training deep neural networks. JMLR **10**, 1–40 (2009)
21. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: IEEE Conference on Computer Vision and Pattern Recognition (2006)
22. LeCun, Y., Haffner, P., Bottou, L., Bengio, Y.: Object recognition with gradient-based learning. In: Shape, Contour and Grouping in Computer Vision, pp. 319–345. Springer (1999)
23. Li, L.-J., Fei-Fei, L.: What, where and who? classifying events by scene and object recognition. In: ICCV (2007)
24. Li-Jia Li, E.P.X., Su, H., Fei-Fei, L.: Object bank: a high-level image representation for scene classification and semantic feature sparsification. In: Proceedings of the Neural Information Processing Systems (NIPS) (2010)
25. Li-Jia Li, Y.L., Su, H., Fei-Fei, L.: Objects as attributes for scene classification. In: European Conference of Computer Vision (ECCV), International Workshop on Parts and Attributes, Crete, Greece, September 2010
26. Mesnil, G., Dauphin, Y., Glorot, X., Rifai, S., Bengio, Y., Goodfellow, I., Lavoie, E., Muller, X., Desjardins, G., Warde-Farley, D., Vincent, P., Courville, A., Bergstra, J.: Unsupervised and transfer learning challenge: a deep learning approach. In: Guyon I., Dror, G., Lemaire, V., Taylor, G., Silver, D. (Eds.) JMLR W& CP: Proceedings of the Unsupervised and Transfer Learning challenge and workshop, vol. 27, pp. 97–110 (2012)
27. Miller, G.A.: WordNet: a lexical database for english. Commun. ACM **38**(11), 39–41 (1995)
28. Oliva, A., Torralba, A.: Building the gist of a scene: the role of global image features in recognition. In: Visual Perception, Progress in Brain Research, vol. 155 (2006)
29. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: ICCV (2011)
30. Pearson, K.: On lines and planes of closest fit to systems of points in space. Phil. Mag. **2**(6), 559–572 (1901)
31. Quattoni, A., Torralba, A., Recognizing indoor scenes. In: CVPR (2009)
32. Ranzato, M., Poultney, C., Chopra, S., LeCun, Y.: Efficient learning of sparse representations with an energy-based model. In: NIPS'06 (2007)
33. Rifai, S., Mesnil, G., Vincent, P., Muller, X., Bengio, Y., Dauphin, Y., Glorot, X.: Higher order contractive auto-encoder. In: European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD) (2011)
34. Rifai, S., Vincent, P., Muller, X., Glorot, X., Bengio, Y.: Contracting auto-encoders: explicit invariance during feature extraction. In: Proceedings of the Twenty-eight International Conference on Machine Learning (ICML'11), June 2011
35. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: Labelme: a database and web-based tool for image annotation. Int. J. Comput. Vision **77**, 157–173 (2008)
36. Serre, T., Wolf, L., Poggio, T.: Object recognition with features inspired by visual cortex. In: IEEE Conference on Computer Vision and Pattern Recognition (2005)
37. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Trans. Pattern Anal. Mach. Intell. **22**, 1349–1380 (2000)
38. Torralba, A.: Contextual priming for object detection. Int. J. Comput. Vis. **53**(2), 169–191 (2003)

39. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.-A.: Extracting and composing robust features with denoising autoencoders. In: Cohen W.W., McCallum A., Roweis, S.T. (eds.) ICML'08, pp. 1096–1103. ACM (2008)
40. Vogel, J., Schiele, B.: Natural scene retrieval based on a semantic modeling step. In: Proceeedings of the International Conference on Image and Video Retrieval CIVR 2004, Dublin, Ireland, LNCS, vol. 3115, pp. 7 (2004)
41. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: SUN database: large-scale scene recognition from abbey to zoo. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3485–3492. IEEE, June 2010

Supervised Learning of Anatomical Structures Using Demographic and Anthropometric Information

Yoshito Otake, Catherine M. Carneal, Blake C. Lucas, Gaurav Thawait, John A. Carrino, Brian D. Corner, Marina G. Carboni, Barry S. DeCristofano, Michael A. Maffeo, Andrew C. Merkle and Mehran Armand

Abstract A supervised learning approach to predict anatomical structures derived from computed tomography (CT) images using demographic and anthropometric information is proposed. The approach applies a dimensionality reduction technique to a training dataset to learn a low-dimensional manifold representing variation of organ geometry or variation of the CT intensities itself, which computes a mapping between a low-dimensional feature vector and the organ geometry or

Y. Otake (✉) · B.C Lucas
Department of Computer Science, Johns Hopkins University, Baltimore, MD, USA
e-mail: otake@jhu.edu

B.C Lucas
e-mail: blucas5@jhu.edu

Y. Otake · M. Armand
Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD, USA
e-mail: Mehran.Armand@jhuapl.edu

C.M. Carneal · A.C. Merkle · M. Armand
Applied Physics Laboratory, Johns Hopkins University, Laurel, MD, USA
e-mail: Katy.Carneal@jhuapl.edu

A.C. Merkle
e-mail: Andrew.Merkle@jhuapl.edu

G. Thawait · J.A. Carrino
Department of Radiology, Johns Hopkins Hospital, Baltimore, MD, USA
e-mail: gthawai1@jhmi.edu

J.A. Carrino
e-mail: jcarrin2@jhmi.edu

B.D. Corner · M.G. Carboni · B.S. DeCristofano · M.A. Maffeo
US Army Natick Soldier Research Development and Engineering Center, Natick, MA, USA
e-mail: brian.d.corner@mail.mil

M.G. Carboni
e-mail: marina.g.carboni@mail.mil

B.S. DeCristofano
e-mail: barry.s.decristofano@mail.mil

M.A. Maffeo
e-mail: michael.a.maffeo.civ@mail.mil

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_14

CT volume. Regression analysis is then applied to determine a regression function between the low-dimensional feature coordinates and external measurements of the subjects such as demographic or anthropometric data. Then for an unseen subject, the low-dimensional feature coordinates are predicted from external measurements using the computed regression function. Subsequently, the organ geometry or the CT volume is estimated from the mapping computed in the dimensionality reduction. As an example case, lung shapes and thoracic CT scans were analyzed based on available demographic parameters (age, gender, race) and anthropometric measurements (height, weight, and chest dimensions). The training dataset consisted of lung shapes represented as a topologically consistent point distribution model (PDM) and CT volumes (256^3 voxels, 1.5^3 mm/voxel) of 124 subjects. The prediction error of lung shape of an unknown subject based on 11 independent demographic and anthropometric variables was 10.71 ± 5.48 mm. Isomap analysis of CT volumes revealed that 95 % of the total variance was explained with 4 dimensions, and the computed mapping clearly captured trends in anatomical variation. This suggested a potential for a direct CT-volume based statistical analysis using dimensionality reduction, which we call a voxel-based statistical atlas. Potential application areas of the proposed approach includes subject-specific ergonomic design in personal protective equipment or population-specific finite-element modeling in biomechanical analysis. Examples also include the use of a predicted patient-specific CT volume as it a prior information for image quality improvement in low dose CT, and optimization of CT scanning protocols.

Keywords Supervised learning · Dimensionality reduction · Organ geometry · Demographic and anthropometric data · Regression analysis · Statistical shape atlas · Allometry.

1 Introduction

Machine learning approaches in the analysis of organ geometries using statistical shape atlases are a prevalent trend in various target application fields, such as cardiac modeling [1], pelvis shape analysis for dose reduction in computed tomography (CT) [2], 4-dimensional lung motion modeling [3], and a small animal research using Micro-CT [4].

Most existing statistical shape atlases of human organs are created from a training dataset composed of an anonymized CT dataset, thus the analyses were mostly confined to organ shape among a select disease group or subject population. To our knowledge, the relationship between anthropometric and demographic data with a statistical atlas of generalized population has not yet been investigated.

In order to address this gap, we propose a supervised learning approach to analyze correlation between a subject's external characteristics and their internal organ geometry derived from CT data. As an initial feasibility study, we collected a thoracic CT datasaet together with externally-available patient features, including demographic

information and several anthropomorphic metrics. The lungs were segmented from thoracic CT data as a target organ and considered its geometric features as a cloud of connected points (Point Distribution Model, PDM).

Combined dimensionality reduction and regression analysis were used to demonstrate the ability to predict the lung's complex anatomy solely from external-derived subject characteristics.

Demographic or stature-based prediction of information about internal organ structures, which are typically measured via expensive medical imaging or other invasive methods, may be useful in a variety of application scenarios ranging from medical device development to personalized medicine and protection. Also, prediction of a patient-specific CT volume may open new areas of research in CT reconstruction and image-guided surgery. Prior information such as a previously scanned CT or CAD models of an implant in the reconstruction FOV have been used in CT reconstruction to improve reconstruction quality [5–7]. The patient-specific CT volume predicted from non-invasively measured external information can be used as a prior for those reconstruction methods, which would significantly improve image quality in low dose CT. A roughly estimated CT volume can also be used in optimization of patient-specific CT scanning protocol. Another potential application area is a priori information for surgical guidance. Preoperative CT volumes are frequently used for surgical guidance through rigid and/or deformable registration with intraoperative images (e.g., X-ray projection [8], ultrasound [9], etc.). However, the necessity of preoperative CT scanning, which impart ionizing radiation to the patient, is a limiting factor for surgical guidance system. The trade-off between the benefit of improved surgical quantitative data and accuracy versus ionizing radiation limits the application of surgical guidance system to more challenging surgical treatments. The patient-specific CT volumes predicted from non-invasive measurements would broaden application of such surgical guidance to more common surgeries which does not include preoperative CT in its routine protocol.

2 Methods

2.1 Materials

We used existing radiological CT scans of the chest region from 124 patients. Following Johns Hopkins Institutional Review Board (IRB) approval, we searched the radiology archives at Johns Hopkins Hospital for thoracic or chest CT scans of males and females ages 17–45. Only very strictly normal scans of lung were included in this study (normal by report and inspection). Any scan with obvious or minimal pathology was excluded. Scans that showed lungs without disease but with findings different from normal (such as atelectasis, normal variants) were also excluded. Subject characteristics of age, gender, ethnicity, height and weight were extracted from their medical records archives. In order to reduce population bias in the sta-

Table 1 Number of subjects in each population group in the training dataset

	White (n)	Black (n)	Hispanic (n)	Other (n)	Total (%)
Male (n)	19	15	14	12	48.4
Female (n)	19	17	14	14	51.6
Total (%)	31.6	25.8	22.6	21.0	100

tistical atlas, selection involved patients with relatively even distribution of gender and ethnicity. For the purpose of this initial study, ethnic groups' bins include White, Black, Hispanic, and Other. Subjects were anonymized after extraction. Table 1 and Fig. 1 show the distribution of demographic property of the population in the training dataset.

External measurements of each subject's chest span, chest depth, chest breadth, and inter-nipple distance were manually approximated from landmarks on the CT images. These measurements were selected to correspond with those used in common anthropometric surveys [10]. Chest span (cranio-caudal) was defined as the vertical distance between highest level of first rib to the lower costophrenic angle. Chest breadth (or width) was defined as the skin to skin depth of the chest at the carinal plane at the level of nipples. Finally, the inter-nipple distance measurement was made in an axial plane view where both nipples were visible.

2.2 Construction of Training Datasets

We selected a template CT image from the acquired dataset. We manually segmented this template and used it to generate a template tetrahedral mesh volume consisting of 112,602 vertices and 509,034 tetrahedrons.

We developed a software pipeline for creating statistical atlas as follows. An intensity-based deformable registration method (Mjolnir [11]) was applied to deformably register the CT data of each subject to the template CT. The resulting deformation field was applied to the template mesh to create a tetrahedral mesh

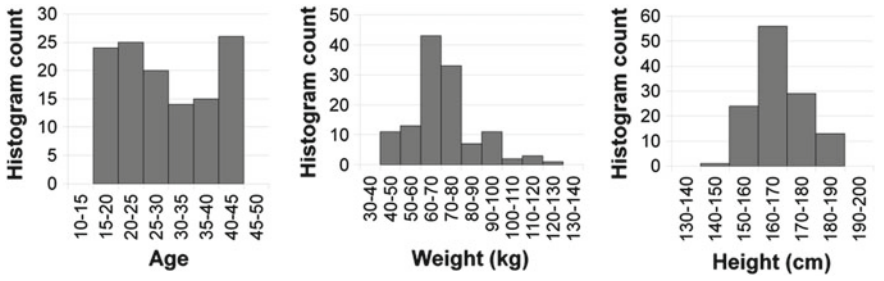


Fig. 1 Demographic property distributions of the population in training dataset

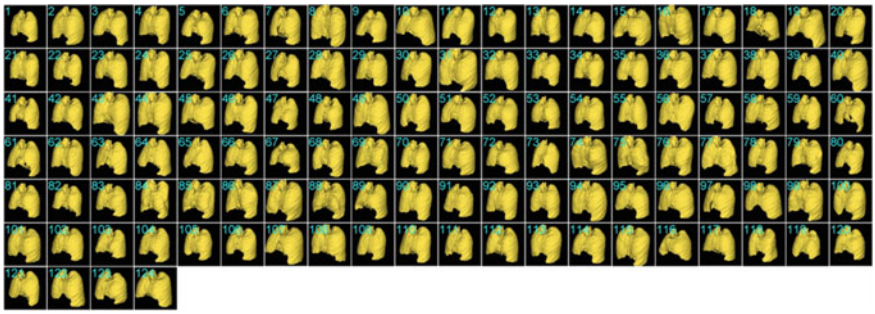


Fig. 2 Training dataset. Point Distribution Model (PDM) of lung geometry of 124 subjects were created from CT dataset. Automatic segmentation combined with a deformable registration algorithm (Mjolnir [11]) using one subject (subject #44) as a template was employed to obtain topology-consistent meshes. Thus point correspondence was inherently solved. The subjects were chosen in such a way that the demographic characteristics were well balanced

representing each individual subject. Thus, the geometric point correspondence, which is one of the key considerations in a typical statistical atlas construction process, was inherently solved in our pipeline.

Figure 2 shows the entire training dataset that we used in this study, while Fig. 3 shows the mean shape and standard deviation among each population group classified based on race and gender.

The mean shape for white males was the largest, with its surface extending over 10 mm beyond the grand mean. The other female demographic had the smallest lung shape, with average lung surfaces about 10 mm smaller than the grand mean. The average shapes for white females and black males were overall similar to the grand mean.

2.3 Proposed Approach

The workflow of the proposed method to predict internal lung anatomy from subject demographic and anthropometric data is a 2-stage process, consisting of a learning step and a prediction step (Fig. 4).

In the learning step, we modelled each shape instance as a vector of X, Y, Z coordinates of all the mesh vertices and applied dimensionality reduction algorithm creating low dimensional feature coordinates for each subject. Two types of dimensionality reduction algorithms were tested, including principal component analysis (PCA) and Isomap [12]. Linear least square regression analysis was performed to compute a regression function between the feature coordinates (also called mode weights in PCA) and the external measurements of the subject.

The prediction step predicted the feature coordinates of an unknown subject from its external measurements, and subsequently the organ shape was estimated from

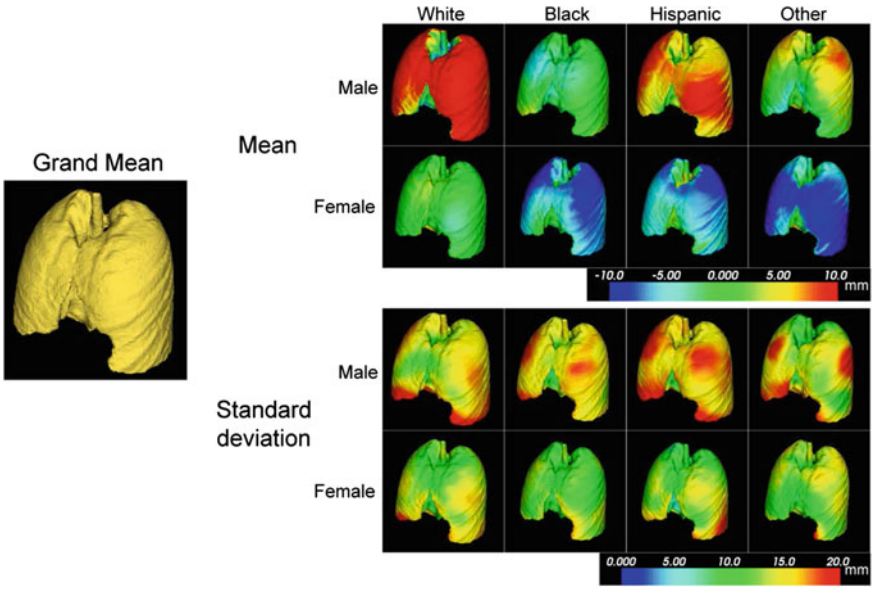


Fig. 3 Mean shape (*upper 2 rows*) and standard deviation (*lower 2 rows*) of the training dataset in each population group. Colormap of mean indicates the displacement of each vertex from the grand mean (mean of the entire population) along the normal direction of a triangle mesh at each vertex, positive indicating outward direction

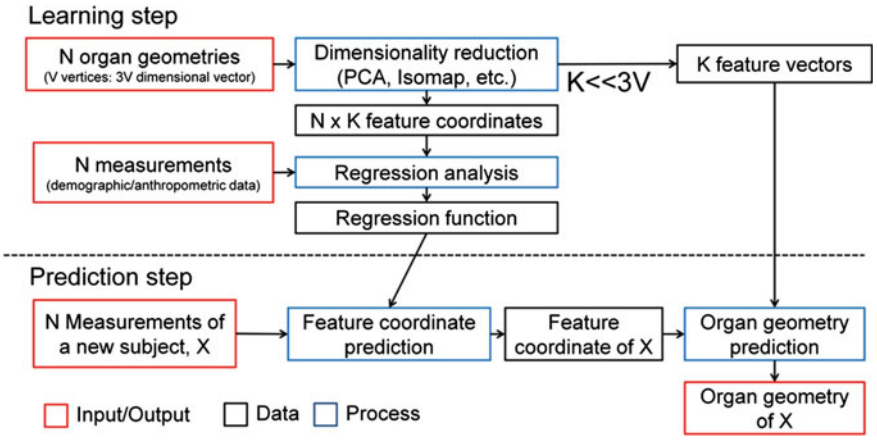


Fig. 4 Overview of the workflow of organ geometry prediction. Learning step reduces dimensionality of the organ geometry based on the training dataset and produces feature vectors and feature coordinates of each geometry. Then a regression function was created by a regression analysis on the demographic/anthropometric data and the feature coordinates. Prediction step determines shape of the target organ of a new subject based on feature coordinates predicted by the regression function

the predicted feature coordinates. As is the case with a typical regression analysis, the approach makes it possible to analyze magnitude and direction of correlation between each external measurement and the feature coordinates which encode the organ shape. The proposed approach makes the analysis of the complex variation of organ shapes represented as a large dimensional vector tractable by using dimensionality reduction. The regression step can employ more general classes of non-linear regression methods, although a simple linear least square approach was used in our feasibility study reported here.

The following subsections detail each step in the proposed workflow.

2.3.1 Dimensionality Reduction

For analysis of organ geometry, mesh vertices of each subject were represented as a $3V$-dimensional vector x_i, where V is the number of vertices and i is the index of the subject (V =112,602 in our analysis). For analysis of CT volumes, voxel intensities of each subject were represented as a V-dimensional vector x_i, where V is the number of voxels ($V = 256^3 = 16,777,216$ in our analysis). We performed dimensionality reduction on both analyses in the same manner.

Principal Component Analysis

PCA was performed on the lung dataset $\{x_i, i = 1, \ldots, N\}$ (N: number of subjects) creating a new feature coordinate system that represents each geometry

$$x_i = \bar{x} + \sum_{j=1}^{M} a_j^i e_j \tag{1}$$

where e_j represents the feature vectors (principal mode vectors), which is eigenvectors of the covariance matrix of x_i sorted according to decreasing eigenvalues λ_j. is the mean shape and a_j^i are the feature coordinates (mode weights) that correspond to each feature vector. M is the number of feature vectors.

Given a set of feature coordinates for an unknown subject $\{a_j^u, j = 1, \ldots, M\}$, its organ geometry x_u is estimated (reconstructed) by (1) (Fig. 5).

Isomap

Isomap is a type of non-linear dimensionality reduction method modeling training datasets as a weighted graph based on its distance matrix. The distance matrix produces a new distance measure called geodesic distance, and classical eigen analysis, multidimensional scaling (MDS [13]) is then applied on the geodesic distances. The connectivity of each data point in the neighborhood graph is defined as its

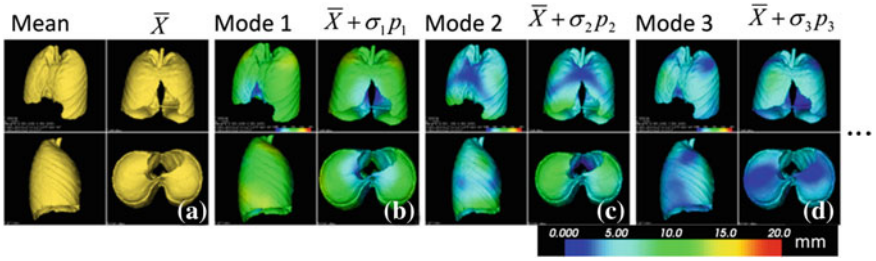


Fig. 5 Results of PCA analysis on point distribution model of lungs of the training dataset. **a** mean shape, mean plus one standard deviation in the direction of mode 1 (**b**), mode 2 (**c**), and mode 3 (**d**). The colormap (0–20 mm) shows displacement of each vertex from the mean shape

k-Nearest Neighbors (kNN) in the high-dimensional space. In this paper, a simple Euclidean distance was used as the distance metric in the high-dimensional space and K = 6 was chosen for kNN. Isomap computes a mapping between a high-dimensional vectors $\{x_i, i = 1, \ldots, N\}$ and a low-dimensional feature coordinates $\{a^i_j, j = 1, \ldots, M\}$for each subject i.

For reconstruction of the organ geometry of an unseen subject x_u from its feature coordinates, we used kNN interpolation in the feature space. kNN was computed based on Euclidean distance between the feature coordinates and an inverse distance was used as the weight [14] as follows.

$$x_u = \sum_{i \in kNN} x_i \frac{d_i^{-p}}{\sum_{j=1}^{k} d_j^{-p}}$$
$$d_j = \sqrt{\sum_{m=1}^{M} \left(a^u_m - a^j_m\right)^2} \tag{2}$$

where d_i is the Euclidean distance between two M dimensional vector a^u and a^j. We employed a simple interpolation scheme to reduce the computation time in our initial implementation, however, a more computationally intensive interpolation method such as radial basis function (RBF) [15] can also be applied in this step.

2.3.2 Linear Least Square Regression on Feature Coordinates and Measurements

We performed linear least square regression analysis on the low-dimensional feature coordinates (a^i_j) and external measurements $\{x_i, i = 1, \ldots, N\}$. Here X_i represents a K-vector consisting of K measurements of ith subject.

Using X_i as independent variables and a^i_j as a dependent variable, we computed regression coefficients A_m (intercept) and $B_{m,k}$ for each feature coordinate independently. Thus the computed regression function can be written as follows.

$$\begin{cases} a_1 = A_1 + \sum_{k=1}^{k} B_{1,k} X_k \ (\textit{feature coord.1}) \\ a_2 = A_2 + \sum_{k=1}^{k} B_{2,k} X_k \ (\textit{feature coord.2}) \\ \qquad\qquad \vdots \\ a_M = A_M + \sum_{k=1}^{k} B_{M,k} X_k \ (\textit{feature coord.M}) \end{cases} \tag{3}$$

2.3.3 Prediction of Organ Geometry from External Measurements

To compute the organ geometry of an unseen subject x_u from a set of external measurement of the subject X_u, we first computed a set of organ's feature coordinates $\{a_j^u, j = 1, \ldots, M\}$ using the regression function (3). Then the organ geometry was computed based on the feature coordinates as described in sections "Principal Component Analysis and Isomap respectively".

2.4 Validation Method

In order to evaluate accuracy of the proposed method, leave-out validation tests on the organ geometries were conducted. In the first set of tests, 2 subjects (#33 and #47) were left out. The proposed learning step was performed to the training dataset excluding the 2 subjects. Organ geometries of the 2 left-out subjects were predicted using the proposed approach and compared with each subject's true geometry. Distance between the vertices of the predicted and the true shape were computed as an error metric and colormapped on the predicted shape.

The second validation tests were a series of leave-one-out tests. Each subject was left-out one at a time, and the same test described above was repeated.

3 Results and Discussion

3.1 Comparison Between Two Dimensionality Reduction Algorithms

Figure 6 shows an example of sorting of the training datasets based on the first 2 principal modes (feature vectors) using the 2 different dimensionality reduction methods. Subjects in the training datasets are sorted according to their feature coordinates and plotted in Fig. 6a, b. Figure 6c, d demonstrate lung shapes each corresponding to those plots above.

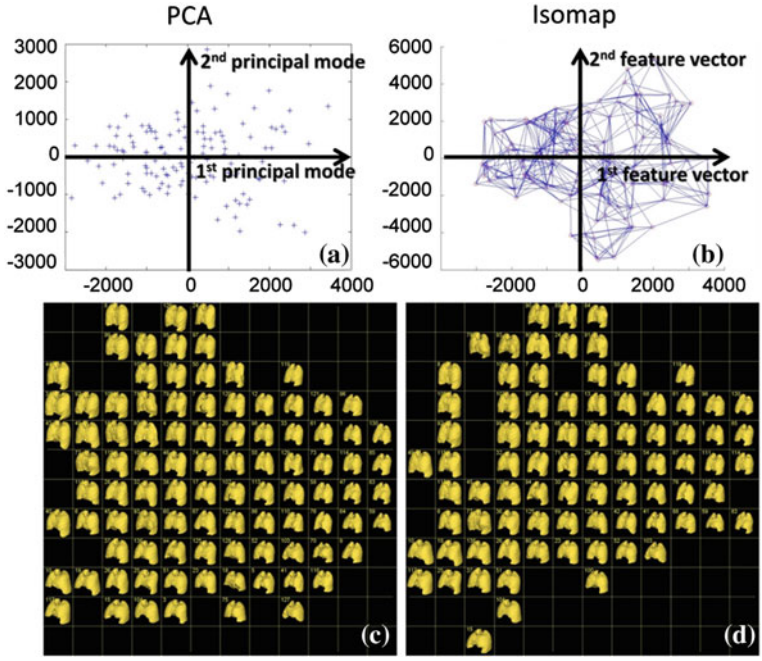


Fig. 6 A comparison between 2 different dimensionality reduction algorithms: PCA and Isomap. The training datasets were sorted based on the first 2 principal modes (*feature vectors*). **a**, **b** plots showing distribution of the training dataset, **c**, **d** lung shape of each dataset that corresponds to the points in (**a**, **b**). The results produced by PCA and Isomap was similar

Figure 7 shows an example of interpolation between 2 subjects in the feature space using the 2 different methods. The interpolations were performed in its feature coordinates and 4 sequentially interpolated shapes (20, 40, 60, 80 % between the two shapes) were shown.

As previous work noted [16], PCA and Isomap produced similar results, which suggested that the modeling (parameterization) based on PDM does not produce a highly nonlinear manifold. However, as shown in [12], a different type of input dataset, such as face images, creates nonlinear manifold which can only be captured by nonlinear dimensionality reduction methods. We explore the potential of nonlinear algorithm in the following section in the analysis of CT volumes.

3.2 Dimensionality Reduction on CT Datasets

Isomap analysis was performed on the original CT volumes of 124 training subjects. Figure 8 illustrates results of the analysis. 124 CT volumes were sorted (Fig. 8c–e) based on the first two feature coordinate computed by Isomap (Fig. 8a). Surprisingly,

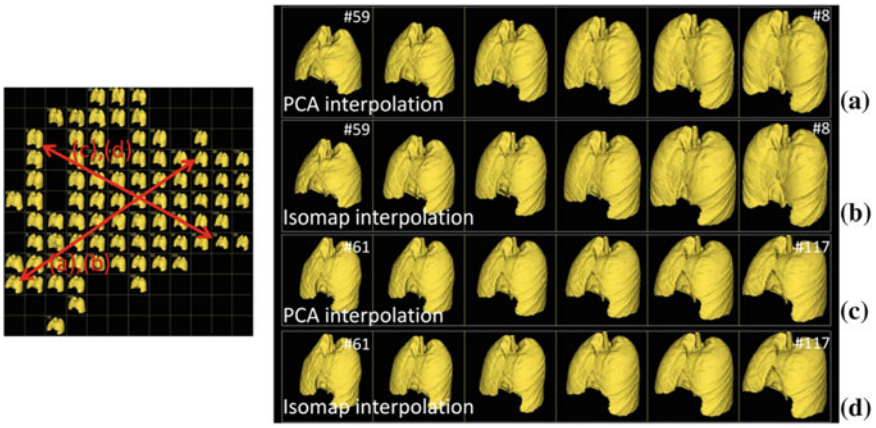


Fig. 7 A comparison of interpolation using 2 methods. Interpolation from subject #59 to #8 using **a** Isomap and **b** PCA, from subject #61 to #117 using **c** Isomap and **d** PCA. Interpolated geometries were computed by k-NN interpolation (K = 6) of feature coordinates (mode weights) of the two subjects. Similar to the sorting result (Fig. 6), PCA and Isomap produced similar results

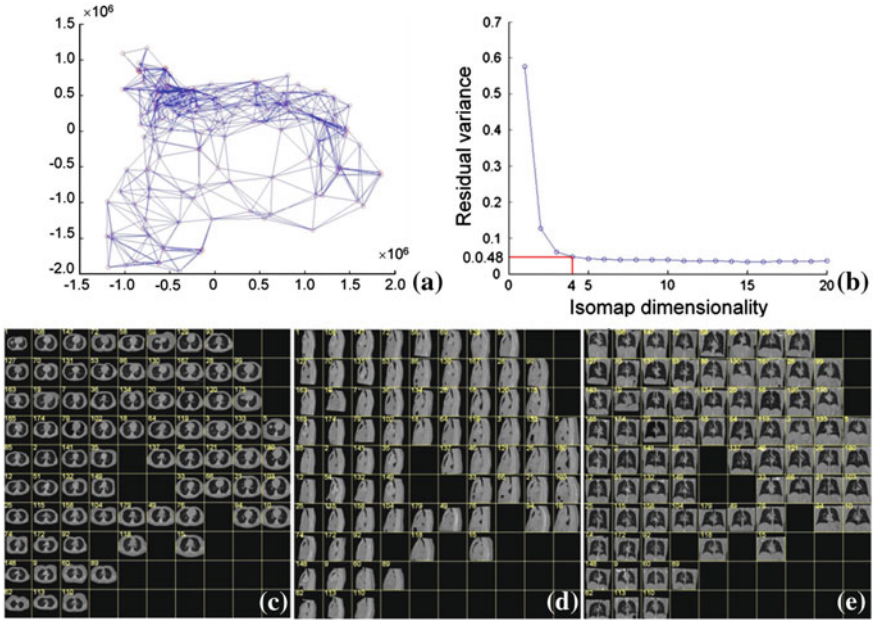


Fig. 8 Isomap analysis of CT volumes of 124 training subjects. **a** two dimensional manifold. **b** residual variance as a function of dimensionality. More than 95 % of variation was explained by only 4 dimensions. **c–e** axial, sagittal and coronal slices of the CT volumes. Isomap captured 2 major trends in the anatomical variation, thick–thin (*horizontal axis*) and tall–short (*vertical axis*) directly from the CT volumes

	Intercept	Age	Race[B]	Race[H]	Race[O]	Gender [F]	Weight	Height	IN Distance	Chest Breadth	Chest Depth	Chest Span	R Square
Mode 1	-81.547	0.037	-3.278	-3.186	-2.609	-2.550	0.036	8.356	0.143	0.671	0.545	1.280	0.787
Mode 2	-5.305	-0.012	-0.098	-1.682	-1.137	-1.156	0.002	7.108	0.274	0.263	0.338	-1.024	0.368
Mode 3	14.117	-0.031	-0.821	1.242	-1.463	-0.058	0.044	-13.172	0.145	0.117	0.098	-0.099	0.226
Mode 4	1.628	0.026	-0.033	0.951	0.591	-1.413	-0.006	6.832	0.024	0.238	-0.343	-0.516	0.203
Mode 5	1.023	0.043	-1.396	-0.911	-0.926	0.418	-0.040	-0.622	-0.003	-0.323	0.598	-0.055	0.282
Mode 6	0.787	0.052	0.491	0.826	-1.023	-0.647	-0.013	1.339	-0.084	0.089	-0.136	-0.051	0.132
Mode 7	2.630	0.036	-0.729	-1.541	-1.386	-0.037	-0.034	1.375	-0.063	0.117	-0.121	-0.088	0.137
Mode 8	0.323	0.041	-0.670	0.050	-0.758	-0.164	-0.019	2.522	-0.136	-0.027	-0.010	-0.009	0.113
Mode 9	3.296	0.007	-0.111	0.400	0.399	-0.386	0.003	-1.592	0.025	0.013	-0.048	-0.035	0.029
Mode 10	-4.787	-0.039	0.145	0.203	0.604	0.366	-0.025	-0.542	-0.064	0.071	0.234	0.093	0.116

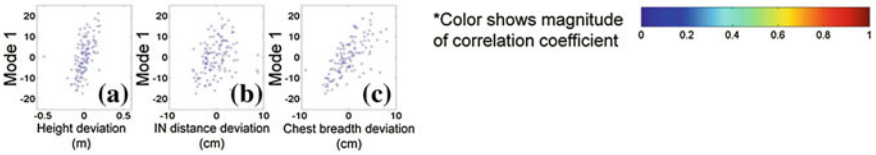


Fig. 9 Results of linear least square analysis on the first 10 mode weights and 11 demographic/anthropometric measurements. The table shows coefficient of the computed regression functions. Magnitude of correlation coefficient was colormapped in the table. Strong correlation between mode 1 and a few anthropometric data is observed. Chest depth and chest span showed higher correlation with mode 4 and 5. Distribution of the representative measurements (height, IN distance, chest breadth) versus 1st mode weight were shown below *left* (**a–c**)

it turned out that 95 % of the total variation was explained by the first four dimensions and the first two dimensions clearly captured two major trends in the anatomical variation (see Fig. 8c–e, where subjects were sorted thinner to thicker from left to right and shorter to taller from bottom to top). This suggested a strong potential for use of Isomap in supervised learning of CT volumes.

3.3 Regression on Mode Weights and Measurements

Results of the linear least square analysis on the first 10 mode weights are shown in Fig. 9. There is strong correlation between mode 1 and several anthropometric data. As indicated from the table color bar, the subject data with strongest correlation included chest breadth and chest span. Chest breadth, gender, and overall height also showed strong correlation. The three plots (Fig. 9a–c) demonstrate the correlation between mode 1 and the individual variables of height (a), inter-nipple distance (b), and chest breadth (c).

3.4 Leave-Out Validation Test

We left out two subjects (#33 and #47), and performed PCA and regression analysis using the other 122 subjects. The accuracy of the prediction of each mode weight was validated using the left-out 2 subjects. Table 2 shows the error in prediction of

Table 2 Results of mode weight prediction in leave-two-out validation test

	True mode weights (mm)		Predicted mode weights (mm)		Prediction error (mm)	
Subject ID	#33	#47	#33	#47	#33	#47
Mode 1	−8.54	−15.5	−6.68	−11.36	1.85	4.14
Mode 2	−4.10	1.78	−3.80	1.16	0.30	−0.62
Mode 3	−1.56	−0.16	−0.11	−1.50	1.46	−1.34
Mode 4	1.54	−2.10	−0.81	0.51	−2.35	2.61
Mode 5	−1.40	1.83	0.84	3.08	2.24	1.24
Mode 6	−2.28	−0.09	0.75	−0.95	3.03	−0.87
Mode 7	2.34	−2.03	1.10	0.14	−1.25	2.17
Mode 8	0.56	−0.69	0.78	0.07	0.22	0.76
Mode 9	−1.23	−1.39	0.18	0.40	1.41	1.79
Mode 10	−0.61	−0.08	−0.38	−0.53	0.23	−0.45

the mode weights. Despite the large inter-subject variation in the true mode weights (column 2 and 3), the proposed method predicted the mode weight with about 2 mm error on average.

Figure 10 shows the result of prediction of the lung geometry. We compared our prediction result (middle column) to the grand mean shape (right column), since the mean shape is the best prediction when no additional information (external

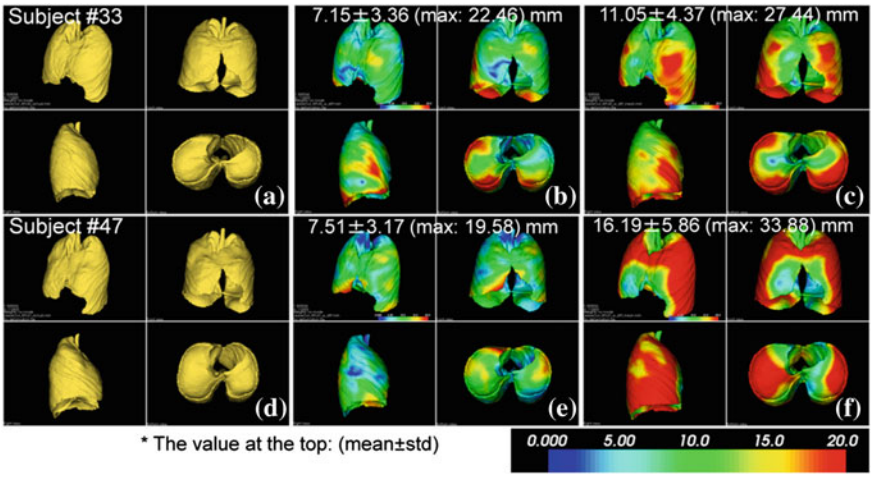


Fig. 10 Results of the left-two validation test. **a–c** prediction of subject #33 (43 y.o., female, 48.53 kg, 152 cm). **a** true shape, **b** predicted shape, **c** mean shape. **d–f** true, predicted and mean shape of subject #47 (44 y.o, female, 49.9 kg, 160 cm). The color map shows the error at each vertex from the true shape. The predicted geometries were produced based on the predicted mode weights using kNN interpolation (K = 6) in the feature space

measurements) was involved. Compared to the error distribution in mean shape, our prediction clearly showed improved results in both subjects, especially around the inferior regions of the lung lobes.

Results of the repeated leave-one-out validation tests are shown in Fig. 11. The distance error was about 10 mm on average. A few outliers that showed much larger

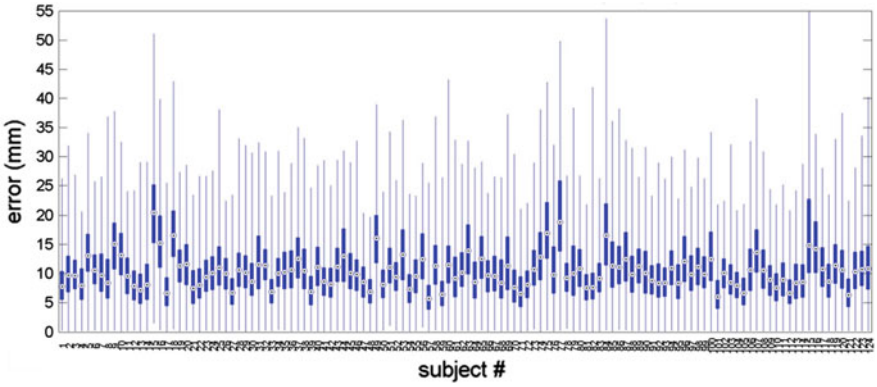


Fig. 11 Results of leave-one-out validation test. All 124 subjects were left out and validated one at a time. Each plot shows the displacement error (mm) at 112,602 vertices of the lung (*box plot* 25–75 %, *whisker plot* maximum and minimum, *dot* median). Mean and standard deviation of the error over the entire subjects were 10.71 ± 5.48 mm

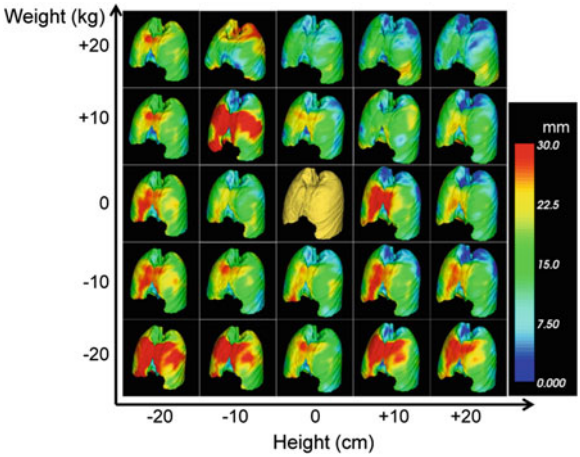


Fig. 12 An example of lung shape prediction based on 2 demographic parameters (*height* and *weight*) using Isomap. Feature vectors were extracted from 124 training datasets using Isomap. Regression function was computed to predict feature coordinate of a new instance based on the 2 demographic parameters. The figures show lung shapes when height and weight were varied $[-20 + 20]$ cm, $[-20 + 20]$ kg respectively from a typical subject (#36, 68.03 kg, 173 cm). Color map indicates distance at each vertex from the subject's lung (shown at the center)

error such as #15 or #115 were attributed as the error in the learning step due to either segmentation or registration error.

Figure 12 shows a simple application example of the proposed workflow where the lung shape was predicted based on 2 simple external measurements, height and weight.

4 Conclusions

We proposed a supervised learning framework using a statistical shape atlas of human internal organs to predict an individual's lung anatomy from their external characteristics (demographic and anthropometric information). By incorporating dimensionality reduction methods, the proposed approach can perform regression analysis with a reasonably small number of variables, making the analysis of correlation between complex shape variation and demographic information tractable.

We applied the proposed method on an initial dataset of 124 subjects and demonstrated prediction of the lung geometry within 10 mm average error. Improvement of the predictive models would likely be achieved by expanding the training dataset. Future work includes sample size analysis to determine the sufficient number of samples for a particular application. Additionally, although only four external anthropometric features were selected in this paper, improvement of the predictive models may be increased by increasing the number of external features employed.

The proposed supervised learning based workflow consisting of dimensionality reduction and regression analysis is more broadly applicable to various cases such as ergonomic design in industry, population-specific finite element modeling, prior information in low dose CT scanning, and patient-specific optimization of CT scanning protocol.

Acknowledgments This research was supported in part by the United States Army Natick Soldier Research Development and Engineering Center. The opinions expressed are those of the authors alone and do not reflect the views of the U.S. Army. Approved for unlimited public release, US Army Natick Soldier RDEC, PAO #U12-424.

References

1. Frangi, A.F., et al.: Automatic construction of multiple-object three-dimensional statistical shape models: application to cardiac modeling. IEEE Trans. Med. Imaging **21**(9), 1151–1166 (2002)
2. Chintalapani, G., et al.: Statistical atlas based extrapolation of CT data. Medical Imaging, Visualization. Image-Guided Procedures, and Modeling **2010**, 7625 (2010)
3. Ehrhardt, J., et al.: Statistical modeling of 4D respiratory lung motion using diffeomorphic image registration. IEEE Trans. Med. Imaging **30**(2), 251–265 (2011)

4. Hongkai, W., Stout, D.B., Chatziioannou, A.F.: Estimation of mouse organ locations through registration of a statistical mouse atlas with micro-CT images. IEEE Trans. Med. Imaging **31**(1), 88–102 (2012)
5. Stayman, J.W., et al.: Model-based tomographic reconstruction of objects containing known components. IEEE Trans. Med. Imaging **31**(10), 1837–1848 (2012)
6. Otake, Y., et al.: Model-based cone-beam CT reconstruction for image-guided minimally invasive treatment of hip osteolysis, pp. 86710Y–86710Y (2013)
7. Lauzier, P.T., Chen, G.H.: Characterization of statistical prior image constrained compressed sensing (PICCS): II. Application to dose reduction. Med. Phys. **40**(2), 021902 (2013)
8. Otake, Y., et al.: Intraoperative image-based multiview 2D/3D registration for image-guided orthopaedic surgery: incorporation of fiducial-based C-arm tracking and GPU-acceleration. IEEE Trans. Med. Imaging **31**(4), 948–962 (2012)
9. Nam, W.H., et al.: Automatic registration between 3D intra-operative ultrasound and pre-operative CT images of the liver based on robust edge matching. Phys. Med. Biol. **57**(1), 69–91 (2012)
10. Gordon, C.C., et al.: Anthropometric survey of US army personnel: methods and summary statistics, DTIC Document (1988/1989)
11. Ellingsen, L.M., et al.: Robust deformable image registration using prior shape information for atlas to patient registration. Comput. Med. Imaging Graph **34**(1), 79–90 (2010)
12. Tenenbaum, J.B., Silva, Vd, Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science **290**(5500), 2319–2323 (2000)
13. Borg, I., Groenen, P.J.: Modern Multidimensional Scaling: Theory and applications. Springer, New York (2005)
14. Shepard, D.: A two-dimensional interpolation function for irregularly-spaced data. In: Proceedings of the 1968 23rd ACM National Conference1968, ACM pp. 517–524
15. Press, W.H., et al.: Numerical Recipes in C+: The Art of Scientific Computing. Cambridge University Press, Cambridge (2009)
16. Seshamani, S., Chintalapani, G., Taylor, R.: Iterative refinement of point correspondences for 3D statistical shape models. Med. Image. Comput. Comput. Assist. Interv. **14**(Pt 2), 417–425 (2011)

Wikifying Novel Words to Mixtures of Wikipedia Senses by Structured Sparse Coding

Balázs Pintér, Gyula Vörös, Zoltán Szabó and András Lőrincz

Abstract We extend the scope of Wikification to novel words by relaxing two premises of Wikification: (i) we wikify without using the surface form of the word (ii) to a mixture of Wikipedia senses instead of a single sense. We identify two types of "novel" words: words where the connection between their surface form and their meaning is broken (e.g., a misspelled word), and words where there is no meaning to connect to—the meaning itself is also novel. We propose a method capable of wikifying both types of novel words while also dealing with the inherently large-scale disambiguation problem. We show that the method can disambiguate between up to 1,000 Wikipedia senses, and it can explain words with novel meaning as a mixture of other, possibly related senses. This mixture representation compares favorably to the widely used bag of words representation.

Keywords Interpreting novel words · Wikification · Link disambiguation · Natural language processing · Structured sparse coding

---

The work was carried out while Zoltán Szabó was working at Eötvös Loránd University, Hungary

B. Pintér · Gy. Vörös · Z. Szabó · A. Lőrincz (✉)
Faculty of Informatics, Eötvös Loránd University, Pázmány P. sétány 1/C, Budapest 1117, Hungary
e-mail: bli@elte.hu

Gy. Vörös
e-mail: vorosgy@inf.elte.hu

A. Lőrincz
e-mail: lorincz@inf.elte.hu

Z. Szabó
Gatsby Computational Neuroscience Unit, University College London, Alexandra House, 17 Queen Square, London WC1N 3AR, UK
e-mail: zoltan.szabo@gatsby.ucl.ac.uk

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*, Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_15

1 Introduction

Wikification aims to help users and computers alike in understanding texts by enriching them with encyclopedic knowledge in the form of links to Wikipedia articles [1]. However, Wikification concerns itself only with correct and known words: neologisms, misspelled words and the like fall outside its scope.

These *novel words* are different in that the connection between their surface form[1] and their meaning is broken (e.g., a misspelled word), or—in the more involved case—there is no meaning to connect to (e.g., a word with a completely new meaning). This property makes them particularly hard to interpret, but it also makes them the words that need interpreting the most.

This paper extends the scope of Wikification to *novel words* by interpreting them (i) *without relying on their surface form* and (ii) as a *weighted mixture of Wikipedia senses*, instead of as a single sense.

Usually, Wikification consists of two phases: *link detection* and *link disambiguation*. The detection phase identifies the terms and phrases from which links should be made. The disambiguation phase identifies the appropriate Wikipedia article for each detected term to link to. For example, the term *bank* could link to an article about financial institutions or river banks. We consider only disambiguation, as the words to be disambiguated are assumed given: they are the novel words in the text.

Similarly to Mihalcea [1], we regard Wikipedia as a *sense inventory*, where each link can be thought of as a sense-annotated word. In each link, the anchor text of the link—the word—is annotated with the target Wikipedia page—the sense.

Novel words can be of two types with respect to this sense inventory. In the first case, a novel surface form is—maybe incorrectly—associated with an already known meaning. An example for correct word use is a neologism where a new word gets associated with an already known sense (e.g., neologisms created by clipping: professor → prof, facsimile → fax). Examples for words used incorrectly include misspelled words, mixed up words like homophones, scanning or Optical Character Recognition errors, errors introduced by automatic speech recognition, etc. For the sake of simplicity, we also refer to these as *novel words*, although they may be completely unintelligible (e.g., a word completely blurred in a scanned document).

In the second case, the meaning of the novel word itself is also novel—it is not present in the sense inventory. In many cases, these words can be explained by a mixture of senses. A striking example is neologisms created by blending, like edutainment (from education and entertainment) and netiquette (from network and etiquette) [2]. Even in less clear-cut cases, finding a set of senses closely related to the novel meaning could help users and computer algorithms alike to understand it.

To interpret these novel words, we have to overcome a new difficulty. As we do not rely on the surface form of the target word,[2] the *complexity of the disambiguation problem increases*. Current methods for Wikification treat the disambiguation of

[1] The form of a word as it appears in the text.

[2] The word to be explained with Wikipedia senses.

different word types[3] independently. In the case of novel words, we cannot formulate an independent problem for each surface form; we have to disambiguate among hundreds or thousands of senses at once instead of about a dozen. This vast number of candidate senses results in a large-scale problem, and this is why the new difficulty appears.

Typical methods to disambiguate words with *correct* surface form apply the *distributional hypothesis*. According to the distributional hypothesis, words that occur in the same contexts tend to have similar meanings [3]. Because our new disambiguation problem without using the surface form is large-scale, exceptions to the distributional hypothesis occur more frequently. Particularly, let us call two contexts *spuriously similar* if they are similar but belong to words that denote different senses. The number of spuriously similar contexts tends to increase inherently with the number of candidate senses. There is more chance to select a wrong sense from among 1,000 senses than from among 10: the learning problem becomes considerably harder.

To counter the effect of spurious similarities, we use the *distributional hypothesis* in a novel way. We introduce structured sparse coding [4] to diminish the effect of spurious similarities of contexts by matching the structure in the regularization to the structure of the problem (Sect. 3).

The **contributions** of the paper are summarized as follows: (i) we propose a method to interpret novel words as weighted mixtures of Wikipedia senses. (ii) We show that structured sparsity reduces the effect of spurious similarities of contexts. (iii) We perform large-scale evaluations where we disambiguate among 1,000 Wikipedia senses at once.

In the next section we review related work. Our method and results are described in Sects. 3 and 4. We discuss our results in Sect. 5 and conclude in Sect. 6.

2 Related Work

The main differences between previous methods for Wikification and ours is that they consider the disambiguation problems of different word types independently, and they wikify to a single Wikipedia sense. We relaxed these two premises to make interpreting novel words possible.

Mihalcea et al. [1] introduced the concept of Wikification: they proposed a method to automatically enrich text with links to Wikipedia articles. They used keyword extraction to detect the most important terms in the text, and disambiguated them to Wikipedia articles with supervised learning using the contexts. The same task was solved in [5] more efficiently. Here, contexts were taken into account also for the detection phase. Disambiguation was done using sense *commonness* and sense *relatedness* scores.

[3] In "A rose is a rose is a rose", there are three word types (a, rose, is), but eight word tokens.

Unlike the previously mentioned works, which introduce links to important terms in the text chiefly to achieve better readability, the goal of [6] was to add as many links as possible to help information retrieval. The terms were disambiguated by assuming that coherent documents refer to entities from one or a few related topics or domains. Ratinov et al. [7] proposed a similar disambiguation system called GLOW (Global Wikification), which used several local and global features to obtain a set of disambiguations that are coherent in the whole text.

In information retrieval and speech recognition, unintelligible words pose a practical problem. The TREC-5 confusion track [8] studied the impact of data corruption introduced by scanning or Optical Character Recognition errors on retrieval performance. In the subsequent spoken document retrieval tracks [9], the errors were introduced by automatic speech recognition.

Structured sparsity has been successfully applied to natural language processing problems, e.g., in [10, 11]. Jenatton et al. [10] apply sparse hierarchical dictionary learning to learn hierarchies of topics from a corpora of NIPS proceedings papers. In a more recent application [11], structured sparsity was used to perform effective feature template selection on three natural language processing tasks: chunking, entity recognition, and dependency parsing.

3 The Method

The novel word is explained as a weighted mixture of Wikipedia senses. Particularly, we assign a vector of coefficients to each novel word—an *interpretation vector*—where each coefficient corresponds to a single Wikipedia sense.

The interpretation vector is determined in two steps. First, we formulate a linear model with a structured sparsity inducing regularization and compute a representation vector $\boldsymbol{\alpha}$. In the second step, this representation vector is condensed to yield an interpretation vector.

We start with a set of Wikipedia senses the novel word could be interpreted as. For each *sense*, we collect a number of *contexts* from Wikipedia. A context of a sense consists of the N non-stopword words occurring before and after the anchor of the link that points to the corresponding Wikipedia page. For example, the anchor text `bar` could point to (and be tagged with) `Bar_(law)`, `Bar_(unit)`, `Bar_(establishment)`, etc. There can be at most $2N$ words in a context.

The presented method makes use of a collection of such *contexts* arranged in a word-context matrix $\mathbf{D}$ [12] (Fig. 1). In this matrix, each context is a column represented as a bag-of-words vector $\mathbf{v}$ of word frequencies, where v_i is the number of occurrences of the ith word in the context.

To compute the representation vector $\boldsymbol{\alpha}$, the context $\mathbf{x} \in \mathbb{R}^m$ of the target word is approximated linearly with the columns of the word-context matrix $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_n] \in \mathbb{R}^{m \times n}$, called the dictionary in the terminology of sparse coding. The columns of the dictionary contain contexts, each labeled with the sense $l_i \in L$ the context was collected for. Please note that multiple contexts can be, and in many cases are,

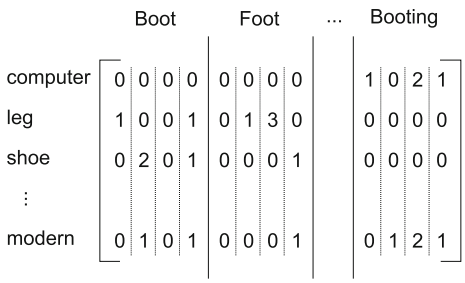


Fig. 1 The word-context matrix **D**. Each column is a context of a Wikipedia sense (e.g., *Boot*, *Foot*). Each element D_{ij} of the matrix holds the number of occurrences of the ith word in the jth context. For example, the word *leg* occurs three times in the 7th context, which is the 3rd context labeled with *Foot*

tagged with the same sense: $l_i = l_j$ is possible. There are m words in the vocabulary, and n contexts in the dictionary.

The representation vector $\boldsymbol{\alpha}$ consists of the coefficients of a linear combination

$$\mathbf{x} \approx \alpha_1 \mathbf{d}_1 + \alpha_2 \mathbf{d}_2 + \cdots + \alpha_n \mathbf{d}_n. \tag{1}$$

For each target word, whose context is $\mathbf{x} \in \mathbb{R}^m$, a representation vector $\boldsymbol{\alpha} = [\alpha_1; \alpha_2; \ldots; \alpha_n] \in \mathbb{R}^n$ is computed.

We introduce the structured sparsity inducing regularization by organizing the contexts in **D** into *groups*. Each group contains the contexts annotated with a single sense. Sparsity on the groups is realized by computing $\boldsymbol{\alpha}$ with a *group Lasso* regularization [13] determined by the labels.

The groups are introduced as a family of sets $\mathcal{G} = \{G_l\}_{l \in L} \subseteq 2^{\{1,\ldots,n\}}$. There are as many sets in $\mathcal{G}$ as there are distinct senses in L. For each sense $l \in L$, there is exactly one set $G_l \in \mathcal{G}$ that contains the indices of all the columns $\mathbf{d}_i$ tagged with l. $\mathcal{G}$ forms a partition.

The representation vector $\boldsymbol{\alpha}$ of the target word whose context is $\mathbf{x}$ is computed as the minimum of the loss function

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^n} \frac{1}{2} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \sum_{l \in L} w_l ||\boldsymbol{\alpha}_{G_l}||_2, \tag{2}$$

where $\boldsymbol{\alpha}_{G_l} \in \mathbb{R}^{|G_l|}$ denotes the vector where only the coordinates present in the set $G_l \subseteq \{1, \ldots, n\}$ are retained.

The first term is the approximation error, the second one realizes the structured sparsity inducing regularization. Parameter $\lambda > 0$ controls the tradeoff between the two terms. The parameters $w_l > 0$ denote the weights for each group G_l.

If each group is a singleton (i.e., $\mathcal{G} = \{\{1\}, \{2\}, \ldots, \{n\}\}$) the Lasso problem [14] is recovered:

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^n} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \sum_{i=1}^{n} w_i |\alpha_i|. \tag{3}$$

Setting $\lambda = 0$ yields the least squares cost function.

For the sake of simplicity, we represent each sense with the same number of contexts: there are an equal number of columns in $\mathbf{D}$ for each label $l \in L$ ($|G_1| = |G_2| = \cdots = |G_{|L|}|$). The weights w_l of the groups are set to 1.

In the *second step*, the target word is disambiguated to a mixture of Wikipedia senses based on the weights in this vector. We utilize the group structure to condense the vector $\boldsymbol{\alpha} \in \mathbb{R}^n$ to a vector $\mathbf{s} \in \mathbb{R}^{|L|}$ where each coordinate corresponds to a single sense. The interpretation vector is obtained by summing the weights in each group $G_l \in \mathcal{G}$. The weight for each sense $l \in L$ in the mixture is

$$s_l = \sum_i (\boldsymbol{\alpha}_{G_l})_i. \tag{4}$$

The structured sparsity inducing regularization fulfills three purposes. Firstly, it allows us to conveniently condense the representation vector $\boldsymbol{\alpha}$ to the interpretation vector $\mathbf{s}$ based on the groups. Secondly, it allows us to explain each target word with only a few senses. This is important mainly for applications where human users interpret the results.

Thirdly, and most importantly, the structured sparsity inducing regularization allows us to reduce the effect of spurious similarities of contexts in the large-scale disambiguation problem, as it selects *whole groups of contexts*.

Each group $G_l \in \mathcal{G}$ contains contexts tagged with the same sense $l \in L$, and only a few groups can be selected. The 2-norm in the loss function favors dense representations: it tries to represent each selected sense densely in the representation vector $\boldsymbol{\alpha}$. The method tends to choose representations where most of the contexts are active in the group of a selected sense over representations where only a few contexts are active. Intuitively, a context that is similar to the context of the target word only by accident—the context in the group of an incorrect sense—won't be selected, as most of the other contexts in its group will be dissimilar, and so inactive. In the group of the correct sense, most of the contexts will be similar and active, so that will be selected instead.

An important consequence of reducing the effect of spurious similarities is increased accuracy in large-scale problems compared to other algorithms (Sects. 4 and 5).

4 Results

We evaluate the proposed method on two tasks for the two types of novel words. In the first task, we use the method to interpret words whose connection between their surface form and their meaning is broken, but the sense they denote is present

in our sense inventory. These include misspelled words, certain neologisms, errors introduced by automatic speech recognition, and the like (Sect. 1).

In the second task, we interpret words with novel meaning. These are words for whom there are no correct senses in our sense inventory. Our expectation is that the meaning of these words can be approximated by mixtures of related senses. We compare the quality of the interpretation vectors to the bag of words contexts by measuring the quality of the clustering they induce.

4.1 The Datasets

The datasets used in our experiments are obtained by randomly sampling the links in Wikipedia. Each dataset consists of contexts tagged with senses $(\mathbf{c}_1, l_1)$, $(\mathbf{c}_2, l_2)$, ... Each tagged context is obtained by processing a link: the bag-of-words vector generated from the context of the anchor text is annotated with the target of the link.

We use the English Wikipedia database dump from October 2010.[4] Disambiguation pages, and articles that are too small to be relevant (i.e., have less than 200 non-stopwords in their texts, or less than 20 incoming and 20 outgoing links) are discarded. Inflected words are reduced to root forms by the Porter stemming algorithm [15].

To produce a dataset, a list of anchor texts are generated that match a number of criteria. These criteria have been chosen to obtain (i) words that are frequent enough to be suitable training examples and (ii) are proper English words. The anchor text has to be a single word between 3 and 20 characters long, must consist of the letters of the English alphabet, must be present in Wikipedia at least 100 times, and must point to at least two different Wikipedia pages, but not to more than 20. It has to occur at least once in *WordNet* [16] and at least three times in the *British National Corpus* [17].

A number of anchor texts are selected from this list randomly, and their linked occurrences are collected along with their N-wide contexts. Each link is processed to obtain a labeled context $(\mathbf{c}_i, l_i)$.

To ensure that there are an equal number of contexts tagged with each sense $l \in L$, d randomly selected contexts are collected for each label. Labels with less than d contexts are discarded. We do not perform feature selection, but we remove the words that appear less than five times across all contexts, in order to discard very rare words.

[4] Downloaded from http://dumps.wikimedia.org/enwiki/.

4.2 Interpreting Novel Words Whose Meaning Is Present in the Sense Inventory

The first task is a disambiguation problem where the algorithm is used to select a *single correct sense* from *all the available senses* in the sense inventory. Given a context $\mathbf{x} \in \mathbb{R}^m$ of a word, the goal is to determine the correct sense $l \in L$. The performance of the algorithms is measured as the accuracy of this classification.

We compare the interpretation vectors computed with group Lasso to three baselines: representations $\boldsymbol{\alpha}$ computed with two different regularizations (least squares and the Lasso) of the linear model described in Sect. 3, and a Support Vector Machine (SVM). The SVM is a multiclass Support Vector Machine with a linear kernel, used successfully for Wikification in previous works [5, 7].

The interpretation vector $\mathbf{s}$ yields a single sense by simply selecting its largest coefficient. Similarly for least squares and the Lasso, the target word is disambiguated to the sense that corresponds to the largest coefficient in $\boldsymbol{\alpha}$. For the SVM, a classification problem is solved using the labeled contexts $(\mathbf{c}_i, l_i)$ as training and test examples.

The minimization problems of both the *Lasso* and *group Lasso* (Eq. 2) are solved by the Sparse Learning with Efficient Projections (SLEP) package [18]. For the *support vector machine*, we use the implementation of LIBSVM [19].

The algorithms are evaluated on five disjoint datasets generated from Wikipedia (Sect. 4.1), each with different senses. We report the mean and standard deviation of the accuracy across these five datasets.

There are $|L| = 1{,}000$ different senses in each dataset, and $d = 50$ contexts annotated with each sense. The algorithms are evaluated on datasets of different sizes (i.e., d and $|L|$ are different), generated from the original five datasets by removing contexts and their labels randomly.

In accord with [20, 21], and others, we use a broad context, $N = 20$. We found that a broad context improves the performance of all four algorithms.

Before evaluating the algorithms, we examined the effect of their parameters on the results. We found that the algorithms are robust: for the Lasso, $\lambda = 0.005$, for the group Lasso, $\lambda = 0.05$, and for the SVM, $C = 1$ was optimal in almost every validation experiment.

In the first evaluation, we examine the effect the number of training examples per candidate sense has on the accuracy of the four algorithms. The starting datasets consist of $|L| = 500$ senses with $d = 10$ contexts (or training examples) each. Stratified 10-fold cross-validation is used to determine the accuracy of the classification: the dataset is partitioned into 10 subsets (the same as d), where each subset contains exactly $|L|$ examples—one annotated with each sense. In one iteration, one subset is used for testing, and the other 9 subsets form the columns of $\mathbf{D}$: there are $|L|$ test examples and $n = (d - 1)|L|$ columns in $\mathbf{D}$ in each iteration. For the SVM, the columns of $\mathbf{D}$ are used as training examples.

To examine the effect of additional contexts, we add contexts to $\mathbf{D}$ for each candidate sense, and examine the change in accuracy. In order to evaluate the effect

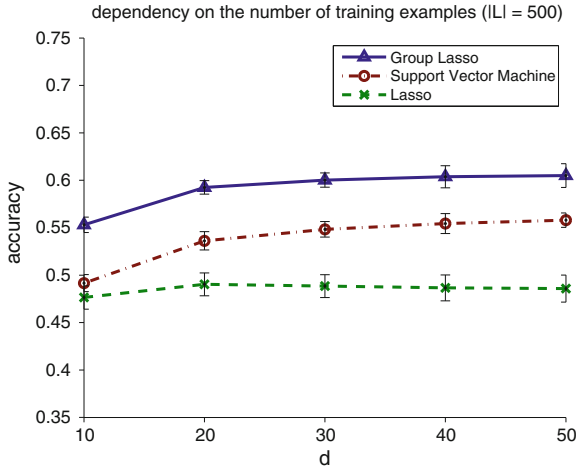


Fig. 2 Dependency of the accuracy on the number of contexts per candidate sense. There are $d-1$ such contexts in each step of the cross-validation, as there is one test example for each sense. The data points are the mean of values obtained on the five datasets. The *error bars* denote the standard deviations. "Group Lasso" means taking the largest weight in the interpretation vector computed with group Lasso. The results of least squares are not illustrated as the standard deviations were very large. It performs consistently below the Lasso

correctly (i.e., to not make the learning problem harder), the test examples remain the same as with $d = 10$. In other words, we perform the same cross-validation as before, only we add additional columns to $\mathbf{D}$ in each step. In Fig. 2, we report the results for $d = 10, 20, 30, 40, 50$.

In the second evaluation, the accuracy of the algorithms is examined as the number of candidate senses $|L|$ increases. As in the first evaluation, there are $d = 10$ examples per candidate sense, and stratified 10-fold cross-validation is performed. Then, the number of examples is raised to $d = 20$ in the same way (i.e., the new examples are not added to the test examples). We report the results for $|L| = 100, 200, \ldots, 1000$ candidate senses in Fig. 3.

4.3 Interpreting Words with Novel Meaning

In this section, we extend our examinations of the presented method to interpret words whose meaning is novel. In practice this means that we remove all knowledge about the senses our target words denote from the dictionary $\mathbf{D}$. The word with novel meaning has to be interpreted based on its relatedness to other, possibly related senses.

Words with novel meaning are simulated by making sure that there is no context in the dictionary tagged with any sense the test examples are tagged with. Wikipedia

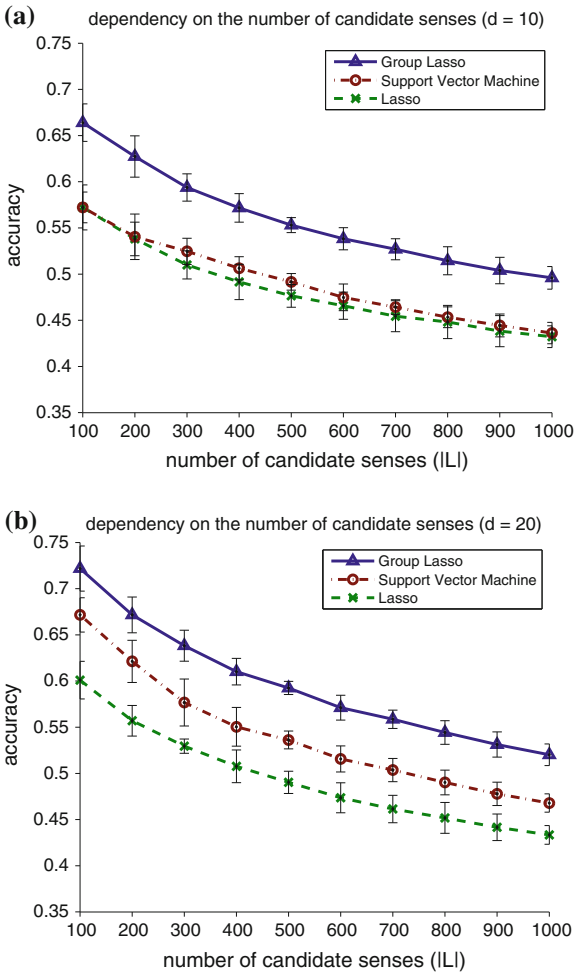


Fig. 3 Dependency of the accuracy on the number of candidate senses, $|L|$. The data points are the mean of values obtained on the five datasets. The *error bars* denote the standard deviations. "Group Lasso" means taking the largest weight in the interpretation vector computed with group Lasso. The results of least squares are not illustrated, as the standard deviations were very large. It performs consistently below the Lasso. **a** dependency on the number of candidate senses (d = 10) **b** dependency on the number of candidate senses (d = 20)

senses in the set T and the contexts tagged by them constitute the test examples (i.e., the contexts of words with novel meaning), while the rest of the senses in L together with their contexts form **D**. The sets T and L, and so the examples for the words with novel meaning and **D** are disjoint: there is not a single context in **D** for any of the senses in T.

The evaluation is based on the labeling of the test examples: for each target word, we already know the sense it denotes. This labeling determines a *clustering* of the resulting interpretation vectors $\mathbf{s} \in \mathbb{R}^{|L|}$: two interpretation vectors belong to the same cluster if and only if they are tagged with the same sense. The quality of the interpretation vectors (the performance of the presented method) is measured as the quality of this clustering.

Clustering quality can be measured by various clustering validation measures [22]. For our purposes, we need to consider different criteria than Liu et al. [22], as we do not evaluate the clustering, but the data. Our measure should be able to compare data in coordinate spaces of different dimension, and it should be somewhat sensitive to noise and clusters of different density. On the other hand, the capability to accurately tell the number of clusters in the dataset is not important for us. Based on these criteria, we chose the well-known *R-squared* measure. R-squared may be considered a measure of the degree of difference between clusters and the degree of homogeneity between groups [23, 24].

If X denotes all the test examples, $\mathbf{c}$ is the center of X, C_t, $t \in T$ are the different clusters, and $\mathbf{c}_t$ are the centers of the clusters, then R-squared is

$$RS = \left(\sum_{\mathbf{x} \in X} \|\mathbf{x} - \mathbf{c}\|_2^2 - \sum_{t \in T} \sum_{\mathbf{x} \in C_t} \|\mathbf{x} - \mathbf{c}_t\|_2^2 \right) / \sum_{\mathbf{x} \in X} \|\mathbf{x} - \mathbf{c}\|_2^2. \tag{5}$$

For these evaluations, we obtain a single dataset by concatenating the five datasets used in the first task into a larger dataset that contains 5,000 senses. The disjoint sets T and L are randomly selected from among these 5,000 senses in each experiment.

Parameter λ was set to $\lambda = 0.05$, the same as in the first task. This value yields interpretation vectors with approximately 30–70 active senses on average. There are $d = 20$ contexts for each sense. We interpret $|T| = 50$ different senses in each experiment, so there are 1,000 target words to interpret. Each experiment is repeated 30 times with different randomly selected senses in both T and L. We report the mean, its standard error, and the standard deviation.

We compare the interpretation vectors to the input bag of words contexts. For each sense $t \in T$, we use the same $d = 20$ contexts that were transformed into the interpretation vectors. For bag of words, we conducted a single set of experiments, as the results do not depend on the value of the parameter $|L|$. We report the results in Fig. 4.

5 Discussion

In the first task, the results are very consistent across the five disjoint datasets, except in the case when the representation vector was computed with least squares. The performance of least squares was the worst of the four algorithms, and it was so erratic that we did not plot it in order to keep the figure uncluttered.

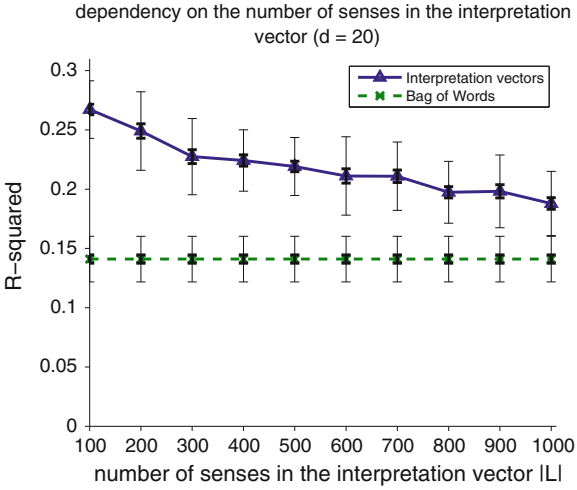


Fig. 4 Interpretation vectors of words with novel meaning versus the bag of words contexts. The R-squared with the bag of words representation is constant, as it does not depend on the number of senses in the interpretation vector, $|L|$. The data points are the mean of 30 experiments. The *thick error bars* denote the standard errors of the mean. The *thin error bars* denote the standard deviations

For group Lasso and the SVM, additional training examples help up to 20 examples per sense (Fig. 2), but only small gains can be achieved by adding more than 20 examples.

The Lasso-based representation does not benefit from new training examples *at all* when there are many candidate senses. This may be the effect of spurious similarities. As more and more contexts are added, the less chance Lasso has to select the right sense from among the candidates.

Classification based on interpretation vectors computed with group Lasso significantly outperforms the other methods, including SVM (Fig. 3). This illustrates the efficiency of our method: structured sparsity decreases the chance of selecting contexts spuriously similar to the context of the target word.

In the second task, we found that even when the correct sense of the novel word is unknown, the interpretation vectors perform much better than the bag of words contexts. This points to the possibility of improving performance in natural language processing tasks by using interpretation vectors instead of a bag of words representation.

As the number of senses in the interpretation vector increases, the learning problem becomes harder, and the performance decreases—similarly to the first task. Although there are more and more senses to represent meaning with, these senses were selected randomly from Wikipedia: the chance for senses that are closely related to the novel meaning to appear is too low to offset the effect of the harder learning problem. Based on this intuition, we believe that there is a promising direction for future improvement of the method.

In these first experiments, we interpreted words with novel meaning as mixtures of senses that were randomly selected from Wikipedia. Our experience suggests that a promising avenue of future research is to preselect the senses systematically based on the context of the target word to increase the chance of closely related senses to appear. We have observed some interesting examples where the (unavailable) novel meaning was represented by a mixture of closely related senses. For example, for the novel meaning `Prime_number`, its hypernym, `Number` was selected. For `Existence`, the method selected `Logos`, `Karma`, and `Eternity`. The most interesting example is that of `Transformers`: it was interpreted as a mixture of `Humanoid`, `Tram`, `Flash_(comics)`, `Cyborg`, and `Hero`. With a slight stretch of the imagination, `Transformers` are `Humanoid` robots (`Cyborg`) that can change into vehicles (`Tram`), and they are also `Heroes` that appear in comic books (`Flash_(comics)`) and animated series.

6 Conclusions

We extended the scope of Wikification to novel words by relaxing its premises: (i) we wikify without using the surface form of the word (ii) to a mixture of Wikipedia senses instead of a single sense.

We identified two types of novel words: words where the connection between their surface form and their meaning is broken, and words where there is no meaning to connect to—the meaning itself is also novel.

We proposed a method capable of wikifying both types of novel words while also dealing with the problem of spuriously similar contexts that intensifies because the disambiguation problem becomes inherently large-scale. The performance of the method was demonstrated on two tasks for the two types of novel words. We found that the method was capable of disambiguating between up to 1,000 Wikipedia senses. Additionally, we used it to explain words with novel meaning as a mixture of other, possibly related senses. This mixture representation compared favorably to the bag of words input contexts.

In these first experiments of interpreting words with novel meaning, the sense inventories were randomly generated from Wikipedia. Our experience suggests that extending the method by constructing the sense inventory in a systematic way based on the context of the target word is a promising direction for future research.

A possible future application of the presented method is the verification of links to Wikipedia. The method assigns a weight to each candidate sense. If the weight corresponding to the target of the link is small in contrast to weights of other pages, the link is probably incorrect.

The method can be generalized, as it can work with arbitrarily labeled text fragments as well as contexts of Wikipedia links. This more general framework may have further applications, as the idea of distributional similarity offers solutions to

many natural language processing problems. For example, topics might be assigned to documents as in centroid-based document classification [25].

Acknowledgments The research has been supported by the 'European Robotic Surgery' EC FP7 grant (no.: 288233). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of other members of the consortium or the European Commission. The research was carried out as part of the EITKIC_12-1-2012-0001 project, which is supported by the Hungarian Government, managed by the National Development Agency, financed by the Research and Technology Innovation Fund and was performed in cooperation with the EIT ICT Labs Budapest Associate Partner Group.

References

1. Mihalcea, R., Csomai, A.: Wikify!: linking documents to encyclopedic knowledge. In: Proceedings of the Conference on Information and Knowledge Management (CIKM), pp. 233–242 (2007)
2. Akmajian, A.: Linguistics: An Introduction to Language and Communication. The MIT press, Cambridge (2001)
3. Harris, Z.: Distributional structure. Word **10**, 146–162 (1954)
4. Bach, F., Jenatton, R., Mairal, J., Obozinski, G.: Optimization with sparsity-inducing penalties. Found. Trends Mach. Learn. **4**, 1–106 (2012)
5. Milne, D., Witten, I.H.: Learning to link with Wikipedia. In: Proceedings of the Conference on Information and Knowledge Management (CIKM), pp. 509–518 (2008)
6. Kulkarni, S., Singh, A., Ramakrishnan, G., Chakrabarti, S.: Collective annotation of Wikipedia entities in web text. In: Proceedings of the Conference on Knowledge Discovery and Data Mining (KDD), pp. 457–466 (2009)
7. Ratinov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to Wikipedia. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp.1375–1384 (2011)
8. Kantor, P.B., Voorhees, E.M.: The TREC-5 confusion track: comparing retrieval methods for scanned text. Inf. Retr. **2**, 165–176 (2000)
9. Garofolo, J.S., Auzanne, C.G.P., Voorhees, E.M.: The TREC spoken document retrieval track: a success story. In: RIAO, pp. 1–20 (2000)
10. Jenatton, R., Mairal, J., Obozinski, G., Bach, F.: Proximal methods for hierarchical sparse coding. J. Mach. Learn. Res. **12**, 2297–2334 (2011)
11. Martins, A.F.T., Smith, N.A., Aguiar, P.M.Q., Figueiredo, M.A.T.: Structured sparsity in structured prediction. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1500–1511 (2011)
12. Turney, P.D., Pantel, P.: From frequency to meaning: vector space models of semantics. J. Artif. Intell. Res. **37**, 141–188 (2010)
13. Yuan, M., Yuan, M., Lin, Y., Lin, Y.: Model selection and estimation in regression with grouped variables. J. R. Stat. Soc. **68**, 49–67 (2006)
14. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. **58**, 267–288 (1994)
15. Porter, M.F.: An algorithm for suffix stripping. Readings in Information Retrieval. Morgan Kaufmann Publishers Inc., San Francisco (1997)
16. Miller, G.A.: WordNet: a lexical database for English. Commun. ACM **38**, 39–41 (1995)
17. BNC Consortium: The British National Corpus, version 2 (BNC World) (2001)
18. Liu, J., Ji, S., Ye, J.: SLEP: Sparse Learning with Efficient Projections. Arizona State University, Tempe (2009)

19. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001)
20. Lee, Y.K., Ng, H.T.: An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 41–48 (2002)
21. Schütze, H.: Automatic word sense discrimination. Comput. Linguist. **24**, 97–123 (1998)
22. Liu, Y., Li, Z., Xiong, H., Gao, X., Wu, J.: Understanding of internal clustering validation measures. In: IEEE 10th International Conference on Data Mining (ICDM), pp. 911–916. IEEE (2010)
23. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: On clustering validation techniques. J. Intell. Inf. Syst. **17**, 107–145 (2001)
24. Sharma, S.: Applied Multivariate Techniques. Wiley, New York (1996)
25. Han, E.H., Karypis, G.: Centroid-based document classification: analysis and experimental results. In: Proceedings of the Conference on Principles of Data Mining and Knowledge Discovery (PKDD), pp. 116–123 (2000)

Measuring Linearity of Planar Curves

Joviša Žunić, Jovanka Pantović and Paul L. Rosin

Abstract In this paper we define a new linearity measure which can be applied to open planar curve segments. We have considered the sum of the distances between the curve end points and the curve centroid. We have shown that this sum is bounded from above by the length of the curve segment considered. In addition, we have proven that this sum equals the length of the curve segment only in the case of straight line segments. Exploiting such a nice characterization of straight line segments, we define a new linearity measure for planar curves. The new measure ranges over the interval (0, 1], and produces the value 1 if and only if the measured line is a perfect straight line segment. Also, the new linearity measure is invariant with respect to translations, rotations and scaling transformations.

Keywords Shape · Shape descriptors · Curves · Linearity measure · Image processing.

1 Introduction

Shape descriptors have been employed in many computer vision and image processing tasks (e.g. image retrieval, object classification, object recognition, object identification, etc.). Different mathematical tools have been used to define the shape descriptors: algebraic invariants [1], Fourier analysis [2], morphological operations [3], integral transformations [4], statistical methods [5], fractal techniques [6], logic [7], combinatorial methods [8], multiscale approaches [9], integral invariants

J. Žunić (✉)
Computer Science, University of Exeter, Exeter EX4 4QF, UK
e-mail: j.zunic@ex.ac.uk

J. Žunić · J. Pantović
Mathematical Institute of the Serbian Academy of Sciences and Arts, Belgrade, Serbia

J. Pantović
Faculty of Technical Sciences, University of Novi Sad, 21000 Novi Sad, Serbia

P.L. Rosin
Cardiff University, School of Computer Science, Cardiff CF24 3AA, UK

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_16

[10], etc. Generally speaking, shape descriptors can be classified into two groups: area based descriptors and boundary based ones. Area based descriptors are more robust (i.e. less sensitive to noise or shape deformations) while boundary based descriptors are more sensitive. A preference for either type of descriptor depends on the application performed and the data available. For example low quality data would require robust descriptors (i.e. area based ones) while high precision tasks would require more sensitive descriptors (i.e. boundary based ones). In the literature so far, more attention has been paid to the area based descriptors, not only because of their robustness but also because they are easier to be efficiently estimated when working with discrete data. Due to the recent proliferation of image verification, identification and recognition systems there is a strong demand for shape properties that can be derived from their boundaries [10–12]. It is worth mentioning that some objects, like human signatures for example, are curves by their nature and area based descriptors cannot be used for their analysis.

In this paper we deal with linearity measures that should indicate the degree to which an open curve segment differs from a perfect straight line segment. Several linearity measures for curve segments are already considered in the literature [13–16].

Perhaps the simplest way to define the linearity measure of an open curve segment is to consider the ratio between the length of the curve considered and the distance between its end points. This is a natural and simple definition which is also called the *straightness index* [17]. It satisfies the following basic requirements for a linearity measure of open curve segments.

- The straightness index varies through the interval $(0, 1]$;
- The straightness index equals 1 only for straight line segments;
- The straightness index is invariant with respect to translation, rotation and scaling transformation on a curve considered.

Also, the straightness index is simple to compute and its behavior can be clearly predicted, i.e. we can see easily which curves have the same linearities, measured by the straightness index. It is obvious that those curves whose end points and the length coincide, have the same straightness index. But the diversity of such curves is huge and the straightness index cannot distinguish among them, which could be a big drawback in certain applications. Some illustrations using simple polygonal curves are shown in Fig. 1.

Fig. 1 Five displayed curves (*solid lines*) have different linearities measured by $\mathcal{L}(\mathcal{C})$. The straightness index has the same value for all five curves

In this paper we define a new linearity measure $\mathcal{L}(\mathcal{C})$ for open curve segments. The new measure satisfies the basic requirements (listed above) which are expected to be satisfied for any curve linearity measure. Since it considers the distance of the end points of the curve to the centroid of the curve, the new measure is also easy to compute. The fact that it uses the curve centroids implies that it takes into account a relative distribution of the curve points.

The paper is organized as follows. Section 2 gives basic definitions and denotations. The new linearity measure for planar open curve segments is in Sect. 3. Several experiments which illustrate the behavior and the classification power of the new linearity measure are provided in Sect. 4. Concluding remarks are in Sect. 5.

2 Definitions and Denotations

Without loss of generality, throughout the paper, it will be assumed (even if not mentioned) that every curve $\mathcal{C}$ has length equal to 1 and is given in an arc-length parametrization. I.e., planar curve segment $\mathcal{C}$ is represented as:

$$x = x(s), \qquad y = y(s), \quad \text{where } s \in [0, 1].$$

The parameter s measures the distance of the point $(x(s), y(s))$ from the curve start point $(x(0), y(0))$, along the curve $\mathcal{C}$.

The centroid of a given (unit length) planar curve $\mathcal{C}$ will be denoted by $(x_{\mathcal{C}}, y_{\mathcal{C}})$ and computed as

$$(x_{\mathcal{C}}, y_{\mathcal{C}}) = \left(\int_{\mathcal{C}} x(s)\, ds, \int_{\mathcal{C}} y(s)\, ds \right). \tag{1}$$

Taking into account that the length of $\mathcal{C}$ is assumed to be equal to 1, we can see that the coordinates of the curve centroid, as defined in (1), are the average values of the curve points.

As usual,

$$d_2(A, B) = \sqrt{(x-u)^2 + (y-v)^2}$$

will denote the Euclidean distance between the points $A = (x, y)$ and $B = (u, v)$.

As mentioned, we introduce a new linearity measure $\mathcal{L}(\mathcal{C})$ which assigns a number from the interval $(0, 1]$. The curve $\mathcal{C}$ is assumed to have the length 1. More precisely, any appearing curve will be scaled by the factor which equals the length of it before the processing. So, an arbitrary curve $\mathcal{C}_a$ would be replaced with the curve $\mathcal{C}$ defined by

$$\mathcal{C} = \frac{1}{\int_{\mathcal{C}_a} ds} \cdot \mathcal{C}_a = \left\{ \left(\frac{x}{\int_{\mathcal{C}_a} ds}, \frac{y}{\int_{\mathcal{C}_a} ds} \right) \mid (x, y) \in \mathcal{C}_a \right\}.$$

Shape descriptors/measures are very useful for discrimination among the objects—in this case open curve segments. Usually the shape descriptors have a clear geometric meaning and, consequently, the shape measures assigned to such descriptors have a predictable behavior. This is an advantage because the suitability of a certain measure to a particular shape-based task (object matching, object classification, etc.) can be predicted to some extent. On the other hand, a shape measure assigns to each object (here curve segment) a single number. In order to increase the performance of shape based tasks, a common approach is to assign a graph (instead of a number) to each object. E.g. such approaches define *shape signature* descriptors, which are also 'graph' representations of planar shapes, often used in shape analysis tasks [18, 19], but they differ from the idea used here and in [16].

We will apply a similar idea here as well. To compare objects considered we use *linearity plots* (the approach is taken from [16] where more details can be found) to provide more information than a single linearity measurement. The idea is to compute linearity incrementally, i.e. to compute linearity of sub-segments of $\mathcal{C}$ determined by the start point of $\mathcal{C}$ and another point which moves along the curve $\mathcal{C}$ from the beginning to the end of $\mathcal{C}$. The linearity plot $\mathrm{P}(\mathcal{C})$, associated with the given curve $\mathcal{C}$ is formally defined as follows.

Definition 1 Let $\mathcal{C}$ be a curve given in an arc-length parametrization: $x = x(s)$, $y = y(s)$, and $s \in [0, 1]$. Let $\mathbf{A}(s)$ be the part of the curve $\mathcal{C}$ bounded by the starting point $(x(0), y(0))$ and by the point $(x(s), y(s)) \in \mathcal{C}$. Then, for a linearity measure $\mathcal{L}$, the linearity plot $\mathrm{P}(\mathcal{C})$ is defined by:

$$\mathrm{P}(\mathcal{C}) = \{(s, \mathcal{L}(\mathbf{A}(s)) \mid \ s \in [0, 1]\}. \tag{2}$$

We will also use the *reverse linearity plot* $\mathrm{P}_{rev}(\mathcal{C})$ defined as:

$$\mathrm{P}_{rev}(\mathcal{C}) = \{(s, \mathcal{L}(\mathbf{A}_{rev}(1 - s)) \mid \ s \in [0, 1]\}, \tag{3}$$

where $\mathbf{A}_{rev}(1 - s)$ is the segment of the curve $\mathcal{C}$ determined by the end point $(x(1), y(1))$ of $\mathcal{C}$ and the point which moves from the end point of $\mathcal{C}$, to the start point of $\mathcal{C}$, along the curve $\mathcal{C}$. In other words, $\mathrm{P}_{rev}(\mathcal{C})$ is the linearity plot of the curve $\mathcal{C}'$ which coincides with the curve $\mathcal{C}$ but the start (end) point of $\mathcal{C}$ is the end (start) point of $\mathcal{C}'$. A parametrization of $\mathcal{C}'$ can be obtained by replacing the parameter s, in the parametrization of $\mathcal{C}$, by a new parameter s' such that $s' = 1 - s$. Obviously such a defined s' measures the distance of the point $(x(s'), y(s'))$ from the starting point $(x(s' = 0), y(s' = 0))$ of $\mathcal{C}'$ along the curve $\mathcal{C}'$, as s' varies through the interval $[0, 1]$.

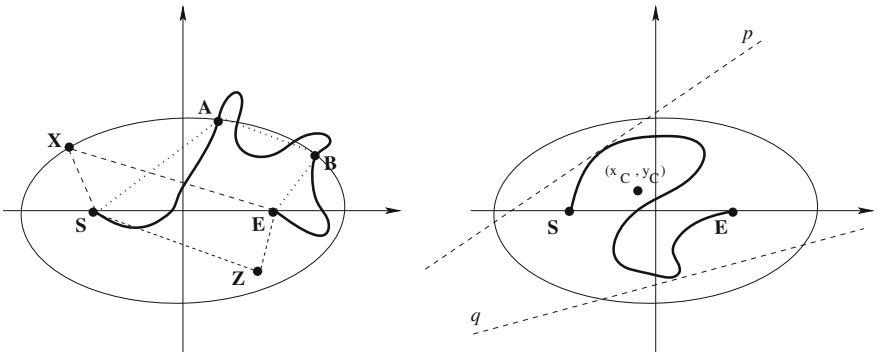


Fig. 2 Terms in the proof of Theorem 1 are illustrated above

3 New Linearity Measure for Open Curve Segments

In this section we introduce a new linearity measure for open planar curve segments.

We start with the following theorem which says that amongst all curves having the same length, straight line segments have the largest sum of distances between the curve end points to the curve centroid. This result will be exploited to define the new linearity measure for open curve segments.

Theorem 1 *Let $\mathcal{C}$ be an open curve segment given in an arc-length parametrization $x = x(s)$, $y = y(s)$, and $s \in [0, 1]$. The following statements hold:*

(a) *The sum of distances of the end points $(x(0), y(0))$ and $(x(1), y(1))$ from the centroid $(x_{\mathcal{C}}, y_{\mathcal{C}})$ of the curve $\mathcal{C}$ is bounded from above by 1, i.e.:*

$$d_2((x(0), y(0)), (x_{\mathcal{C}}, y_{\mathcal{C}})) + d_2((x(1), y(1)), (x_{\mathcal{C}}, y_{\mathcal{C}})) \leq 1. \tag{4}$$

(b) *The upper bound established by the previous item is reached by the straight line segment and, consequently, cannot be improved.*

Proof Let $\mathcal{C}$ be a curve given in an arc-length parametrization: $x = x(s)$ and $y = y(s)$, with $s \in [0, 1]$, and let $S = (x(0), y(0))$ and $E = (x(1), y(1))$ be the end points of $\mathcal{C}$. We can assume, without loss of generality, that the curve segment $\mathcal{C}$ is positioned such that

- the end points S and E belong to the x-axis (i.e. $y(0) = y(1) = 0$), and
- S and E are symmetric with respect to the origin (i.e. $-x(0) = x(1)$),

as illustrated in Fig. 2. Furthermore, let

$$\mathcal{E} = \{X = (x, y) \mid d_2(X, S) + d_2(X, E) = 1\}$$

be an ellipse which consists of points whose sum of distances to the points S and E is equal 1. Now, we prove (a) in two steps.

(i) First, we prove that the curve $\mathcal{C}$ and the ellipse $\mathcal{E}$ do not have more than one intersection point (i.e. $\mathcal{C}$ belongs to the closed region bounded by $\mathcal{E}$).
This will be proven by a contradiction. So, let us assume the contrary, that $\mathcal{C}$ intersects $\mathcal{E}$ at k ($k \geq 2$) points:

$$(x(s_1), y(s_1)),\ (x(s_2), y(s_2)), \ldots,\ (x(s_k), y(s_k)),$$

where $0 < s_1 < s_2 < \cdots < s_k < 1$. Let $A = (x(s_1), y(s_1))$ and $B = (x(s_k), y(s_k))$. Since the sum of the lengths of the straight line segments $[SA]$ and $[AE]$ is equal to 1, the length of the polyline $SABE$ is, by the triangle inequality, bigger than 1. Since the length of the arc $\widehat{SA}$ (along the curve $\mathcal{C}$) is not smaller than the length of the edge $[SA]$, the length of the arc $\widehat{AB}$ (along the curve $\mathcal{C}$) is not smaller than the length of the straight line segment $[AB]$, and the length of the arc $\widehat{BE}$ (along the curve $\mathcal{C}$) is not smaller than the length of the straight line segment $[BE]$, we deduce that the curve $\mathcal{C}$ has length bigger than 1, which is a contradiction. A more formal derivation of the contradiction $1 < 1$ is

$$\begin{aligned} 1 &= d_2(S, A) + d_2(A, E) \\ &< d_2(S, A) + d_2(A, B) + d_2(B, E) \\ &\leq \int_{\widehat{SA}} ds + \int_{\widehat{AB}} ds + \int_{\widehat{BE}} ds = \int_{\mathcal{C}} ds \\ &= 1. \end{aligned} \tag{5}$$

So, $\mathcal{C}$ and $\mathcal{E}$ do not have more than one intersection point, implying that $\mathcal{C}$ lies in the closed region bounded by $\mathcal{E}$.

(ii) Second, we prove that the centroid of $\mathcal{C}$ does not lie outside of $\mathcal{E}$.
The proof follows easily:

- the convex hull $CH(\mathcal{C})$ of $\mathcal{C}$ is the smallest convex set which includes $\mathcal{C}$ and, consequently, is a subset of the region bounded by $\mathcal{E}$;
- The centroid of $\mathcal{C}$ lies in the convex hull $CH(\mathcal{C})$ of $\mathcal{C}$ because it belongs to every half plane which includes $\mathcal{C}$ (the intersection of such half planes is actually the convex hull of $\mathcal{C}$ (see [20]));
- the two items above give the required:

$$(x_{\mathcal{C}}, y_{\mathcal{C}}) \in CH(\mathcal{C}) \subset region\_bounded\_by\_\mathcal{E}.$$

Finally, since the centroid of $\mathcal{C}$ does not lie outside $\mathcal{E}$, the sum of the distances of the centroid $(x_{\mathcal{C}}, y_{\mathcal{C}})$ of $\mathcal{C}$ to the points S and E may not be bigger than 1, i.e.

$$\begin{aligned} & d_2((x(0), y(0)), (x_{\mathcal{C}}, y_{\mathcal{C}})) + d_2((x(1), y(1)), (x_{\mathcal{C}}, y_{\mathcal{C}})) \\ & = d_2(S, (x_{\mathcal{C}}, y_{\mathcal{C}})) + d_2(E, (x_{\mathcal{C}}, y_{\mathcal{C}})) \\ & \leq 1. \end{aligned}$$

This establishes (a).

To prove (b) it is enough to notice that if $\mathcal{C}$ is a straight line segment of length 1, then the sum of its end points to the centroid of $\mathcal{C}$ is 1. □

Now, motivated by the results of Theorem 1, we give the following definition for a new linearity measure $\mathcal{L}(\mathcal{C})$ for open curve segments.

Definition 2 Let $\mathcal{C}$ be an open curve segment, whose length is 1. Then, the linearity measure $\mathcal{L}(\mathcal{C})$ of $\mathcal{C}$ is defined as the sum of distances between the centroid $(x_{\mathcal{C}}, y_{\mathcal{C}})$ of $\mathcal{C}$ and the end points of $\mathcal{C}$. I.e.:

$$\mathcal{L}(\mathcal{C}) = \sqrt{(x(0) - x_{\mathcal{C}})^2 + (y(0) - y_{\mathcal{C}})^2} + \sqrt{(x(1) - x_{\mathcal{C}})^2 + (y(1) - y_{\mathcal{C}})^2}$$

where $x = x(s)$, $y = y(s)$, $s \in [0, 1]$ is an arc-length representation of $\mathcal{C}$.

The following theorem summarizes desirable properties of $\mathcal{L}(\mathcal{C})$.

Theorem 2 *The linearity measure $\mathcal{L}(\mathcal{C})$ has the following properties:*

(i) $\mathcal{L}(\mathcal{C}) \in (0, 1]$, *for all open curve segments $\mathcal{C}$;*
(ii) $\mathcal{L}(\mathcal{C}) = 1 \Leftrightarrow$ *$\mathcal{C}$ is a straight line segment;*
(iii) $\mathcal{L}(\mathcal{C})$ *is invariant with respect to the similarity transformations.*

Proof Item (i) is a direct consequence of Theorem 1.

To prove (ii) we will use the same notations as in the proof of Theorem 1 and give a proof by contradiction. So, let us assume the following:

- the curve $\mathcal{C}$ differs from a straight line segment, and
- the sum of distances between the end points, and the centroid of $\mathcal{C}$ is 1.

Since $\mathcal{C}$ is not a straight line segment, then $\mathbf{d}_2(S, E) < 1$, and the centroid $(x_{\mathcal{C}}, y_{\mathcal{C}})$ lies on the ellipse $\mathcal{E} = \{X = (x, y) \mid \mathbf{d}_2(X, S) + \mathbf{d}_2(X, E) = 1\}$. Further, it would mean that there are points of the curve $\mathcal{C}$ belonging to both hyperplanes determined by the tangent on the ellipse $\mathcal{E}$ passing through the centroid of $\mathcal{C}$. This would contradict the fact that $\mathcal{C}$ and $\mathcal{E}$ do not have more than one intersection point (which was proven as a part of the proof of Theorem 1).

To prove item (iii) it is enough to notice that translations and rotations do not change the distance between the centroid and the end points. Since we assume that $\mathcal{C}$ is represented by using an arc-length parametrization: $x = x(s)$, $y = y(s)$, with the parameter s varying through $[0, 1]$, the new linearity measure $\mathcal{L}(\mathcal{C})$ is invariant with respect to scaling transformations as well. □

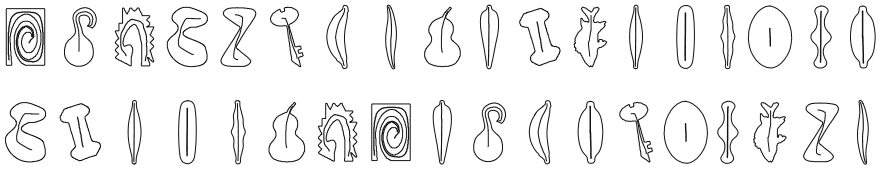


Fig. 3 The shapes are ranking by their axes in increasing order according to linearity $\mathcal{L}$ (*top row*) and sigmoidality S_3 (*bottom row*)

4 Experiments

In this section we provide several experiments in order to illustrate the behavior and efficiency of the linearity measure introduced here.

First Experiment: Illustration. The first example shows in Fig. 3 the results of ranking a set of shapes by the properties of their axes. In the upper row the shapes are ordered according to linearity $\mathcal{L}$ while for comparison in the lower row the shapes are ordered according to Rosin's S_3 sigmoidality measure [21].

Second Experiment: Illustration. The second example shows how the linearity measure can be used as an error measure for polygonal approximation in the same manner as in [16]. For each curve segment the error (i.e. its deviation from linearity) is calculated as $(1 - \mathcal{L}(\mathbf{C}_{ij})) \cdot m_{0,0}(\mathbf{C}_{ij})$, where $\mathbf{C}_{ij}$ denotes the section of curve between $\mathbf{C}(i)$ and $\mathbf{C}(j)$, while $m_{0,0}(\mathbf{C}_{ij})$ is the length of $\mathbf{C}_{ij}$. The optimal polygonal approximation which minimises the summed error over the specified number of curve segments can then be determined using dynamic programming. Results are shown in Fig. 4 of polygonal approximations obtained using different error measures, namely the standard L_2 and L_∞ errors norms on the distances between the curve segment and the corresponding straight line segment as well as an error term based on the linearity measure ($\mathcal{L}_0$) in [16]. It can be seen that there are differences between the polygonal approximations produced by the various error terms, although they are relatively small except for very coarse approximations.

Third Experiment: Illustration. To demonstrate how various shapes produce a range of linearity values, Fig. 5 shows the first two samples of each handwritten digit (0–9) from the training set captured by Alimoğlu and Alpaydin [22] plotted in a $2D$ feature space of linearity $\mathcal{L}(\mathcal{C})$ versus rectilinearity $\mathcal{R}_1(\mathcal{C})$ [23].

Despite the variability of hand writing, most pairs of the same digit are reasonably clustered. The major separations occur for:

- "2" since only one instance has a loop in the middle;
- "4" since the instance next to the pair of "7"s is missing the vertical stroke;
- "5" since the uppermost right instance is missing the horizontal stroke.

The full data set consists of 7,485 digits for training and 3,493 digits for testing. A nearest neighbour classifier using Mahalanobis distances was trained on the training

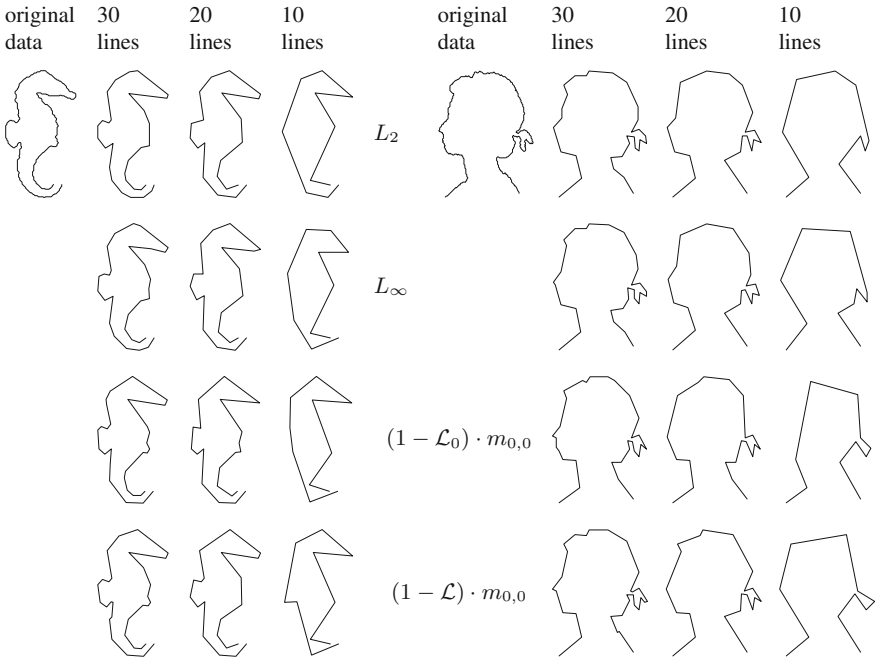


Fig. 4 Two shapes with their optimal polygonal approximations using the L_2 error function (*top row*), L_∞ error function (*second row*), the $(1 - \mathcal{L}_0) \cdot m_{0,0}$ error function (*third row*), and the $(1 - \mathcal{L}) \cdot m_{0,0}$ error function (*fourth row*). Approximations are determined using 30, 20 and 10 line segments

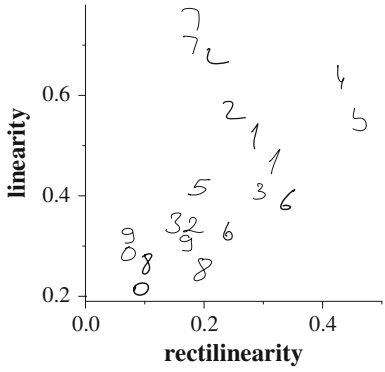


Fig. 5 Handwritten digits ordered by linearity and rectilinearity

data with just linearity as a single feature, and was applied to the test set, producing an accuracy of 21.39 %. This value is low since a richer feature set is required for discrimination. As a step towards obtaining this, the digits were simplified by applying Ramer's polygonal approximation [24] at various thresholds ({2, 4, 8, 16, 32}); some examples are shown in Fig. 6. Linearity was computed at each scale to

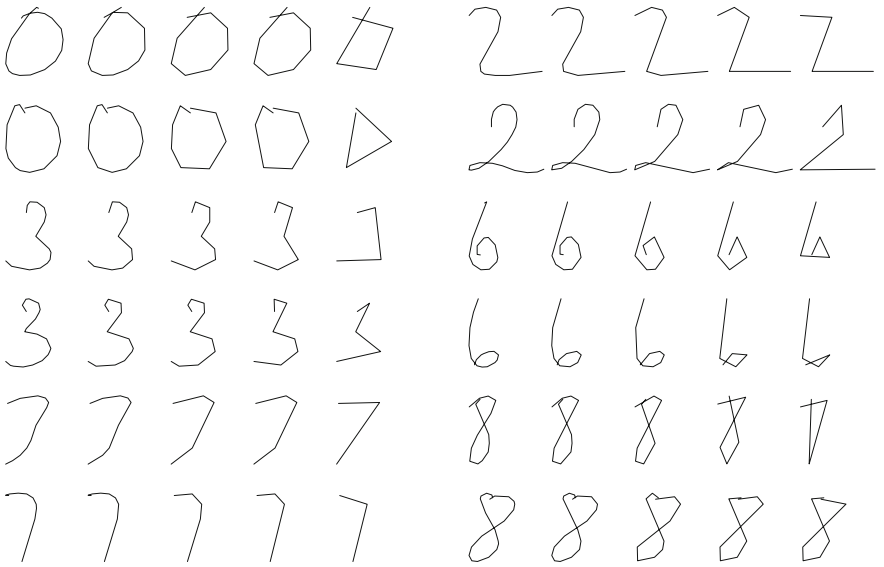


Fig. 6 Examples of digits simplified at five scales

obtain five features per digit, and this improved accuracy to 33.61 %. Finally, by further extending the feature vector to include the first seven Hu moment invariants [1] and six further moment invariants designed for character recognition [25], accuracy was increased to 88.38 %.

Fourth Experiment: Filtering Edges. Figure 7 shows the application of the linearity to filtering edges. The edges were extracted from the images using the Canny detector [26], connected into curves, and then thresholded according to total edge magnitude and length [27]. Linearity was measured in local sections of curve of length 25, and sections above (or below) a linearity threshold were retained. It can be seen that retaining sections of curve with $\mathcal{L}(\mathcal{C}) < 0.5$ finds small noisy or corner sections. Keeping sections of curve with $\mathcal{L}(\mathcal{C}) > 0.9$ or $\mathcal{L}(\mathcal{C}) > 0.95$ identifies most of the significant structures in the image.

Experiments are also shown in which Poisson image reconstruction is performed from the image gradients [28]. In the middle column of Fig. 8 all the connected edges with minimum length of 25 pixels (shown in the first column in Fig. 7) are used as a mask to eliminate all other edges before reconstruction. Some fine detail is removed as expected since small and weak edges have been removed in the pre-processing stage.

When linearity filtering is applied, and only edges corresponding to sections of curve with $\mathcal{L}(\mathcal{C}) > 0.95$ are used (see the fourth column in Fig. 7) then the image reconstruction (Fig. 8 right column) retains only regions that are locally linear structures (including sections of large curved objects).

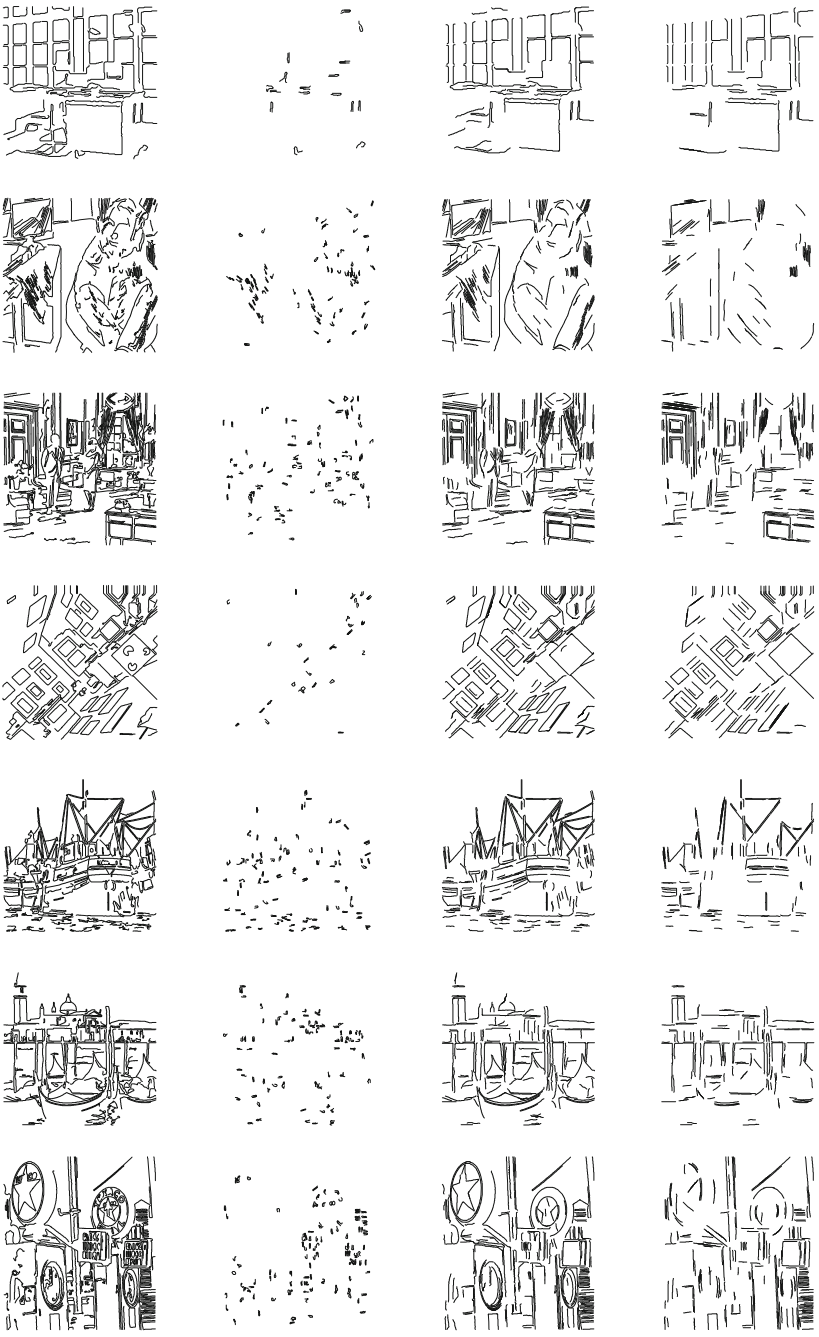


Fig. 7 Filtering connected edges by linearity. *Left/first column* connected edges (minimum length: 25 pixels); *second column* sections of curve with $\mathcal{L}(C) < 0.5$; *third column* sections of curve with $\mathcal{L}(C) > 0.9$; *fourth column* sections of curve with $\mathcal{L}(C) > 0.95$

Fig. 8 Reconstructing the image from its filtered edges. (*left column*) original intensity image; (*middle column*) image reconstructed using all connected edges (minimum length: 25 pixels); (*right column*) image reconstructed using sections of curve with $\mathcal{L}(\mathcal{C}) > 0.95$

Fig. 9 Examples of genuine *(first three columns)* and forged (*last three columns*) signatures

Fifth Experiment: Signature Verification. For this application we use data from Munich and Perona [29] to perform signature verification. The data consists of pen trajectories for 2,911 genuine signatures taken from 112 subjects, plus five forgers provided a total of 1,061 forgeries across all the subjects. Examples of corresponding genuine and forged signatures are shown in Fig. 9. To compare signatures we use the linearity plots defined by (2) and (3) to provide more information than a single linearity measurement. Linearity plot examples are in Fig. 10.

The quality of match between signatures $\mathcal{C}_1$ and $\mathcal{C}_2$ is measured by the similarity between the linearity plots $\mathrm{P}(\mathcal{C}_1)$ and $\mathrm{P}(\mathcal{C}_2)$. This similarity is measured by the area bounded by the linearity plots $\mathrm{P}(\mathcal{C}_1)$ and $\mathrm{P}(\mathcal{C}_2)$ and by the vertical lines $s = 0$ and

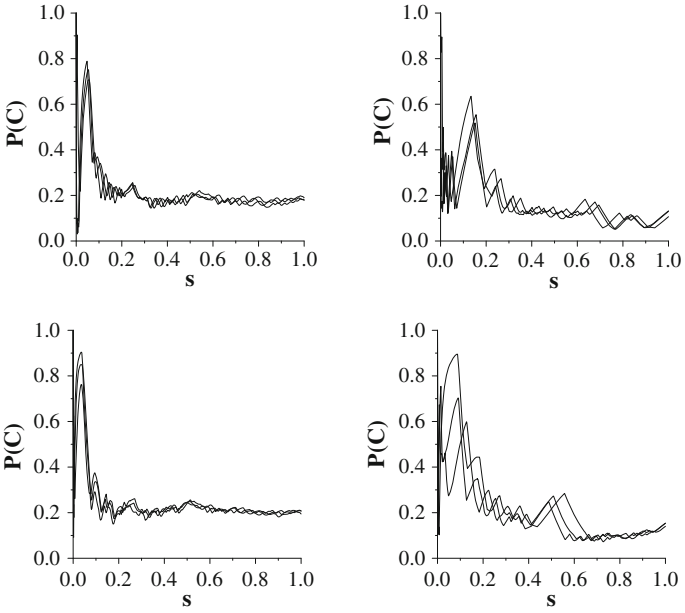


Fig. 10 Examples of linearity plots for the genuine signatures (*top row*) and the forged signatures (*bottom row*) in Fig. 9

$s = 1$. Figure 10 demonstrates the linearity plots for the signatures shown in Fig. 9. The plots in the first (second respectively) row contain the three genuine (forged respectively) signatures from Fig. 9.

Nearest neighbour matching is then performed on all the data using the leave-one-out strategy. Signature verification is a two class (genuine or fake) problem. Since the identity of the signature is already known, the nearest neighbour matching is only applied to the set of genuine and forged examples of the subject's signature. Computing linearity of the signatures using $\mathcal{L}(\mathcal{C})$ produces 96.9 % accuracy. This improves the results obtained by using the linearity measure defined in [16] which achieved 93.1 % accuracy.

5 Conclusions

This paper has described a new shape measure $\mathcal{L}(\mathcal{C})$ for computing the linearity of open curve segments. For a given unit length curve $\mathcal{C}$, its assigned linearity measure $\mathcal{L}(\mathcal{C})$ is computed as the sum of the distances of the end points of $\mathcal{C}$ to the centroid of $\mathcal{C}$. Of course, if the curve considered has an arbitrary length, then the assigned linearity measure is computed as the ratio of the sum of distances of the curve end

points to the curve centroid and the curve length. Such a defined open curve linearity measure $\mathcal{L}(\mathcal{C})$ satisfies the basic requirements for a linearity measure:

- $\mathcal{L}(\mathcal{C})$ is in the interval (0, 1];
- $\mathcal{L}(\mathcal{C})$ equals 1 only for straight line segments;
- $\mathcal{L}(\mathcal{C})$ is invariant with respect to translation, rotation and scaling transformations on the curve.

In addition $\mathcal{L}(\mathcal{C})$ is both extremely simple to implement and efficient to compute.

The effectiveness of the new linearity measure is demonstrated on a variety of tasks. Since the linearity measure $\mathcal{L}(\mathcal{C})$ is a single number, in order to increase the discrimination power in object classification tasks, based on a use of $\mathcal{L}(\mathcal{C})$, we have employed two methods. The first one is based on a use of *linearity plots*, where the quantity $\mathcal{L}(\mathcal{C})$ is replaced with a graph. The second one is based on an idea from [30]: (i) Several shapes (i.e. open curves) were computed from an object (i.e. digit curves in the presented example) by applying a tunable polygonal approximation algorithm; (ii) $\mathcal{L}(\mathcal{C})$ values, assigned to each of such shapes/curves, were used for the classification.

Acknowledgments This work is partially supported by the Serbian Ministry of Science and Technology/project III44006/OI174008.

References

1. Hu, M.: Visual pattern recognition by moment invariants. IRE Trans. Inf. Theory **8**(2), 179–187 (1962)
2. Bowman, E., Soga, K., Drummond, T.: Particle shape characterisation using Fourier descriptor analysis. Geotechnique **51**(6), 545–554 (2001)
3. Ruberto, C.D., Dempster, A.: Circularity measures based on mathematical morphology. Electron. Lett. **36**(20), 1691–1693 (2000)
4. Rahtu, E., Salo, M., Heikkilä, J.: A new convexity measure based on a probabilistic interpretation of images. IEEE Trans. Patt. Anal. Mach. Intell. **28**(9), 1501–1512 (2006)
5. Melter, R., Stojmenović, I., Žunić, J.: A new characterization of digital lines by least square fits. Pattern Recognit. Lett. **14**(2), 83–88 (1993)
6. Imre, A.: Fractal dimension of time-indexed paths. Appl. Math. Comput. **207**(1), 221–229 (2009)
7. Schweitzer, H., Straach, J.: Utilizing moment invariants and Gröbner bases to reason about shapes. Comput. Intell. **14**(4), 461–474 (1998)
8. Acketa, D., Žunić, J.: On the number of linear partitions of the (m, n)-grid. Inf. Process. Lett. **38**(3), 163–168 (1991)
9. Direkoglu, C., Nixon, M.: Shape classification via image-based multiscale description. Pattern Recognit. **44**(9), 2134–2146 (2011)
10. Manay, S., Cremers, D., Hong, B.W., Yezzi, A., Soatto, S.: Integral invariants for shape matching. IEEE Trans. Pattern Anal. Mach. Intell. **28**(10), 1602–1618 (2006)
11. Mio, W., Srivastava, A., Joshi, S.: On shape of plane elastic curves. Int. J. Comput. Vis. **73**(3), 307–324 (2007)
12. Stojmenović, M., Žunić, J.: Measuring elongation from shape boundary. J. Math. Imaging Vis. **30**(1), 73–85 (2008)

13. Gautama, T., Mandić, D., Hull, M.V.: A novel method for determining the nature of time series. IEEE Trans. Biomed. Eng. **51**(5), 728–736 (2004)
14. Gautama, T., Mandić, D., Hulle, M.V.: Signal nonlinearity in fMRI: a comparison between BOLD and MION. IEEE Trans. Med. Images **22**(5), 636–644 (2003)
15. Stojmenović, M., Nayak, A., Žunić, J.: Measuring linearity of planar point sets. Pattern Recognit. **41**(8), 2503–2511 (2008)
16. Žunić, J., Rosin, P.: Measuring linearity of open planar curve segments. Image Vis. Comput. **29**(12), 873–879 (2011)
17. Benhamou, S.: How to reliably estimate the tortuosity of an animal's path: straightness, sinuosity, or fractal dimension? J. Theoret. Biol. **229**(2), 209–220 (2004)
18. El-ghazal, A., Basir, O., Belkasim, S.: Farthest point distance: a new shape signature for Fourier descriptors. Signal Process. Image Commun. **24**(7), 572–586 (2009)
19. Zhang, D., Lu, G.: Study and evaluation of different Fourier methods for image retrieval. Image and Vision Computing **23**(1), 3349 (2005)
20. Valentine, F.: Convex Sets. McGraw-Hill, New York (1964)
21. Rosin, P.: Measuring sigmoidality. Pattern Recognit. **37**(8), 1735–1744 (2004)
22. Alimoğlu, F., Alpaydin, E.: Combining multiple representations for pen-based handwritten digit recognition. ELEKTRIK: Turk. J. Electr. Eng. Comput. Sci. **9**(1), 1–12 (2001)
23. Žunić, J., Rosin, P.: Rectilinearity measurements for polygons. IEEE Trans. Patt. Anal. Mach. Intell. **25**(9), 1193–3200 (2003)
24. Ramer, U.: An iterative procedure for the polygonal approximation of plane curves. Comput. Graph. Image Process. **1**, 244–256 (1972)
25. Pan, F., Keane, M.: A new set of moment invariants for handwritten numeral recognition. In: IEEE International Conference on Image Processing, pp. 154–158 (1994)
26. Canny, J.: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **8**(6), 679–698 (1986)
27. Rosin, P.: Edges: saliency measures and automatic thresholding. Mach. Vis. Appl. **9**(4), 139–159 (1997)
28. Pérez, P., Gangnet, M., Blake, A.: Poisson image editing. ACM Trans. Graph. **22**(3), 313–318 (2003)
29. Munich, M., Perona, P.: Visual identification by signature tracking. IEEE Trans. Patt. Anal. Mach. Intell. **25**(2), 200–217 (2003)
30. Aktaş, M., Žunić, J.: A family of shape ellipticity measures for galaxy classification. SIAM J. Imaging Sci. **6**(2), 765–781 (2013)

Video Segmentation Framework Based on Multi-kernel Representations and Feature Relevance Analysis for Object Classification

S. Molina-Giraldo, J. Carvajal-González, A.M. Álvarez-Meza and G. Castellanos-Domínguez

Abstract A video segmentation framework to automatically detect moving objects in a scene using static cameras is proposed. Using Multiple Kernel Representations, we aim to enhance the data separability into the scene by incorporating multiple information sources into the process, and employing a relevance analysis each source is automatically weighted. A tuned Kmeans technique is employed to group pixels as static or moving objects. Moreover, the proposed methodology is tested for the classification of people and abandoned objects. Attained results over real-world datasets, show how our approach is stable using the same parameters for all experiments.

Keywords Background subtraction · Multiple kernel learning · Relevance analysis · Data separability

1 Introduction

A system that monitors an area by camera and detects moving people or objects is called a surveillance system. Intelligent video surveillance systems can achieve unsupervised results using video segmentation, with which the moving objects can be extracted from video sequences. Many segmentation algorithms have been proposed. Among them, algorithms with background subtraction usually show superior performance [1]. Background subtraction is a typical and crucial process for a surveillance system to detect moving objects that may enter, leave, move or left unattended in the

S. Molina-Giraldo (✉) · J. Carvajal-González · A.M. Álvarez-Meza · G. Castellanos-Domínguez
Signal Processing and Recognition Group, Universidad Nacional de Colombia, Manizales, Colombia
e-mail: smolinag@unal.edu.co

J. Carvajal-González
e-mail: jpcarvajalg@unal.edu.co

A.M. Álvarez-Meza
e-mail: amalvarezme@unal.edu.co

G. Castellanos-Domínguez
e-mail: cgcastellanosd@unal.edu.co

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_17

surveillance region. Unattended objects as bags or boxes in public premises such as airports, terminal bus and train stations are a threat for these places because they can be used as a mean of terrorist attacks, especially for bombs [2].

Image sequences with dynamic backgrounds often cause false classification of pixels, one common solution is to map alternate color spaces, however it has fallen to solve this problem and an enhanced solution is the use of image features, where the distributions at each pixel may be modelled in a parametric manner using a mixture of Gaussians [3] or using non-parametric kernel density estimation [4]. The self organizing maps have been also explored as an alternative for the background subtraction task, because of their nature to learn by means of a self-organized manner local variations [5], however, these techniques have the drawback of manually setting a large number of parameters.

In this work, a methodology called Weighted Gaussian Kernel Video Segmentation (WGKVS) is proposed, which aims to construct a background model and then, incorporating multiple information sources by a MKL framework, performs a background subtraction enhancing thus the representation of each pixel. A relevance analysis for the automatic weight selection of the MKL approach is made. Furthermore, a tuned Kmeans technique is employed to group pixels as static or moving objects. The proposed WGKVS is tested in the surveillance task of the classification of abandoned objects in the scene. In this regard, using the segmented frame, the objects detected as not belonging to the background model that are spatially split, are relabelled as new independent objects and then characterized with the methodology implemented in [2] for further classification.

The remainder of this work is organized as follows. In Sect. 2, the proposed methodology is described. In Sect. 3, the experiments and results are presented. Finally, in Sects. 4 and 5 we discuss and conclude about the attained results.

2 Theoretical Background

2.1 Background Initialization

The first step of the proposed WGKVS is a background model initialization. Given an image sequence $\boldsymbol{H}$ with q frames, we propose to use a subsequence of frames $\boldsymbol{H}_{(t:k)}$ to initialize a background model based on the approach exposed in [6]. This approach, using an optical flow algorithm is successfully able to construct a statistical background model with the most likely static pixels during the subsequence for each RGB component, and it also computes its standard deviation. We also propose to compute a background model using the normalized RGB components (rgb) in order to suppress the shadows casted by the moving objects as described in Ref. [4]. Hence, a background model is stored in a matrix $\boldsymbol{Y}$.

2.2 Background Subtraction Based on Multi-kernel Learning and Feature Representation

Recently, machine learning approaches have shown that the use of multiple kernels, instead of only one, can be useful to improve the interpretation of data [7]. Given a frame $\boldsymbol{F}$ from the image sequence $\boldsymbol{H}$ and a background model $\boldsymbol{Y}$ obtained from the same sequence, using a set of p feature representations for each frame pixel $f_i = \{f_i^z : z = 1, \ldots, p\}$ and each pixel $y_i = \{y_i^z : z = 1, \ldots, p\}$ belonging to the background model, based on the Multi-Kernel Learning (MKL) methods [8], a background subtraction procedure can be computed via the function:

$$\kappa_\omega \left(f_i^z, y_j^z \right) = \omega_z \kappa \left(f_i^z, y_i^z \right), \tag{1}$$

subject to $\omega_z \geq 0$, and $\sum_{i=1}^{p} \omega_z = 1$ ($\forall \omega_z \in \mathbb{R}$). Regarding to video segmentation procedures, each pixel of each frame $\boldsymbol{F}$ can be represented by a dissimilarity measure with a background model by using p different image features (e.g. Color components, textures), in order to enhance the performance of further segmentation stages. Specifically, we propose to use the RGB and the rgb components as features and as basis kernel $\kappa\{\cdot\}$, a Gaussian kernel $\boldsymbol{G}$ defined as:

$$\boldsymbol{G}^z \left(f_i^z, y_i^z \right) = \exp \left(-\frac{1}{2} \left(\frac{\left| f_i^z - y_i^z \right|}{\sigma_i^z} \right)^2 \right), \tag{2}$$

where σ_i^z corresponds to a percentage of the standard deviation of pixel y_i in the feature z from the background model.

As it can be seen from (1), it is necessary to fix the ω_z free parameters, to take advantage, as well as possible of each feature representation. To complete the feature space, the row m and column position n are added as features, in order to keep the local relationships among pixels. Therefore, a feature space $\boldsymbol{X}_{((m \times n) \times 8)}$ is obtained.

2.3 MKL Weight Selection Based on Feature Relevance Analysis

Using the feature space $\boldsymbol{X}$, we aim to select the weights values ω_z in MKL by means of a relevance analysis. This type of analysis is applied to find out a low-dimensional representations, searching for directions with greater variance to project the data, such as Principal Component Analysis (PCA), which is useful to quantify the relevance of the original features, providing weighting factors taking into consideration that the best representation from an explained variance point of view will be reached [9]. Given a set of features $\left(\boldsymbol{\eta}_z : z = 1, \ldots, p\right)$ corresponding to each column of the input data matrix $\mathbf{X} \in \mathbb{R}^{r \times p}$ (a set of p features describing a pixel image h_i), the relevance of $\boldsymbol{\eta}_z$ can be identified as ω_z, which is calculated as $\boldsymbol{\omega} = \sum_{j=1}^{d} \left| \lambda_j \mathbf{v}_j \right|$,

with $\boldsymbol{\omega} \in \mathbb{R}^{p\times 1}$, and where λ_j and $\mathbf{v}_j$ are the eigenvalues and eigenvectors of the covariance matrix $\mathbf{V} = \mathbf{X}^\top\mathbf{X}$, respectively.

Therefore, the main assumption is that the largest values of ω_z lead to the best input attributes. The d value is fixed as the number of dimensions needed to conserve a percentage of the input data variability. Then using $\boldsymbol{\omega}$, a weighted feature space is computed as: $\hat{\boldsymbol{X}} = \boldsymbol{X} \times diag(\boldsymbol{\omega})$.

2.4 Video Segmentation Based on Kmeans Clustering Algorithm

In order to segment the moving objects, a Kmeans clustering algorithm with a fixed number of clusters equal to two is employed over $\hat{\boldsymbol{X}}$, hence, the objects that do not belong to the background model (moving objects) are grouped in a cluster and the objects that belong to the background model (static objects) in the other one. Initially, the clusters are located at the coordinates given by the matrix $\boldsymbol{Q}$, which is obtained by the cluster initialization algorithm called *maxmin* described in Ref. [10], making the segmentation process more stable. Finally, with the attained labels $\boldsymbol{l}$, using a post-process stage, groups of pixels detected as moving objects that do not surpass a value u of minimum size for an object are deleted. The results are stored into a binary matrix $\boldsymbol{S}$. In Fig. 1 is illustrated the general scheme for WGKVS.

2.5 Quantitative Measures

For measuring the accuracy of the proposed methodology for moving object segmentation, three different pixel-based measures have been adopted:

- $Recall = t_p/\left(t_p + f_n\right)$
- $Precision = t_p/\left(t_p + f_p\right)$
- $Similarity = t_p/\left(t_p + f_n + f_p\right)$

where t_p (true positives), f_p (false positives) and f_n (false negatives) are obtained while comparing against a hand-segmented ground truth. A method is considered

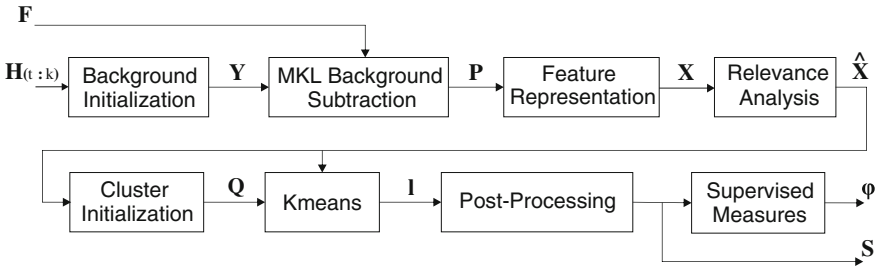


Fig. 1 WGKVS scheme

good if it reaches high *Recall* measures, without sacrificing *Precision*. *Similarity* has been adopted with the only aim of further comparing the results achieved by other proposed algorithms.

2.6 Object Characterization and Classification

The WGKVS approach is applied into a real world surveillance task: the classification of abandoned objects. Using the segmented frame $\boldsymbol{S}$, the groups detected as moving objects that are spatially split, are relabelled as new independent objects. With these new labels, each object is enclosed in a bounding box, and using the characterization process described in Ref. [2], each object is represented by 14 geometrical and 7 statistical features. A Knn classifier is trained using images belonging to the classes: people and baggage objects.

3 Experiments

The proposed methodology is tested using three different Databases. Each Database includes image sequences that represent typical situations for testing video surveillance systems. Following, the Databases are described.

A-Star-Perception: This Database is publicly available at http://perception.i2r.a-star.edu.sg. It contains 9 image sequences recorded in different scenes. Hand-segmented ground truths are available for each sequence, thus, supervised measures can be used. For concrete testing, the sequences: WaterSurface, Fountain, Shopping-Mall and Hall are used. The first two sequences are recorded in outdoor scenarios which present high variations due to their nature, hence the segmentation process posses a considerable challenge. The other two sequences are recorded in public halls, in which are present many moving objects casting strong shadows and crossing each other, hindering the segmentation task.

Left-Packages: Publicly available at http://homepages.inf.ed.ac.uk/rbf/ CAVIAR-DATA1, this Database contains five different image sequences recorded at an interior scenario which has several illumination changes. The main purpose of this database is the identification of abandoned objects (a box and a bag). For testing, hand-segmented ground truths from randomly selected frames are made.

MSA: This Database is publicly available at http://cvprlab.uniparthenope.it. It contains a single indoor sequence, with stable lighting conditions, nonetheless, strong shadows are casted by the moving objects. The purpose of this sequence is also the detection of abandoned objects, in this case a briefcase. Hand-segmented ground truths from randomly selected frames are made in order to give a quantitative measure.

Three different experiments are performed, in all of them, the free parameter σ_i^z is heuristically set as five times the standard deviation of each pixel representation

y_i^z. The minimum size of a detected moving object u is set as a percentage of the total size of the image $0.005 \times (m \times n)$.

The first experiment aims to prove the effectiveness of the proposed WGKVS approach when incorporating more information sources into the segmentation process with an automatic weighting selection. To this end, the image sequences WaterSurface, Fountain, ShoppingMall, Hall, LeftBag and LeftBox are used. The WGKVS segmentation results are compared against GKVS (WGKVS with all equal weights), and traditional GKVS-RGB (GKVS using only RGB components). In Fig. 2 are shown the different segmentation results for the frame 1,523 of the sequence WaterSurface. The relevance weights are shown in Fig. 3. In Tables 1, 2 and 3 are exposed the attained results for each method.

The second type of experiments are performed to compare the WGKVS algorithm against a traditional video segmentation algorithm named Self-Organizing Approach to Background Subtraction (SOBS), which builds a background model by learning

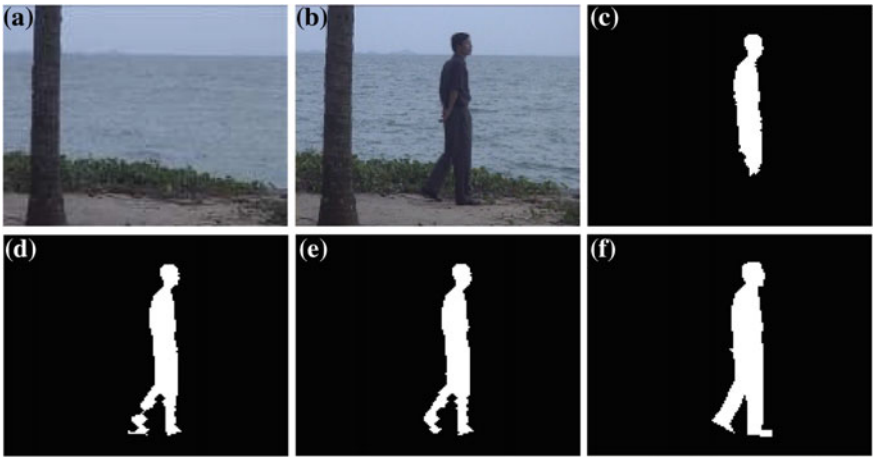


Fig. 2 WaterSurface (Frame 1523). **a** Background model, **b** Original frame, **c** GKVS-RGB, **d** GKVS, **e** WGKVS, **f** Ground truth

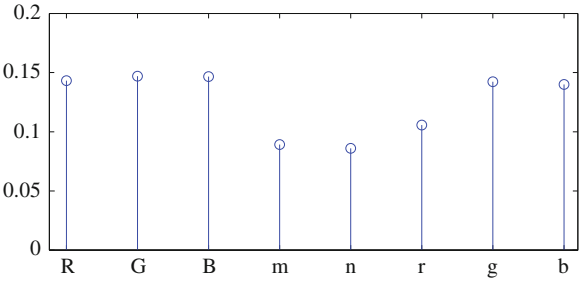


Fig. 3 Relevance weights for sequence WaterSurface (Frame 1523)

Table 1 Segmentation performance for GKVS-RGB

Video	Recall	Precision	Similarity
WaterSurface	0.677	0.995	0.676
Fountain	0.509	0.897	0.480
ShoppingMall	0.436	0.385	0.302
Hall	0.489	0.809	0.434
LeftBag	0.610	0.839	0.555
LeftBox	0.697	0.906	0.647

Table 2 Segmentation performance for GKVS

Video	Recall	Precision	Similarity
WaterSurface	0.762	0.995	0.759
Fountain	0.559	0.909	0.528
ShoppingMall	0.571	0.680	0.442
Hall	0.518	0.829	0.462
LeftBag	0.614	0.842	0.560
LeftBox	0.699	0.910	0.651

Table 3 Segmentation performance for WGKVS

Video	Recall	Precision	Similarity
WaterSurface	0.770	0.994	0.767
Fountain	0.587	0.908	0.552
ShoppingMall	0.643	0.715	0.512
Hall	0.520	0.837	0.473
LeftBag	0.627	0.848	0.571
LeftBox	0.729	0.915	0.674

in a self-organizing manner the scene variations, and detects moving object by using a background subtraction [5]. The SOBS video segmentation approach has been used as a reference to compare video segmentation approaches and it has been also included in surveillance systems surveys [11]. The software for the SOBS approach is publicly available at http://cvprlab.uniparthenope.it/index.php/dow-nload/92.html. For testing, the 10 parameters of the SOBS approach are left as default. In Figs. 4 and 5 are the segmentation results using WGKVS and SOBS for the frame 0996 of the sequence LeftBag and frame 1980 of the sequence ShoppingMall respectively. In Table 4 are the segmentation results for the SOBS algorithm.

Finally, the third type of experiment is made in order to test the proposed WGKVS for the classification of abandoned objects. In this sense, the process described in Sect. 2.6 is employed. For testing, the sequences: LeftBag, LeftBox and MSA are used. The aim is to classify objects as: people or baggage objects (e.g. briefcases, boxes, backpacks, suitcases). A knn classifier is trained using a dataset of 70 images of people and 82 images of baggage objects, and as validation, we use the objects segmented by the WGKVS. It is important to remark, that the objects from the dataset

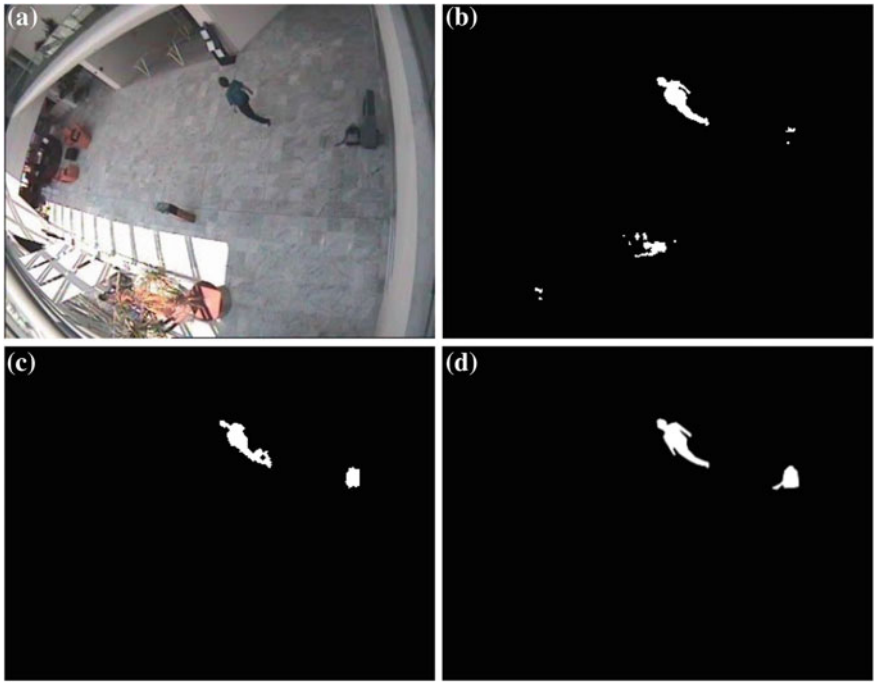


Fig. 4 LeftBag (Frame 0996). **a** Original frame, **b** SOBS, **c** WGKVS, **d** Ground truth

used for training are characterized by the same process. In Fig. 6, are shown some resulting bounded objects. In total, 38 objects are used in the validation database, 11 belong to the baggage objects class and 27 to the people class. In Fig. 7, are shown two samples of the characterization process for a person and a bag. The classification results are exposed in Table 5.

4 Discussion

From the attained results of experiment one, it can be seen that when working only with the RGB components, the method does not perform very good, lacking of extra information that could enhance the clustering process (see Fig. 2c and Table 1). When the rgb components and the spatial information are incorporated, the performance improves by a 9.95 % of the similarity measure (see Fig. 2d and Table 2). Using the proposed WGKVS methodology, the best results are achieved improving the similarity measure by 4.32 % over the GKVS (Fig. 2e and Table 3). The results for the second experiment, expose that the proposed WGKVS methodology clearly surpass the attained results of the SOBS algorithm using its default parameters, and as can be seen in Figs. 4 and 5, our approach achieves more reliable results for further stages

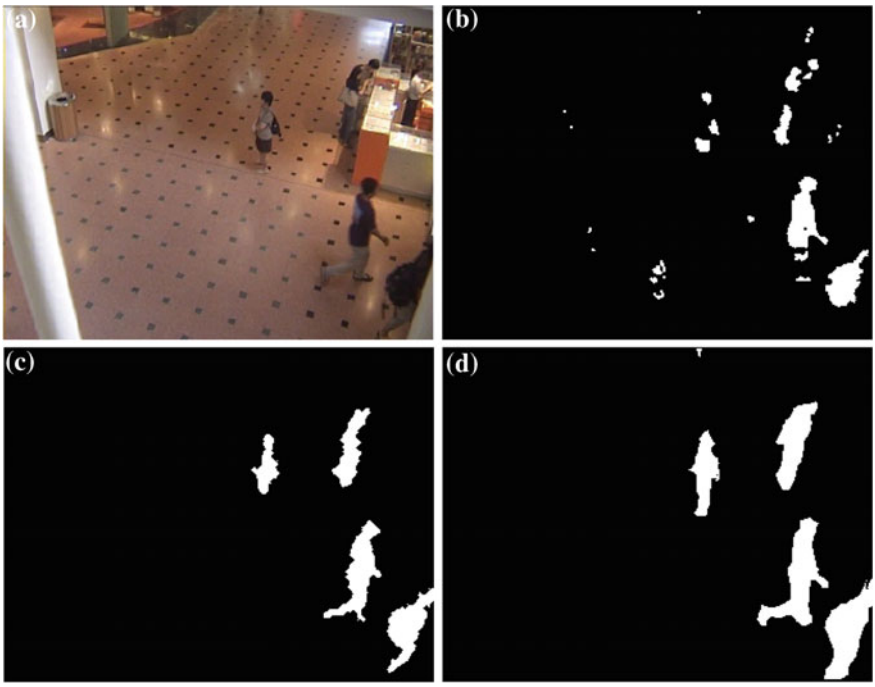


Fig. 5 ShoppingMall (Frame 1980). **a** Original frame, **b** SOBS, **c** WGKVS, **d** Ground truth

Table 4 Segmentation performance for SOBS

Video	Recall	Precision	Similarity
WaterSurface	0.709	0.998	0.708
Fountain	0.349	0.971	0.346
ShoppingMall	0.522	0.861	0.482
Hall	0.708	0.888	0.648
LeftBag	0.472	0.642	0.373
LeftBox	0.746	0.806	0.634

like the classification of objects. The obtained segmented objects by the WGKVS for the third experiment (see Fig. 6), are accurate for an adequate characterization process (see Fig. 7). The latter can be corroborated with a classification performance of 84.21 %. The misclassified samples belonging to the people class, are objects where the complete body of the person is not in the scene.

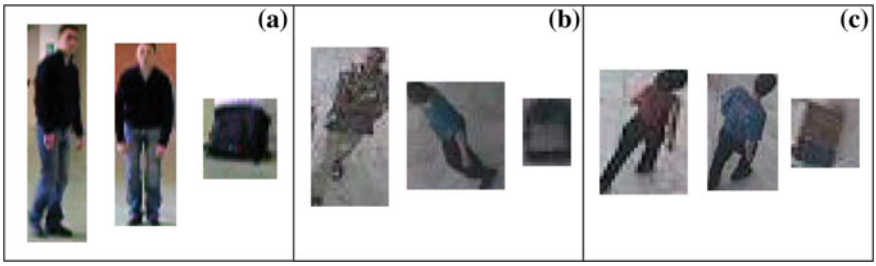


Fig. 6 Segmented object samples using WGKVS. **a** MSA, **b** LeftBag, **c** LeftBox

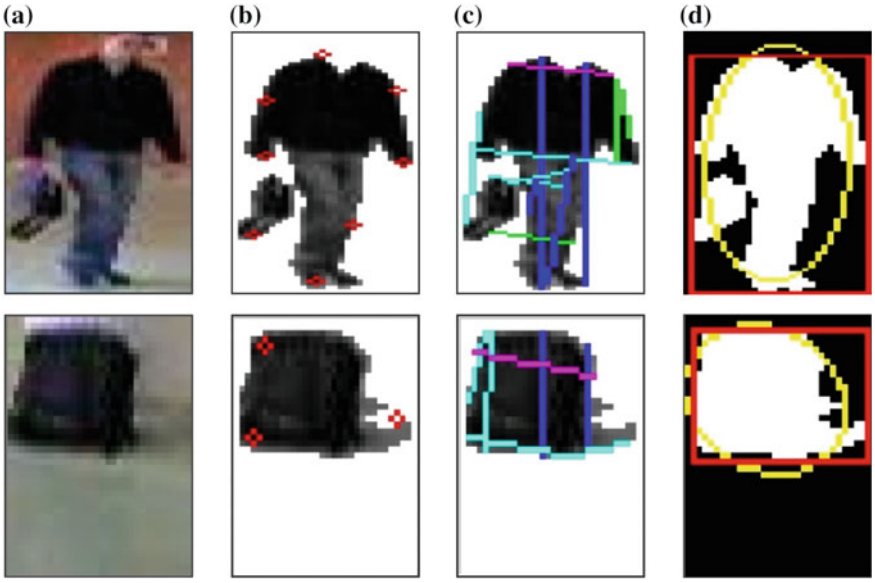


Fig. 7 Geometrical features examples. **a** Original object, **b** Corners, **c** Lines, **d** Fitting shapes

Table 5 Confusion matrix using the Knn classifier

	People	Baggage objects
People	21	6
Baggage objects	1	10

5 Conclusions

We have proposed a methodology called WGKVS, which using image sequences recorded by stationary cameras, segments the moving objects from the scene. The aim of the proposed WGKVS is to construct a background model based on an optical flow methodology, and using a MKL background subtraction approach, incorporates different information sources, each source is weighted using a relevance analysis

and a tuned Kmeans algorithm is used to segment the resulting weighted feature space. Experiments showed that the weighted incorporation of the spatial and rgb features enhances the data separability for further clustering procedures. Moreover, the attained results expose that the proposed WGKVS has stable results using the same parameters for all the experiments, and that it is suitable for supporting real surveillance applications like the classification of abandoned objects. As future work, the inclusion of other features which could enhance the process and a methodology for the automatic actualization of the background model are to be studied. Furthermore, the proposed WGKVS is to be implemented as a real time application.

Acknowledgments This research was carried out under grants provided by a M.Sc. and a Ph.D. scholarship provided by Universidad Nacional de Colombia, and the project 15,795, funded by Universidad Nacional de Colombia.

References

1. Chen, T.W., Hsu, S.C., Chien, S.Y.: Robust video object segmentation based on k-means background clustering and watershed in ill-conditioned surveillance systems. In: IEEE International Conference on Multimedia and Expo, pp. 787–790 (2007)
2. González, J.C., Álvarez-Meza, A., Castellanos-Domínguez, G.: Feature selection by relevance analysis for abandoned object classification. In: CIARP, pp. 837–844 (2012)
3. Klare, B., Sarkar, S.: Background subtraction in varying illuminations using an ensemble based on an enlarged feature set. In: Computer Vision and Pattern Recognition Workshops, IEEE Computer Society Conference on CVPR Workshops, pp. 66–73 (2009)
4. Elgammal, A., Duraiswami, R., Harwood, D., Davis, L.: Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. Proc. IEEE **90**, 1151–1163 (2002)
5. Maddalena, L., Petrosino, A.: A self-organizing approach to background subtraction for visual surveillance applications. IEEE Trans. Image Process. **17**, 1168–1177 (2008)
6. Gutchess, D., Trajkovics, M., Cohen-Solal, E., Lyons, D., Jain, A.: A background model initialization algorithm for video surveillance. In: Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on IEEE, vol. 1, pp. 733–740 (2001)
7. Rakotomamonjy, A., Bach, F.R., Canu, S., Grandvalet, Y.: SimpleMKL. J. Mach. Learn. Res. **9**, 2491–2521 (2008)
8. Gonen, M., Alpaydin, E.: Localized multiple kernel regression. In: Proceedings of the 20th International Conference on Pattern Recognition (ICPR) (2010)
9. Daza-Santacoloma, G., Arias-Londoo, J.D., Godino-Llorente, J.I., Senz-Lechn, N., Osma-Ruz, V., Castellanos-Domnguez, G.: Dynamic feature extraction: an application to voice pathology detection. Intell. Autom. Soft Comput. **15**(4), 665–680 (2009)
10. Cuesta-Frau, D., Pérez-Cortés, J., Andreu-Garcia, G.: Clustering of electrocardiograph signals in computer-aided holter analysis. Comput. Methods Programs Biomed. **72**, 179–196 (2003)
11. Raty, T.: Survey on contemporary remote surveillance systems for public safety. IEEE Trans. Man, Cybern., Part C: Appl. Rev. **40**, 493–515 (2010)

Quality-Based Super Resolution for Degraded Iris Recognition

Nadia Othman, Nesma Houmani and Bernadette Dorizzi

Abstract In this paper we address the problem of low-quality iris recognition via super resolution approaches. We introduce two novel quality measures, one computed Globally (GQ) and the other Locally (LQ), for fusing at the pixel level (after a bilinear interpolation step) the images corresponding to several shots of a given person. These measures derive from a local GMM probabilistic characterization of good quality iris texture. We performed two types of experiments. The first one considers low resolution video sequences coming from the MBGC portal database: it shows the superiority of our approach compared to score-based or average image-based fusion methods. Moreover, we show that the LQ-based fusion outperforms the GQ-based fusion with a relative improvement of 4.79 % at the Equal Error Rate functioning point. The second experiment is performed on CASIA v4 database containing sequences of still images with degraded quality resulting in severe segmentation errors. We show that the image fusion scheme improves greatly the performance and that the LQ-based fusion is mainly interesting for low FAR values.

Keywords Iris recognition · Video · Quality · Super resolution · Fusion of images

1 Introduction

The excellent performance of biometric systems based on the iris are obtained by controlling the quality of the images captured by the sensors, by imposing certain constraints on the enrolled subjects, such as standing at a fixed distance from the

N. Othman (✉) · B. Dorizzi
CNRS UMR 5157 SAMOVAR, Institut Mines-Télécom/Télécom SudParis, 9 Rue Charles Fourier, 91011 Evry, France
e-mail: nadia.othman@telecom-sudparis.eu

B. Dorizzi
e-mail: bernadette.dorizzi@telecom-sudparis.eu

N. Houmani
Laboratoire SIGMA, ESPCI-ParisTech, 10 Rue Vauquelin, 75005 Paris, France
e-mail: nesma.houmani@espci.fr

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_18

camera and looking directly at it, and by using algorithmic measurements of the image quality (contrast, illumination, texture richness, etc.).

However, when working with moving subjects, as in the context of video surveillance or portal scenarios for border crossing, many of these constraints become impossible to impose. An "iris on the move" (IOM) person recognition system was evaluated by the NIST by organizing the Multiple Biometric Grand Challenge (MBGC) [1]. The image of the iris is acquired using a static camera as the person is walking toward the portal. Hence, a sequence of images of the person's face is acquired, which normally contain the areas of the eyes (periocular region).

The results of the MBGC show degradation in performance of iris systems in comparison to the IREX III evaluation [2] based on databases acquired in static mode. At a 1 % False Acceptance Rate (FAR), the algorithm that gave the best results in both competitions obtains 92 % of correct verification on the MBGC database, as compared to 98.3 % on the IREX III database [2].

Indeed, acquisition from a distance causes a loss in quality of the resulting images, showing a lack of resolution, often presenting blur and low contrast between the boundaries of the different parts of the iris.

Nevertheless, images acquired in still conditions may also suffer from degradation and variability due to the presence of eyeglasses, specular reflections and dilatation. This results in segmentation errors, which influence particularly the quality of the normalized iris texture. In that case two normalized iris images resulting from different acquisitions of the same person may present a high discrepancy. Therefore, if the segmentation module is not reliable enough, one can notice a significant decrease of the recognition performance on such degraded normalized images.

One way to try to circumvent this issue is to use some redundancy that arises from the availability of several images of the same eye in the database. These images can be extracted either from a video sequence (case of MBGC) or from a set of images acquired at different moments as in CASIA v4. A first approach consists in fusing the scores coming from the frame-by-frame matching (1 to 1) using some operators like the mean or the min. This has been shown to be efficient but at the price of a high computational cost [3]. Another direction is to fuse the images at the pixel level, exploiting this way the redundancy of the iris texture at an early stage and to perform the feature extraction and matching steps on the resulting fused images. At this level of study, the arising question is how to perform this fusion stage so that the performance can be improved compared to 1 to 1 or score fusion schemes.

At our knowledge, few authors have considered the problem of fusing images of low quality in iris videos for improving recognition performance. The first paper is that of Fahmy [4] who proposed a super resolution technique based on an autoregressive signature model for obtaining high resolution images from successive low resolution ones. He shows that the resulting images are valuable only if the initial low-resolution images are blur-free and focused, stressing already the bad influence of low quality images in the fusion. In [3], Hollingsworth et al. proposed to perform a simple averaging of the normalized iris images extracted from the video for matching NIR videos against NIR videos from the MBGC database. When compared to a fusion of scores, the results are similar but with a reduced complexity. In the same spirit,

Nguyen et al. [5, 6] proposed to fuse different images of the video at a pixel level after an interpolation of the images. They use a quality factor in their fusion scheme, which allows giving less importance to images of bad quality. The interpolation step is shown very efficient as well as the quality weighting for improving recognition performance. Note that they considered a protocol similar to MBGC, where they compare a video to a high quality still image. More recent works [7, 8] explored the fusion in the feature domain using PCA or PCT but not on the same MBGC protocol as they usually degrade artificially the image resolution in their assessment stage.

For still images, these fusion techniques do not seem to have been envisaged maybe because of the computational burden and the need of multi-shot acquisitions as compared to simple 1 to 1 comparison.

In this work, we will consider the two above-mentioned situations, namely video sequences of low resolution resulting from an acquisition at a distance and sequences of still images, presenting variability and therefore segmentation defaults. We will follow the same approach in the two situations.

In our work, like in [6], we propose to fuse the different frames of the sequence at the pixel level, after an interpolation stage that allows increasing the size of the resulting image by a factor of 2. When dealing with videos, contrary to [6], we do not follow the MBGC protocol that compares a video to a still high quality image reference. Indeed, we consider in our work, a video against video scenario, more adapted to the re-identification context, meaning that we will use several frames in both low quality videos to address the person recognition task. In the still images context, we will consider different scenarios including the fusion or not of several test images, that we will discuss comparatively.

The above literature review dealing with super resolution in the iris on the move context has stressed the importance of choosing adequately the images involved in the fusion process. Indeed, integration of low quality images leads to a decrease in performance producing a rather counterproductive effect.

In this work, we will therefore concentrate our efforts in the proposition of a novel way of measuring and integrating quality measures in the image fusion scheme. More precisely, our first contribution is the proposition of a global quality measure for normalized iris images, defined in [9], as a weighting factor in the same way as proposed in [6]. The interest of our quality measure compared to [6] is its simplicity and the fact that its computation does not require identifying in advance the type of degradations that can occur in the iris images. Indeed, our measure exploits a local Gaussian Mixture Model-based characterization of the iris texture. Bad quality normalized iris images are therefore images containing a large part of non-textured zones, resulting from segmentation errors or blur.

Taking benefit of this local measure, we propose as a second novel contribution to perform a local weighting in the image fusion scheme, allowing this way to take into account the fact that degradations can be different in different parts of the iris image. This means that regions free from occlusions will contribute more in the reconstruction of the fused image than regions with artifacts, such as eyelid or eyelash occlusion and specular reflection. Thus, the quality of the reconstructed image will

be better and we expect this scheme to lead to a significant improvement in the recognition performance.

This paper is organized as follows. In Sect. 2, we describe our proposed Local and Global quality-based super resolution approaches. In Sect. 3, we present the comparative experiments that we performed on the MBGC and CASIA v4 databases. Finally, conclusions are given in Sect. 4.

2 Local and Global Quality-Based Super Resolution

In this Section, we first briefly describe the iris recognition system used in this work. Then, we recall the definition of the used local and global quality measures applied on the normalized images (for more details refer to [9, 10]). We also explain how we have adapted such measures to the context of iris images of a given sequence. After that, we describe the super resolution process allowing interpolation and fusion of images. Finally, we summarize the global architecture of the system that we propose for person recognition from a sequence of iris images using these local and global quality measures. Note that a sequence can be either frames of a video or several stills images, of a given person.

2.1 The Iris Recognition System OSIRISv2

In the present work, we use the open source iris recognition system OSIRISv2, inspired by Daugman's approach [11], which was developed in the framework of the BioSecure project [12]. The segmentation part uses the circular Hough transform and an active contour approach to detect the contours of the iris and of the pupil as circles. The normalization step is based on Daugman's rubber-sheet model. The classification part is based on Gabor phase demodulation and Hamming distance classification.

However, in the case of the video scenario, we used a manual segmentation instead of the one given by OSIRISv2. Indeed, one of the difficulties encountered in the videos of the MBGC database lies in the very low contrast between the boundaries of the iris, the pupil and the sclera, for which OSIRISv2 is not efficient. Moreover, sometimes, light spots are present on these boundaries, which cause border detection errors. For this reason, we performed a manual segmentation of the circular iris boundaries. The advantage of this protocol is that it allows a good segmentation of the iris texture and this way we can assess the interest of our approach on the problem of decrease of resolution independently of segmentation problems. The impact of segmentation errors will be studied exclusively in our second part of experiments with still images of CASIA v4 database for which we will use the OSIRISv2 segmentation module. In both cases, we use OSIRISv2 for the normalization, feature extraction and matching steps.

2.2 Local Quality Measure

As in [10], we use a Gaussian Mixture Model (GMM) to give a probabilistic measure of the quality of local regions of the iris. In this work, the GMM is learned on small images extracted from the MBGC and Casia v4 databases showing a good quality texture free from occlusions. So, giving to this GMM a normalized input iris image, this model will give a low probability on the noisy regions, which result from blur or artifacts as shown in [9]. The interest of this approach is that there is no need to recognize in advance the type of noise present in the images such as eyelid or eyelash occlusion, specular reflection and blur.

For the video context, we trained the GMM with 3 Gaussians on 95 sub-images free from occlusions, selected manually from 30 normalized images taken randomly from MBGC database. For the still images context, we also trained the GMM with 3 Gaussians on 85 sub-images free from occlusions, taken manually from 25 normalized images of CASIA V4 database. In the same way as in [9], the model is based on four local observations grouped in the input vector: the intensity of the pixel, the local mean, the local variance and the local contrast measured in a 5×5 neighborhood of the pixel. The quality measure associated to a sub-image of an image is given by the formula:

$$Q_{local}(w) = exp^{-\frac{1}{d}\sum_{i=1}^{d} |\log(p(x_i/\lambda)) - \bar{a}|} \tag{1}$$

where d is the size of the sub-image (w), x_i is the input vector of our GMM, $p(x_i/\lambda)$ is the likelihood given by the GMM λ to the input vector x_i, and $\bar{a}$ is the mean log-likelihood on the training set. We use a negative exponential to obtain a value between 0 and 1. The closest Q value will be to 1, the highest are the chances that the sub-image w is of good quality, namely free from occlusion and highly textured.

2.3 Global Quality Measure

The local measure presented in Sect. 2.2 can also be employed for defining a global measure of the quality of the entire image. To this end, we divide the normalized image (of size 64×512) in overlapping sub-images of size 8×16 and we average the probabilities given by the local GMM of each sub-image as follows:

$$Q_{global} = \frac{1}{N} \sum_{n} Q_{local}(w_n) \tag{2}$$

where N is the number of sub-images and $Q_{local}(w_n)$ is the GMM local quality of the nth sub-image.

2.4 Super Resolution Implementation

MBGC's images suffer from poor resolution, which degrades significantly iris recognition performance. Super resolution (SR) approaches can remedy to this problem by generating high-resolution images from low-resolution ones. Super resolution can also be seen as a way of implementing fusion at the image level. For this reason, even if the still images of CASIA v4 do not suffer from low resolution, we tested this approach on such sequences as a way to compensate, via a fusion procedure, the decrease of quality resulting from segmentation errors.

Among the various SR schemes, we chose in this work a simple version similar to that exploited in [5], resulting into a double resolution image using a bilinear interpolation. After interpolating each normalized image of the sequence, a step of registration is generally needed before pixel's fusion to ensure that those pixels are correctly aligned with each other in the sequence. In the present work, experiments showed that implementing a registration step did not produce any better recognition performance. Indeed, the process of normalization already performs a scaling of the iris zone, allowing an alignment of the pixels, which is sufficient for the present implementation of super resolution.

This set of normalized interpolated images is then fused to obtain one high-resolution image. We introduce some quality measures in this fusion process. More precisely, as done in [5], we weight the value of each pixel of each image by the same factor, namely the Global Quality (GQ) (defined in Sect. 2.3) of the corresponding image. We also propose a novel scheme using our Local Quality (LQ) measure (defined in Sect. 2.2). In this latter case, we compute the local quality measures of all the sub-images as defined in Sect. 2.3 and we generate a matrix of the same size as the normalized image which contains the values of the quality of each sub-image. This matrix is then bilinearly interpolated. Finally, we weight the value of each pixel of each interpolated image by its corresponding value in the interpolated quality matrix. Figure 1 illustrates this LQ-based fusion process, which is more detailed in Sect. 2.5.

2.5 Architecture of the Local Quality-Based System

Figure 2 presents the general architecture of our LQ-based system. The main steps of such system are described as follow:

For each image of the sequence:

- Segment the iris using two non-concentric circles approximation for the pupillary and limbic boundaries,
- Normalize the segmented iris zone with Daugman's rubber sheet technique,
- Generate masks and measure the local quality on the normalized and masked images, using the GMM already learned,
- Interpolate the normalized images and their corresponding masks and local quality matrix to a double resolution using the bilinear interpolation.

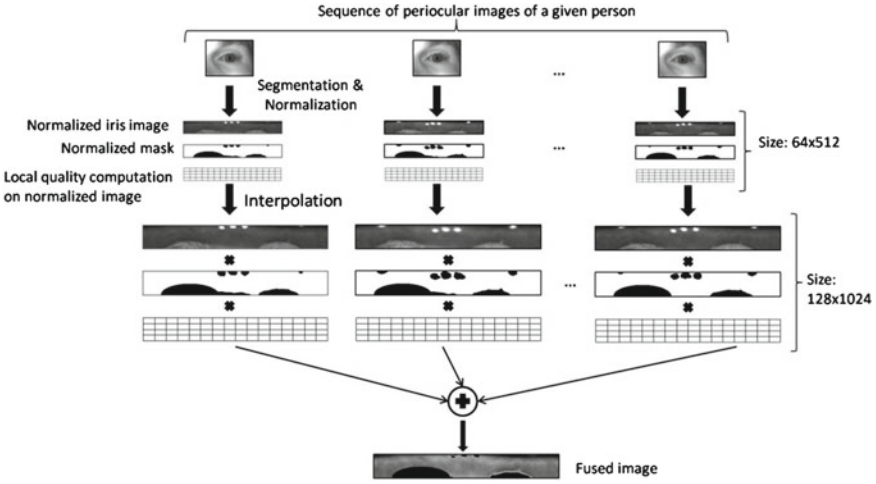


Fig. 1 Fusion process of the proposed local quality-based method

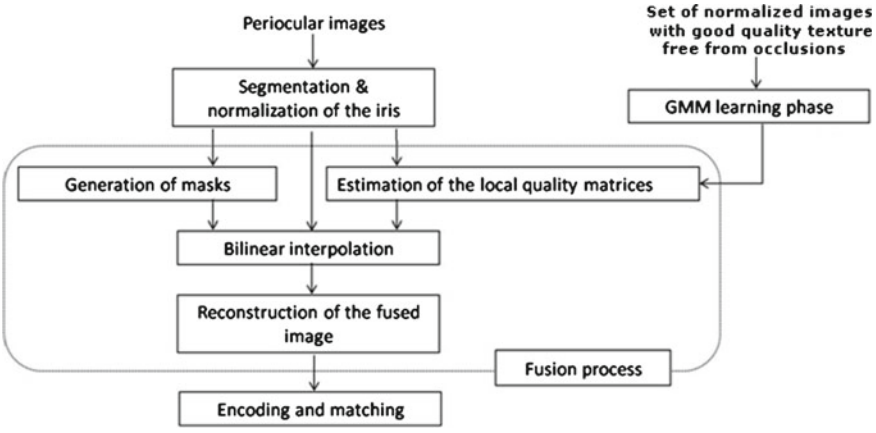


Fig. 2 Diagram of the local quality-based system for video-based iris recognition

Finally, for all the images, generate the fused image as follows:

$$I_{fused} = \frac{\sum_{i=1}^{F} I^i(x, y) * M^i(x, y) * Q^i(w)}{\sum_{i=1}^{F} M^i(x, y) * Q^i(w)} \tag{3}$$

where F is the total number of images, $I^i(x, y)$ and $M^i(x, y)$ are the values of the pixel in the position (x, y) of, respectively, the ith interpolated normalized image and mask. $Q^i(w)$ is the local quality of the sub-image (w) to which the pixel (x, y) belongs.

The last steps of the recognition process, namely feature extraction and matching (as recalled previously in Sect. 2.1), are performed on the fused reconstructed images. Note that from one sequence of F images, we obtain only one image performing this way an important and efficient compression of the information.

3 Experimental Results

As already mentioned, the proposed method has been evaluated on two challenging databases: Multiple Biometric Grand Challenge (MBGC) database and CASIA-Iris-Thousand (CASIA v4) database. The images of the first one suffer from poor resolution and blur. The images of the second database are better in terms of quality but they are considered as difficult to segment due to the spots on the boundaries of the pupil and the iris and the presence of eyeglasses. In this way, we can show the interest of our fusion scheme for two distinct problems for iris recognition, which are the bad quality of the images and the wrong segmentation of the iris.

3.1 Multiple Biometric Grand Challenge's Results

Database and Protocols. The proposed method has been evaluated on the portal dataset composed of Near Infra-Red (NIR) faces videos used during the Multiple Biometric Grand Challenge (MBGC) organized by the National Institute of Standards and Technology (NIST) [1]. This MBGC database was acquired by capturing facial videos of 129 subjects walking through a portal located at 3 m from a NIR camera. Although the resolution of the frames in the video is $2{,}048 \times 2{,}048$, the number of pixels across the iris is about 120, which is below the minimum of 140 pixels considered as the minimum to ensure a good level of performance [11]. The images suffer not only from low resolution but also from motion blur, occlusion, specular reflection and high variation of illumination between the frames. Examples of bad quality images are shown in Fig. 3.

To segment the iris, first we have to detect and track the eyes in the sequence. The detection is generally guided by the presence of spots that are located around

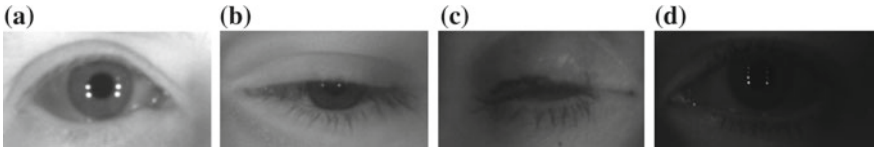


Fig. 3 Examples of bad quality images: **a** out of focus, **b** eyelid and eyelashes occlusions, **c** *closed* eye, **d** *dark* contrast

the eyes in the MBGC database. Due to the important variation of illumination that can be observed between the frames across one sequence, we manually discard darker frames as done in [3, 5]. After that, blurred frames from the sequence were removed by using wavelet's transformation. After this pre-processing process, the used database is composed of 108 subjects; each one possesses two sequences with at least 4 frames per sequence.

As mentioned in Sect. 2.1, we perform a manual segmentation of the iris boundaries, which provides the center coordinates and the diameter parameter of the two circles defining the iris area. We then use OSIRISv2 for the normalization, feature extraction and matching steps. For finding the occlusion masks, we use an adaptive filter similar to that proposed in [13] but adapted to images extracted from a video sequence.

In this work, we compare NIR videos to NIR videos like in [3]. For each person, we use the first sequence as a target and the second one as a query.

Experiments and Results. The proposed approach is compared to other fusion score methods such as Multi-Gallery Simple-Probe (MGSP), Multi-Gallery Multi-Probe (MGMP) and also to fusion signal methods as simple averaging of images and weighted super-resolution.

3.1.1 Fusion at the Score Level

- *Matching 1 to 1:* All the frames in the video of a person are considered as independent images and used for performing inter-class and intra-class comparisons. This system was used as a baseline system to compare the other methods.
- *Matching N to 1, Multi-Gallery Simple-Probe:* In this case, the different images in the video are considered dependent as they represent the same person. If the number of samples in the gallery and the probe are respectively N and 1 per person, we get N Hamming distance scores which can be fused by making a simple average [14] or the minimum of all the scores [15].
- *Matching N to M, Multi-Gallery Multi-Probe:* In this case, we consider M images in the probe and N images in the gallery. We thus get N^*M scores per person and combine them by taking the average or the minimum.

The performance results of these score fusion schemes are shown in Table 1.

As shown in Table 1, the best score's fusion scheme (MGMP) reduces the Equal Error Rate (EER) from 14.32 to 4.66 %. This indicates that recognition performance

Table 1 Equal Error Rate (EER) values of the score's fusion methods

Methods	EER (in %)	
Matching 1 to 1 (baseline)	14.32	
	Minimum	Average
Matching N to 1(MGSP)	9.30	10.27
Matching M to N (MGMP)	4.66	5.65

can be further improved by the redundancy brought by the video. However, the corresponding matching time increases considerably when the recognition score is calculated for N^*M matchings.

3.1.2 Fusion at the Signal Level

- *Without a quality measure:* At first, the fusion of images is carried out without using quality measure. For each sequence, we create a single image by averaging the pixels' intensities of the different frames of such a sequence. We experimented two cases: with and without interpolated images. Recognition performance of the two methods is reported in Table 2 at the Equal Error Rate (EER) functioning point.

Table 2 shows that the fusion method based on the interpolation of images before averaging the pixels' intensities outperforms the simple average method, with a relative improvement of 25.30 % at the EER functioning point. This result is coherent with Nguyen's results which states that super resolution (SR) greatly improves recognition performance [5].

By observing Tables 1 and 2, we see that the MPMG-min method is slightly better than the simple average (4.66 vs. 4.9 %). These results are coherent with those obtained by Hollingsworth et al. [3]. However, as explained in [3], the matching time and memory requirements are much lower for image's fusion than score's fusion.

- *With a quality measure (global and local):* Given the considerable improvement brought by the interpolation, we decided to perform further experiments only on SR images. We introduce in the fusion the global quality (GQ) and local quality (LQ) fusion schemes as explained in Sect. 2.4. Recognition performance results at the Equal Error Rate (EER) of all methods are reported in Table 3 and the associated DET-curves are shown in Fig. 4.

As shown in Table 3, introducing our global quality criterion in the fusion gives a high relative recognition improvement (25.95 % at the EER) compared to when no quality is considered. Our method is in agreement with Nguyen's result [6] who

Table 2 Equal Error Rate (EER) of the image's fusion methods without using quality

Strategy of fusion	EER (in %)
Simple average of normalized iris	4.90
Simple average of interpolated normalized iris (SR)	3.66

Table 3 Equal Error Rate (EER) of the image's fusion methods with and without quality measures

Strategy of fusion	EER (in %)
Without quality	3.66
With global quality	2.71
With local quality	2.58

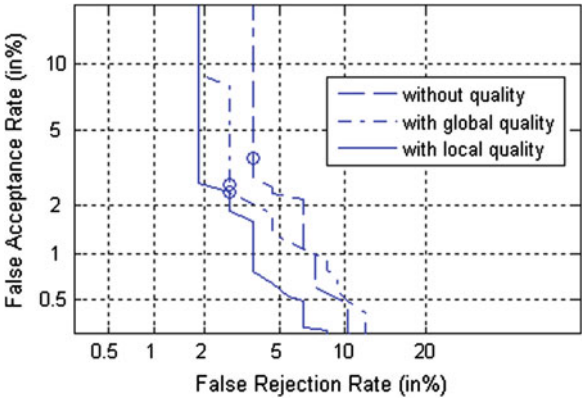


Fig. 4 DET-curves of the three image's fusion approaches

obtains an improvement of 11.5 % by introducing his quality measure (but with another evaluation protocol). Compared to his method, our quality is simpler to implement. Indeed, the metric employed by Nguyen to estimate the quality of a given frame includes four independent factors: focus, off-angle, illumination variation and motion blur. After calculating individually each of these quality scores, a single score is obtained with the Dempster-Shafer theory [6]. Our quality measure has the advantage of not requiring an extra strategy of combinations neither knowing in advance the possible nature of the degradation.

By incorporating our GQ measure in the fusion process, the contribution of each frame in the fused image will be correlated to its quality, this way more weight is given to the high quality images.

Table 3 also shows that LQ-based fusion method outperforms the GQ-based fusion method with a relative improvement of 4.79 % at the EER. This is due to the fact that the quality in an iris image is not globally identical: indeed, due for example to motion blur, a region in an iris image could be more textured than another one. Moreover, our LQ measure can detect eventual errors of masks and assign them a low value. The LQ-based fusion scheme allows therefore a more accurate weighting of the pixels in the fusion scheme than the GQ-based method.

3.2 CASIA-Iris-Thousand V4

Database and Protocol. Experiments are carried out on a subset of the challenging CASIA-Iris-Thousand database [16]. The complete database includes 20,000 iris images from 2,000 eyes of 1,000 persons. Thus, each subject has 10 instances of both left and right eye. The images are captured using the dual-eye iris camera using IKEMB-100 produced by IrisKing. In this work, we select randomly 600 subjects.

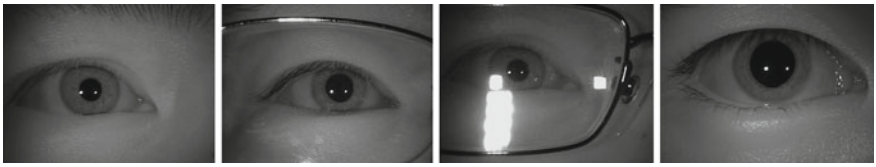


Fig. 5 Examples of images taken from CASIA-Iris-Thousand v4

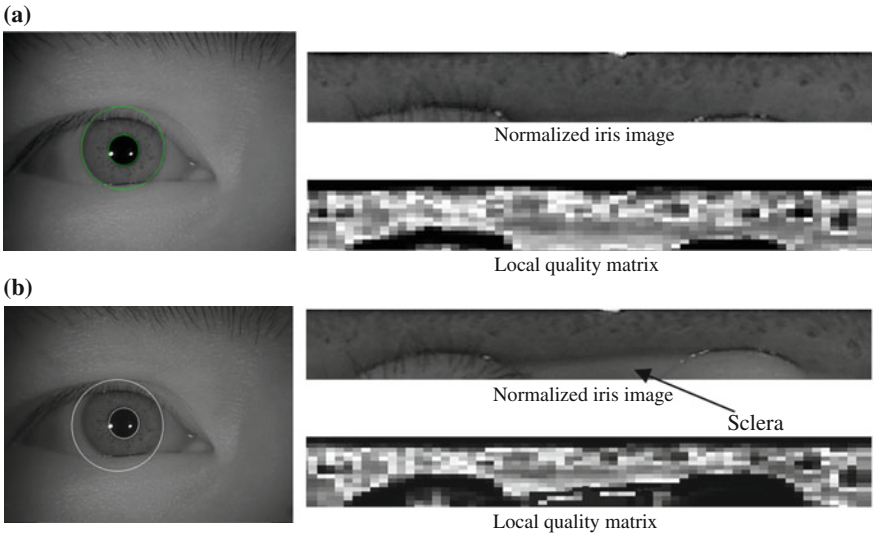


Fig. 6 Examples of two different segmentations on the same eye: **a** good segmentation and corresponding normalization, **b** wrong segmentation and corresponding normalization

The main sources of variations in this database are eyeglasses, specular reflections and dilatation (see Fig. 5), which make the iris segmentation particularly difficult.

To perform the segmentation step, we used the OSIRISv2 reference system described in Sect. 2.1. As mentioned before, the segmentation is based on Hough transform and active contour. This version has been improved in the latest version but, as our purpose is to show the impact of using a local quality measure in the fusion process when the segmentation of the iris fails, we decided to use the second version, which leads to more cases of wrong segmentation. Figure 6 illustrates two examples of such segmentation: a good (Fig. 6a) and a wrong (Fig. 6b) segmentation and their corresponding normalized images and local quality matrix.

On Fig. 6b, we can see a part of the sclera on the normalized iris image. This is due to the wrong localization of the boundary between the iris and the sclera. When fusing the iris images, this zone which does not correspond to iris texture will degrade the results. Therefore, we should give to this zone less importance in the process of fusion. To do this, we will fuse the normalized iris images using their local quality matrices. This way, dark areas of this matrix, which correspond to low quality zones

(zones not belonging to the iris region), will therefore contribute less in the fusion than bright ones.

Apart from this undesirable effect, a bad localization of the borders of the irises also introduces some disparities in the normalized iris images and therefore two images from the same eye can lead to two normalized images in which two points with the same coordinates do not correspond to the same texture. Note that our actual implementation does not take this effect into account.

Experiments and Results. For each subject, we divided arbitrary the 10 instances into two equal parts: we used 5 images as a target sequence (references) and the rest as a query sequence (test). On CASIA v4 database, we performed several experiments:

- As done on MBGC database (Table 1), we analyzed on CASIA v4 database the impact of score fusion methods on recognition performance following the three previously defined protocols: Matching 1 to 1, Matching N to 1, Matching M to N. We report in Table 4, the recognition performance at the EER.
- We also analyzed the impact of image fusion methods considering only the case of interpolated images (super resolution-based fusion). As in CASIA v4 database there are several test images per person (5 images), we fused reference images (as done on MBGC database) and also test images. Recognition performance is computed by comparing the obtained fused test image to the obtained fused reference image. This protocol is close to the MGMP one: in both scenarios, we assume dependency between the reference images and the test images. Recognition performance results are reported in Table 5.

We first observe in Table 4, as expected, that the results are highly improved thanks to the fusion of the scores. A relative improvement of 56.16 % is observed considering the scenario 5 to 1 and of 73.56 % considering the scenario 5 to 5, compared to the scenario 1 to 1.

When comparing the results reported in Tables 4 and 5, we notice that all the images fusion methods outperform the best obtained score fusion (EER = 6.29 %).

Table 4 Equal Error Rate (EER) values of the score's fusion methods on CASIA v4 database

Score's fusion methods	EER (in %)
Matching 1 to 1	23.82
Matching 5 to 1 (MGSP)	9.98
Matching 5 to 5 (MGMP)	6.29

Table 5 Equal Error Rate (EER) values of the image fusion methods. The fusion is carried out on the reference and test images

Images fusion methods: fusing both reference and test images	EER (in %)
Without quality	5.48
With global quality	4.86
With local quality	5.54

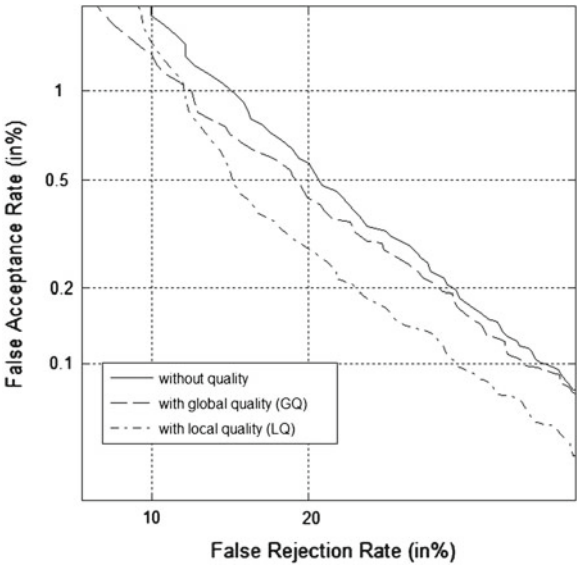


Fig. 7 DET-*curves* of the three image's fusion approaches (scenario 5–5)

This interesting result points out the contribution of the fusion at the image level compared to the fusion at the score level that is high time consuming.

We also notice in Table 5 that the best performance is obtained when the global quality is considered in the images fusion (EER = 4.86 %), while the result at the EER functioning point with the local quality is the worst one. To refine this analysis, we plotted in Fig. 7 the DET curve of the different image fusion methods. We observe that at FAR values lower than 1 %, the local quality-based system leads to a significant improvement in terms of recognition performance compared to the global quality-based system and to when no quality is considered. Indeed at FAR = 0.1 %, FRR = 33.3 % with the local quality measure while FRR = 39.5 % with the global quality measure and FRR = 41.2 % when no quality is considered. This is an interesting result, as low FAR values are often considered in the iris recognition framework.

4 Conclusions

In this paper, we have proposed novel contributions to the problem of iris performance decrease due to some degradation of the iris image. We considered only the case where we have at disposal several shots of the same eye for each person. We have tackled two different situations, namely video sequences of low resolution resulting from an acquisition at a distance (MBGC database) and sequences of multi-shot still images, presenting variability and therefore segmentation defaults (CASIA v4

database). Our approach is based on simple super-resolution techniques improved by taking into account some quality criteria. Our main novelty is the introduction in the fusion scheme, at the pixel level, of a local quality (LQ) measure relying on a GMM estimation of the distribution of a clean iris texture. This LQ measure can also be used to compute a global quality (GQ) measure of the normalized iris image. We have shown on the MBGC database that the LQ-based fusion allows a high improvement in performance compared to other fusion schemes (at the score or image level) or to our GQ-based fusion. The experiments on the CASIA v4 still images database also show a big improvement thanks to the use of image fusion for the references and test sets. While the LQ-based image fusion does not bring any improvement at the EER functioning point compared to the global quality schemes, it is very efficient at low FAR values.

The present work is a first step towards the introduction of super resolution techniques in the context of low quality image sequences. Our first results need to be confirmed by extensive experiments. From a practical point of view, processing videos would require a completely automatic system, which was not implemented in the present work. Indeed, we have manually chosen the adequate images in the video and segmented them manually. The results obtained with the manual segmentation therefore allowed us to conclude on the positive effect of our approach on low resolution images (independently of segmentation errors). An automatic segmentation procedure can replace the manual one but, due to the low quality of MBGC frames, we expect that it will produce a large number of errors (as assessed by the degradation of performance observed in the MBGC competition). Anyhow, the good results that we obtained by fusing the sequences of still images from CASIA v4 (which are badly segmented with OSIRISv2) can make us optimistic in the global performance of an automatic system for processing low resolution iris videos in Near Infra Red.

References

1. MBGC portal challenge version 2 preliminary results. http://www.nist.gov/itl/iad/ig/mbgc-presentations.cfm
2. IREXIII: http://www.nist.gov/itl/iad/ig/irexiii.cfm
3. Hollingsworth, K., Peters, T., Bowyer, K.W., Flynn, P.J.: Iris recognition using signal-level fusion of frames from video. In: IEEE Transactions on Information Forensics and Security, vol. 4, no. 4, pp. 837–848 (2009)
4. Fahmy, G.: Super-resolution construction of IRIS images from a visual low resolution face video. In: Proceedings of the International Symposium on Signal Processing and Its Applications (2007)
5. Nguyen, K., Fookes, C.B., Sridharan, S., Denman, S.: Focus-score weighted super-resolution for uncooperative iris recognition at a distance and on the move. In: Proceedings of the International Conference of Image and Vision Computing (2010)
6. Nguyen, K., Fookes, C.B., Sridharan, S., Denman, S.: Quality-driven super-resolution for less constrained iris recognition at a distance and on the move. In: IEEE Transactions on Information Forensics and Security, vol. 6, no. 4, pp. 1248–1258 (2011)
7. Nguyen, K., Fookes, C.B., Sridharan, S., Denman, S.: Feature-domain super-resolution for iris recognition. In: Proceedings of the International Conference on Image Processing (2011)

8. Jillela, R., Ross, A., Flynn, P.J.: Information fusion in low-resolution iris videos using principal components transform. In: Proceedings of the 2011 IEEE Workshop on Applications of Computer Vision, pp. 262–269 (2011)
9. Cremer, S., Dorizzi, B., Garcia-Salicetti S., Lempiérière, N.: How a local quality measure can help improving iris recognition. In: Proceedings of the International Conference of the Biometrics Special Interest Group (2012)
10. Krichen, E., Garcia-Salicetti, S., Dorizzi, B.: A new probabilistic iris quality measure for comprehensive noise detection. In: Proceedings of the IEEE International Conference on Biometrics: Theory, Applications, and Systems, pp. 1–6 (2007)
11. Daugman, J.: How iris recognition works. IEEE Trans. Circuits Syst. Video Technol. **14**, 21–30 (2007)
12. BioSecure project: http://biosecure.it-sudparis.eu
13. Sutra, G., Garcia-Salicetti, S., Dorizzi, B.: The viterbi algorithm at different resolutions for enhanced iris segmentation. In: Proceedings of the International Conference on Biometrics, pp. 310–316 (2012)
14. Ma, L., Tan, T., Wang, Y., Zhang, D.: Efficient iris recognition by characterizing key local variations. IEEE Trans. Image Process. **13**, 739–750 (2004)
15. Krichen, E., Allano, L., Garcia-Salicetti, S., Dorizzi, B.: Specific texture analysis for iris recognition. Audio- and video-based biometric person authentication, pp. 23–30. Springer, Berlin (2005)
16. http://www.idealtest.org/findTotalDbByMode.do?mode=Iris

Generic Biometry Algorithm Based on Signal Morphology Information: Application in the Electrocardiogram Signal

Tiago Araújo, Neuza Nunes, Hugo Gamboa and Ana Fred

Abstract This work presents the development, test, and implementation of a new biometric identification procedure based on electrocardiogram (ECG) signal morphology. ECG data were collected from 63 subjects during two data-recording sessions separated by six months (Time Instance 1, T1, and Time Instance 2, T2). Two tests were performed aiming at subject identification, using a distance-based method with the heartbeat patterns. In both tests, the enrollment template was composed by the averaging of all the T1 waves for each subject. Two testing datasets were created with five meanwaves per subject. While in the first test the meanwaves were composed with different T1 waves, in the second test T2 waves were used. The T2 waves belonged to the same subjects but were acquired in different time instances, simulating a real biometric identification problem. The classification was performed through the implementation of a kNN classifier, using the meanwave's Euclidean distances as the features for subject identification. The accuracy achieved was 95.2 % for the first test and 90.5 % for the second. These results were achieved with the optimization of some crucial parameters. In this work we determine the influence of those parameters, such as, the removal of signal outliers and the number of waves that compose the test meanwaves, in the overall algorithm performance. In a real time identification problem, this last parameter is related with the length of ECG signal needed to perform an accurate decision. Concerning the study here depicted, we

T. Araújo (✉) · N. Nunes · H. Gamboa · A. Fred
CEFITEC, New University of Lisbon, Caparica, Portugal
e-mail: taraujo87@gmail.com

T. Araújo · N. Nunes · H. Gamboa · A. Fred
Plux - Wireless Biosignals, Lisbon, Portugal
e-mail: nnunes@plux.info

T. Araújo · N. Nunes · H. Gamboa · A. Fred
Instituto de Telecomunicações, Scientific Area of Networks and Multimedia,
Lisbon, Portugal
e-mail: hgamboa@fct.unl.pt

A. Fred
e-mail: afred@lx.it.pt

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_19

conclude that a distance-based method using the subject's ECG signal morphology is a valid parameter for classification in biometric applications.

Keywords Biometry · Classification · Electrocardiography · Meanwave · Signal processing

1 Introduction

Every day, large amounts of confidential data are stored and transferred through the internet. New concerns about security and authentication are arising; speed and efficiency in intruders detection is crucial. Biometric recognition addresses this problem in a very promising point of view. The human, voice, fingerprint, face, and iris are examples of individual characteristics currently used in biometric recognition systems [1]. Recently, several works have studied the electrocardiography (ECG) signal as an intrinsic subject parameter, exploring its potential as a human identification tool [2–4].

Biometry based in ECG is essentially done by the detection of fiducial points and subsequent feature extraction (Fig. 1) [5]. Nevertheless there are some works that use a classification approach without fiducial points detection [6], referring computational advantages, better identification performance and peak synchronization independence.

Since 2007, Institute of Telecommunications (IT) research group has explored this theme addressing it, essentially, in two ways: (i) analysis of the ECG time persistent information, with possible applicability in biometrics over time; and (ii) development of acquisition methods which enabled the ECG signal acquisition with less obtrusive setups, particularly using hands as signal acquisition point.

Following these goals, a recent work proposed a finger-based ECG biometric system, collecting the signals through a minimally intrusive 1-lead ECG setup at the fingers and recurring to Ag/AgCl electrodes without gel [5]. In the same work, an algorithm was developed for comparison between the R peak amplitude from the

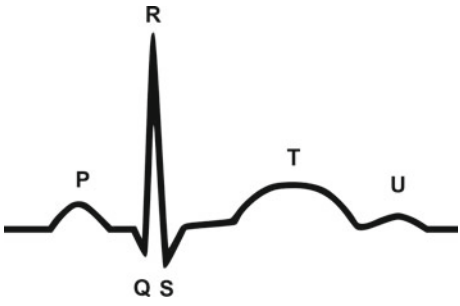


Fig. 1 The ecg fiducial points

heartbeats of test patterns and the R peak from the enrollment template database. The results revealed that this could be a promising technique.

In this work we used the IT ECG database and follow the same methodology as described before, but using a new biometrics classification algorithm based on the heartbeat meanwave's Euclidean distances.

In the following section we will depict the procedure for the ECG data acquisition and pre-processing. We also explain the methodology followed in this study to efficiently classify the heartbeat waves into the respective subject. The results of the classification procedure are exposed and discussed in section three. Conclusions are taken in section four of this paper.

2 Procedure

2.1 Data Collection

ECG data were collected from 63 subjects, 166.55 ± 8.26 cm, 61.82 ± 11.7 Kg and 21 ± 4.46 years old, during two data-recording sessions with six months between them. We divided those acquisitions in two groups, T1 and T2, referring respectively to the first recording instance and the second recording six months after. The subjects were asked to be seated and relaxed in both recordings.

2.2 Signal Acquisition and Conditioning

The signals were acquired by two dried electrodes assembled in a differential configuration [5]. The sensor uses a virtual ground, an input impedance over 1 MΩ, 110 dB of CMRR and gain of 10 in the first stage. The conditioning circuit consists of two filtering levels: (i) bandpass between 0.05 and 1,000 Hz and (ii) notch filter centered in 50 Hz to remove network interference. The final amplification stage has a gain of 100 to improve the resolution of the acquired signal. This system also magnifies the signal after filtering undesired frequencies in each conditioning stage. The signal is then digitalized for further digital processing. This processing consists in: (a) bandpass digital filter (FIR) of 301 order and bandwidth from 5 to 20 Hz, obtained using a hamming window, (b) detection of QRS complexes, (c) segmentation of ECG and determination RR intervals, (d) outliers removal, (e) meanwave computation and feature extraction, and finally (f) the data classification. The signal acquisition and the processing steps (a), (b) and (c) were done by the methodology developed in IT [5].

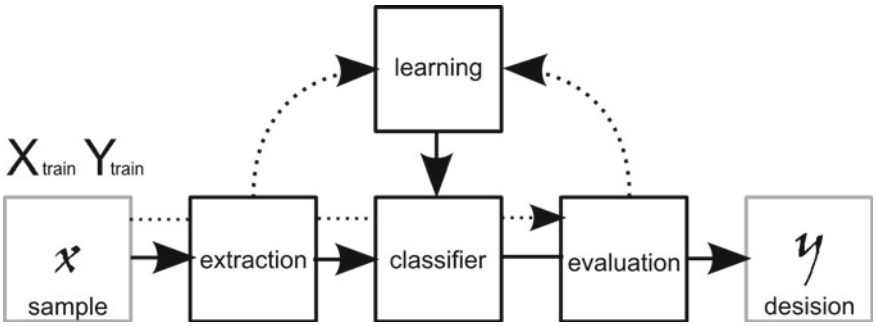


Fig. 2 The process of data classification

In the following section the methodology designed for the implementation of the remaining steps ((d), (e) and (f)) will be described.

2.3 The Process of Data Classification

Data classification is a machine learning technique used to predict group membership for data instances. The main goal of this study was to successfully use the patterns of ECG heartbeats to make subject's identification in different time periods, using a classification method.

Figure 2 depicts the usual process that is followed to classify a set of data.
This process comprises a first stage of feature extraction, making data transformations to generate useful and novel features from a set of candidates. In the data classification there's a supervised learning process.
A first set of data, called training set, is received as input by the classifier, then, with those inputs, it will learn about the features and correspondent classes. The new set of data given, called test set, will match the features with the input training set and associate each sample to the correspondent classes.

2.4 Feature Extraction

The Fig. 3 provides a schematics of the methodology followed in this work.

The data used in this study were divided in two groups: the T1 and T2 acquisitions. In the first test we work with only T1 waves, and in the second test we compare the T2 waves with the T1 template—therefore we can check the differences in classification accuracy when working with acquisitions separated in time from the same subject, simulating a real biometric identification problem.

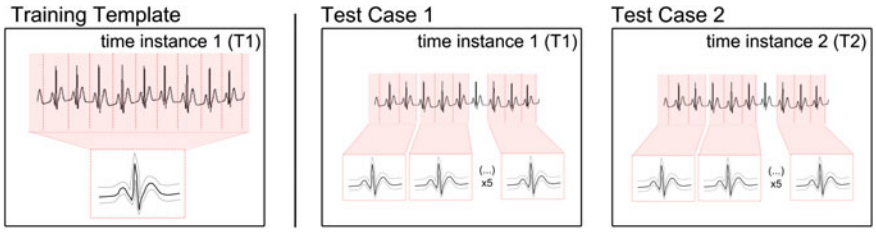


Fig. 3 Template and Tests of the classification process

The dataset defined as template is composed with the T1 subjects' meanwaves, and the features computed for the classification process will be the distance value between the template meanwaves and the meanwaves of future acquisitions (tests).

To compose the template, the first step was to compute a meanwave [7] by the averaging of all T1 waves (which were already segmented into RR-aligned heartbeats). An outliers removal procedure followed, by computing the mean square error distance of each wave to the resulting meanwave. Equation 1 displays the expression for the computation of this distance for only one heartbeat (being l the length, in samples, of the normalized cycle and meanwave). After gathering the distance of each wave to the meanwave, the mean distance value was computed and the waves which presented a distance value higher than two times the mean were removed from the template.

A new meanwave for each subject was then computed without the outliers. Each subject's meanwave was composed with 100 heartbeat waves. This completed the template for the classifier.

$$distance = \sqrt{\frac{\sum_{i=1}^{l}(cycle_i - meanwave_i)^2}{l}} \tag{1}$$

For the first Test dataset, we also used the T1 waves, but divided them randomly into 5 groups, computing one meanwave for each group. Each meanwave was composed with 20 heartbeat waves. Those five test meanwaves were compared, using a distance metric, with the T1 template, for each subject. The distance metric used was the same presented before in Eq. 1, where we used the meanwaves computed from each group instead of each subject's cycle.

For the second Test we followed the same procedure as before but with a calculation of the distance between the T1 template meanwave and the 5 meanwaves from T2 for each subject.

With the distance values computed for both tests we composed two distances' matrices with 63 columns or features, representing the distance of each sample (the Test meanwaves) to each subject's meanwave of the template T1, and 315 (5×63) rows or samples, representing the 5 meanwaves we gathered for each subject and each Test.

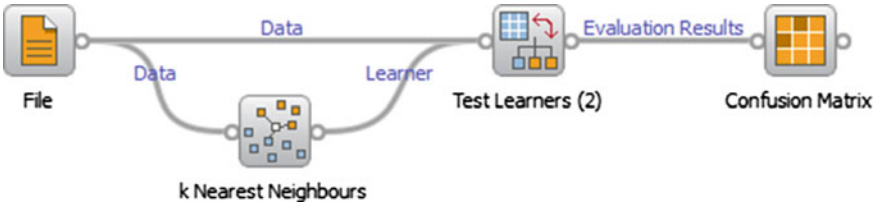


Fig. 4 Schematics used in Orange for classification

2.5 Classifier

To classify the data, a user friendly toolbox [8] was used. As input, it received the distances matrices and used a k-Nearest Neighbor (kNN) classifier with a "leave one out" criterion to learn about the data given. Figure 4 shows the Orange schematics of the data classification and results gathering.

In this image the icons represent the steps of the data classification process: The File icon represents the distance matrices given as input to be classified; The k Nearest Neighbor classifies samples based on the closest class amongst its k nearest neighbors (we used $k = 5$); The test learner represents the stage where the data given is processed by the classification algorithm and the classifier learns about the samples and correspondent classes; The confusion matrix confronts the predictions with the expected results to return the detailed results of the specified classifier.

3 Results and Discussion

3.1 Distance Matrix

Figure 5 shows the distances matrices given as input to the classifier for Test 1 and Test 2 in a color scale image.

The darker colors represent minimum distance values, which are associated to the heartbeat intra-subject distances. For both tests five samples per subject were compared with the meanwave template. Therefore, it is expected to see a diagonal composed with 5 dark cells and all the other cells with lighter colors (in the best case scenario, they would be totally white). As we can see in Fig. 5, the test 1 is closer to the ideal result, as this test comprises waves from the same acquisition both in template and test sets. In the second test the subjects are not so easily visually identified by the distance metric, and therefore it is expected to see a decrease in accuracy for the second test (Table 1).

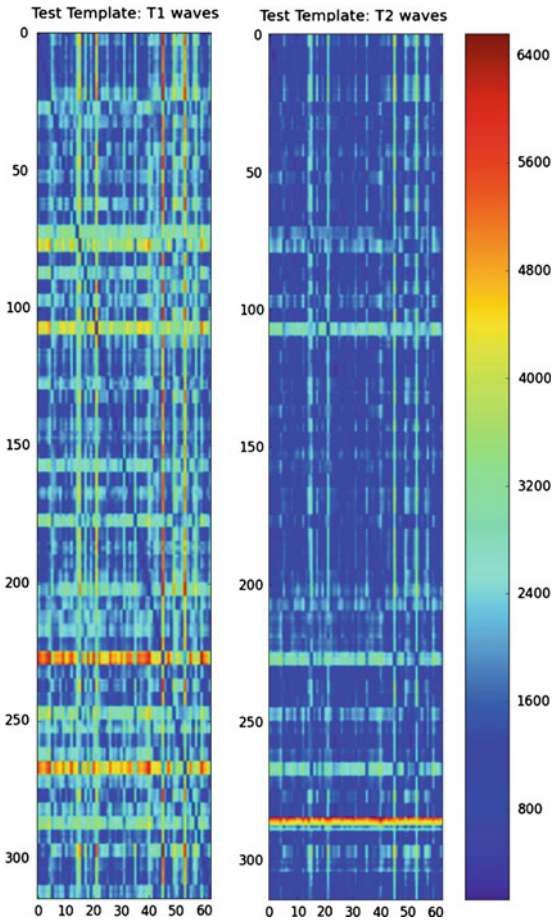


Fig. 5 Distance matrices for Test 1 and Test 2 given as input to the classifier

3.2 Classification Accuracy

After the learning process in Orange, a confusion matrix returned the depicted results of the classifier. An example of that matrix is shown in Table 2.

This matrix gathers the results of the classification for each class (each subject). The ideal case was to have a diagonal always with 5 samples—it represents that all samples were efficiently classified, as we had 5 samples per subject. A cell presenting an inferior value represents that at least one misclassification was made, associating a sample to other class (at least one heatbeat's meanwave was classified as belonging to a different subject).

The final classification results for test 1 and 2, concerning all subjects are included in Table 1.

Table 1 Results for the classification accuracy

	Test 1	Test 2
Accuracy	95.2 %	90.5 %

Table 2 Part of the confusion matrix returned from the classifier

	1	2	3	4	5	6	7	8	9	10	(...)	60	61	62	63
1	**5**	0	0	0	0	0	0	0	0	0	...	0	0	0	0
2	0	**5**	0	0	0	0	0	0	0	0	...	0	0	0	0
3	0	0	**5**	0	0	0	0	0	0	0	...	0	0	0	0
4	0	0	0	**5**	0	0	0	0	0	0	...	0	0	0	0
5	0	0	0	0	**4**	0	0	0	0	**1**	...	0	0	0	0
6	0	0	0	0	0	**5**	0	0	0	0	...	0	0	0	0
7	0	**2**	0	0	0	0	**3**	0	0	0	...	0	0	0	0
8	0	0	0	0	0	0	0	**5**	0	0	...	0	0	0	0
9	0	0	0	0	0	0	0	0	**5**	0	...	0	0	0	0
10	0	0	0	0	0	0	0	0	0	**5**	...	0	0	0	0
(...)	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
60	0	0	0	0	0	0	0	0	0	0	...	**5**	0	0	0
61	0	0	0	0	0	0	0	0	0	0	...	0	**5**	0	0
62	0	0	0	0	0	0	0	0	0	0	...	0	0	**5**	0
63	0	0	0	0	0	0	0	0	0	0	...	0	0	0	**5**

Table 3 Classification accuracy results for Test 1 and Test 2 with and without removal of outliers

	Test 1 (%)	Test 2 (%)
w/ outliers removal	95.2	90.5
w/o outliers removal	88.2	85.4

3.3 Algorithm Parameterization Versus Classification Accuracy

The methodology followed to achieve the depicted results was designed to optimize the classification rate. Before gathering the meanwaves for each subject, an outlier removal algorithm was applied to remove waves which were distant from the template wave. The outliers removal algorithm is relevant to the classification process, as seen in the accuracy rates shown in Table 3. The classification accuracy increases by 2 % and 5 % after removal of the outlier heartbeat waves.

Also stated in the methodology of this work, each of the test sample meanwaves were composed with 20 heartbeat waves from each subject. This was the optimal number of waves to achieve the higher classification rate, as shown in Fig. 6.

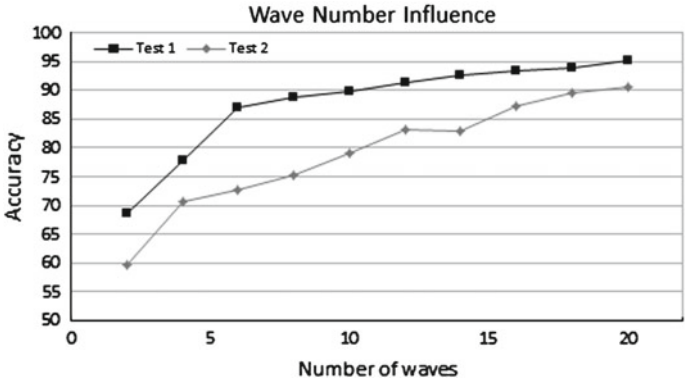


Fig. 6 Influence of the number of waves composing the test sample meanwaves

4 Conclusions

A new biometric classification procedure based on electrocardiogram (ECG) heartbeats meanwave's distances was implemented and depicted in this study. Our goal was to successfully use the patterns of ECG heartbeats to make subject's identification. In order to validate the developed solutions, the methods were tested in a real ECG database. The database was composed by two finger-based ECG acquisitions from 63 subjects. The acquisitions from each subject were separated by six months between them. This fact enabled the evaluation of the algorithm accuracy in a test case scenario, where the test and enrollment template belonged to the first acquisitions, and a real case scenario where we used the first acquisitions as the enrollment template and the second one as test. Using our approach it was possible to obtain accuracy rates of 95.2 % for the test scenario (Test 1) and 90.5 % for the real case scenario (Test 2). Compared with a previous state-of-the-art approach, the results outperform the recent studies on finger-ECG based identifications. Previous works present 89 % [9] and 94.4 % [5] of accuracy.

Future work will be focused on improving the feature extraction process and add features to the classifier, such as the correlation between waves or the intra-subject variability—as we noticed that some subjects had an higher variability in their meanwaves, and therefore the distance computed isn't the best feature per se.

Acknowledgments The authors would like to thank the Escola Superior de Saúde-Cruz Vermelha Portuguesa (ESSCVP) for the data collections infrastructures and subjects providence.

References

1. Jain, A., Hong, L., Pankanti, S.: Biometric identification. Commun. ACM. **42**(2), 90–98 (2000)
2. Silva, H., Gamboa, H., Fred, A.: Applicability of lead v2 ecg measurements in biometrics. In: Proceedings of Med-e-Tel (2007)
3. Coutinho, D. P., Fred, A. L. N., Figueiredo, M. A. T.: Personal identification and authentication based on one-lead ecg using ziv-merhav cross parsing. In: 10th International Workshop on Pattern Recognition in Information Systems (2010)
4. Li, M., Narayanan, S.: Robust ecg biometrics by fusing temporal and cepstral information. In: 20th International Conference on Pattern Recognition (2010)
5. Lourenco, A., Silva, H., Fred, A.: Unveiling the biometric potential of finger-based ecg signals. In: Computational Intelligence and Neuroscience (2011)
6. Plataniotis, K.N., Hatzinakos, D., Lee, J.K.M.: Ecg biometric recognition without fiducial detection. In: Biometric Consortium Conference, Biometrics Symposium (2006)
7. Nunes, N., Araujo, T., Gamboa, H.: Time series clustering algorithm for two-modes cyclic biosignals. In: Fred, A., Filipe, J., Gamboa, H. (eds.) BIOSTEC 2011, CCIS 273, pp. 233–245. Springer, Heidelberg (2012)
8. Orange. http://orange.biolab.si/ (2012)
9. Chan, A.D.C., Hamdy, M.M., Badre, A., Badee, V.: Wavelet distance measure for person identification using electrocardiograms in IEEE Transactions on Instrumentation and Measurement (2008)

Erratum to: A MAP Approach to Evidence Accumulation Clustering

André Lourenço, Samuel Rota Bulò, Nicola Rebagliati, Ana Fred, Mário Figueiredo and Marcello Pelillo

Erratum to:
Chapter 'A MAP Approach to Evidence Accumulation Clustering' in: A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*, DOI 10.1007/978-3-319-12610-4_6

The authors name and their affiliations in 'A MAP Approach to Evidence Accumulation Clustering' should be displayed in first page as shown below:

The online version of the original chapter can be found under DOI 10.1007/978-3-319-12610-4_6

A. Lourenço (✉)
Instituto Superior de Engenharia de Lisboa, Instituto de Telecomunicações, Lisbon, Portugal
e-mail: alourenco@deetc.isel.ipl.pt

A. Fred · M. Figueiredo
Instituto de Telecomunicações, Instituto Superior Técnico, Lisbon, Portugal
e-mail: afred@lx.it.pt

M. Figueiredo
e-mail: mtf@lx.it.pt

S. Rota Bulò
Fondazione Bruno Kessler, Trento, Italy
e-mail: rotabulo@fbk.eu

M. Pelillo
DAIS, Università Ca' Foscari Venezia, Venice, Italy
e-mail: pelillo@dsi.unive.it

N. Rebagliati
VTT, Espoo, Finland
e-mail: nicola.rebagliati@gmail.com

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4_20

A. Lourenço
Instituto Superior de Engenharia de Lisboa, Instituto de Telecomunicações, Lisbon, Portugal
e-mail: alourenco@deetc.isel.ipl.pt

A. Fred · M. Figueiredo
Instituto de Telecomunicações, Instituto Superior Técnico, Lisbon, Portugal
e-mail: afred@lx.it.pt

M. Figueiredo
e-mail: mtf@lx.it.pt

Author Index

A. Fred and M. De Marsico (eds.), *Pattern Recognition Applications and Methods*,
Advances in Intelligent Systems and Computing 318,
DOI 10.1007/978-3-319-12610-4