UNIVERSITÀ DEGLI STUDI DI PARMA
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Dottorato di Ricerca in Tecnologie dell'Informazione
XXVI Ciclo

Pablo Mesejo Santiago

AUTOMATIC SEGMENTATION OF ANATOMICAL STRUCTURES USING DEFORMABLE MODELS AND BIO-INSPIRED/SOFT COMPUTING

DISSERTAZIONE PRESENTATA PER IL CONSEGUIMENTO
DEL TITOLO DI DOTTORE DI RICERCA

Gennaio 2014

UNIVERSITÀ DEGLI STUDI DI PARMA
DOTTORATO DI RICERCA IN TECNOLOGIE DELL'INFORMAZIONE
XXVI Ciclo

AUTOMATIC SEGMENTATION OF ANATOMICAL STRUCTURES USING DEFORMABLE MODELS AND BIO-INSPIRED/SOFT COMPUTING

Coordinatore: Chiar.mo. Prof. Marco Locatelli
Relatore: Chiar.mo. Prof. Stefano Cagnoni

Autore: Pablo Mesejo Santiago

Gennaio 2014

Dedicated to my mother and Sofia,
with gratitude, respect and admiration.

Contents

List of Figures

List of Tables

Acknowledgements

Odio gli indifferenti. Credo che vivere voglia dire essere partigiani.
Chi vive veramente non può non essere cittadino e partigiano.
L'indifferenza è abulia, è parassitismo, è vigliaccheria, non è vita.
Perciò odio gli indifferenti.
Antonio Gramsci, La Città futura, 1917

It was in Spain that [my generation] learned
that one can be right and yet be beaten,
that force can vanquish spirit,
that there are times when courage
is not its own recompense.
Albert Camus

English

First of all, I would like to thank Stefano Cagnoni for his dedication, patience, and constant constructive attitude, as well as for the detailed revision of this Thesis. I want to express my gratitude to him for giving me the opportunity of showing my skills, when choosing me among the more than a hundred candidates that applied for the $MIBISOC$ project, a Marie Curie Initial Training Network of the Seventh Framework Programme (7FP) funded by the European Commission (Grant Agreement: 238819), of which I took part during the last three years. I would also like to thank the Professors who embraced me in the different secondments (Dr. Bai Li in the University of Nottingham and Dr. Sergio Damas in the European Center for Soft Computing),

besides the main organizers and coordinators of the European network in which I could train myself as a researcher, engineer and person, particularly, Óscar Cordón, Sergio Damas, Carmen Peña and Óscar Ibáñez (this one, especially, master and friend). I would like to thank Ottmar Beucher for his generous and useful advices regarding the statistical analysis of the results of soft computing techniques, and Roberto Ugolotti for sharing with me his great talent and for the wonderful job made together. It is also essential to mention here the people whose collaboration has been crucial in some specific parts of this Thesis: Roberto Ugolotti in chapter 5, Davide Valeriani and Alessandro Costalunga for their work in section 6, and Andrea Valsecchi in chapter 7.

Obviously, I also have profound words of thanks to each and every one of the Early Stage Researchers involved in this european network, and especially my fellows in Parma (Youssef, Lara, Pouya, Hamid and Matteo). It has been a pleasure to learn with you during the last three years, and I hope the fact that we have known each other does not become a mere anecdote in our lives.

It was Isaac Newton who wrote, in a letter sent to Robert Hooke (inventor of the term cell) in February 1676, “if I have seen further it is by standing on the shoulders of Giants”. Surely, among others, the giants to whom he was referring were Kepler, Copernicus and Galileo. The origin of such an interesting phrase “standing on the shoulders of Giants” can be found in the twelfth century Neo-Platonist philosopher Bernard of Chartres and, it seems, in other medieval and Renaissance thinkers. I make convenient reference to the expression because, sincerely and humbly, I must admit that, if I have reached something (either in a human, professional or intellectual level) it has been thanks to the privilege and fortune to go, at all times, accompanied by Giants such as my family and friends. I refer, for example, to my grandparents for the sincere affection they have always shown, Sofía’s family for its warm and continuous generosity towards us, Tono and Rosita for being an example and reference in my life, my aunts for believing (fondly but erroneously) that I am “the smartest kid in the world”, all those who have made the effort of visiting us in the different places we have been and, ultimately, to all those people who, over these years, have generously shared their time with us since, do not forget it, time is, undoubtedly, our most precious good.

With respect to my mother and Sofía, there are not words making justice to how much I owe you. Sofía's courage in coming with me to Italy leaving her job, family and friends. My mother's generosity for always giving more than she receives without demanding anything in return and educating me in the values I try to employ every day. Thanks for your patience, for your support and for your affection. This Thesis, with all justice, is dedicated to you.

Italiano

Prima di tutto, vorrei ringraziare Stefano Cagnoni per la sua dedizione, la sua pazienza e il suo costante atteggiamento costruttivo, così come per la revisione dettagliata di questa Tesi e per avermi dato l'opportunità di mostrare le mie capacità, scegliendomi tra un centinaio di candidati che hanno presentato la domanda per il progetto $MIBISOC$, una Marie Curie Initial Training Network del Settimo Programma Quadro (7PQ) finanziato dalla Commissione Europea (Grant Agreement: 238819), a cui ho preso parte nel corso degli ultimi tre anni. Vorrei anche ringraziare i Professori che mi hanno accolto nei diversi periodi di ricerca all'estero (Dr. Bai Li presso l'University of Nottingham e il Dr. Sergio Damas presso l'European Center for Soft Computing), oltre ai principali organizzatori e coordinatori della rete europea nella quale ho potuto formarmi come ricercatore, ingegnere e persona, in particolare, Óscar Cordón, Sergio Damas, Carmen Peña e Óscar Ibáñez (quest'ultimo, soprattutto, maestro e amico). Voglio ringraziare Ottmar Beucher per i suoi consigli generosi e utili per quanto riguarda l'analisi statistica dei risultati di tecniche di soft computing, e Roberto Ugolotti per aver condiviso con me il suo grande talento e per l'ottimo lavoro svolto insieme. E' anche essenziale ricordare qui le persone la cui collaborazione è stata fondamentale in alcune parti specifiche di questa Tesi: Roberto Ugolotti nel capitolo 5, Davide Valeriani e Alessandro Costalunga per il loro lavoro nella sezione 6 e Andrea Valsecchi nel capitolo 7.

Ovviamente, rivolgo profonde parole di ringraziamento anche ad ognuno degli Early Stage Researchers coinvolti in questa rete europea, e soprattutto ai miei compagni a Parma (Youssef, Lara, Pouya, Hamid e Matteo). E' stato un piacere imparare con tutti voi durante gli ultimi tre anni, e spero che il fatto che ci siamo conosciuti, non diventi

un semplice aneddoto nella nostra vita.

Fu Isaac Newton che scrisse, in una lettera inviata a Robert Hooke (inventore del termine cella) nel febbraio 1676 “Se ho visto più lontano è perché stavo sulle spalle di Giganti”. Sicuramente, tra gli altri, i giganti ai quali si riferiva erano Keplero, Copernico e Galileo. L’origine di tale frase interessante “in piedi sulle spalle dei Giganti” si trova nel filosofo neoplatonico del XII secolo Bernard de Chartres e, a quanto pare, in altri pensatori medievali e rinascimentali. Faccio un opportuno riferimento a questa espressione perché, sinceramente e umilmente, devo ammettere che, se ho raggiunto qualcosa (a livello sia umano che professionale o intellettuale) è stato grazie al privilegio e alla fortuna di essere accompagnato, in ogni momento, da Giganti come sono la mia famiglia e i miei amici. Mi riferisco, ad esempio, ai miei nonni per il sincero affetto che hanno sempre dimostrato, alla famiglia di Sofia per la sua affettuosa e costante generosità verso di noi, a Tono e Rosita per essere un esempio e un riferimento nella mia vita, alle mie zie per credere (caramente ma erroneamente) che io sia “il bambino più intelligente del mondo”, a tutti coloro che hanno fatto lo sforzo di farci visita nei diversi luoghi dove siamo stati e, in ultima analisi, a tutte quelle persone che, in questi anni, hanno generosamente condiviso il loro tempo con noi, poiché, non dimenticate che, il tempo è, senza dubbio, il nostro bene più prezioso.

Per quanto riguarda mia madre e Sofía, non ci sono parole che rendano giustizia a quanto devo loro. Il coraggio di Sofía di venire con me in Italia, lasciando il suo lavoro, la famiglia e gli amici. La generosità di mia madre nel dare sempre più di quello che riceve, senza chiedere nulla in cambio, e nell’educarmi ai valori che cerco di impiegare tutti i giorni. Grazie per la pazienza, per il vostro sostegno e per il vostro affetto. Questa Tesi, giustamente, è dedicata a voi.

Galego

En primeiro lugar, gustaríame agradecer a Stefano Cagnoni pola súa dedicación, paciencia, e continua actitude constructiva, así como pola revisión coidadosa desta Tese e por darme a oportunidade de probar a miña valía, ao escollerme entre os mais de cen candidatos que se presentaron ao proxecto $MIBISOC$, unha Rede de Formación Inicial Marie Curie do Sétimo Programa Marco (7PM) financiada pola Comisión

Europea (Grant Agreement: 238819), do que formei parte estes últimos tres anos. Tamén me gustaría agradecer aos profesores que me acolleron nas distintas estadías realizadas (Dr. Bai Li na Universidade de Nottingham e Dr. Sergio Damas no European Center for Soft Computing), así como aos principais organizadores e coordinadores da rede europea na que puiden formarme como investigador, enxeñeiro e persoa, en especial, Oscar Cordón, Sergio Damas, Carmen Peña e Oscar Ibáñez (este último, particularmente, mestre e amigo). Agradecer a Ottmar Beucher os seus xenerosos e útiles consellos no que respecta á análise estatística dos resultados de técnicas de soft computing, e a Roberto Ugolotti por compartir comigo o seu gran talento e polo estupendo traballo realizado xuntos. Resulta tamén fundamental citar aquí as persoas cuxa colaboración resultou decisiva en certas partes concretas desta Tese: Roberto Ugolotti pola súa colaboración no capítulo 5, Davide Valeriani e Alessandro Costalunga polo seu traballo no capítulo 6 e Andrea Valsecchi no capítulo 7.

Obviamente, teño profundas palabras de agradecemento a todos e cada un dos Early Stage Researchers implicados na devandita rede europea, e moi en especial aos meus compañeiros de Parma (Youssef, Lara, Pouya, Hamid e Matteo). Foi un pracer ter a posibilidade de aprender con todos vos durante os últimos tres anos e espero que, o feito de ternos coñecido, non se convirta nunha mera anécdota nas nosas vidas.

Foi Isaac Newton quen escribiu, nunha carta enviada a Robert Hooke (creador do termo célula) en febreiro de 1676, “se vin mais lonxe é porque estou sentado sobre os ombros de Xigantes”. Seguramente, entre outros, os xigantes aos que se refería eran Kepler, Copernico e Galileo. A orixe de tan interesante frase (“a ombros de Xigantes”) encóntrase no filosofo neoplatónico do século XII Bernard de Chartres e, segundo parece, noutros pensadores medievais e renacentistas. Fago conveniente referencia á devandita expresión porque, con total sinceridade, debo recoñecer que, se cheguei a algún sitio (sexa a nivel persoal, profesional ou intelectual), foi grazas ao privilexio e a fortuna de ir, en todo momento, acompañado por uns Xigantes como son a miña familia e amigos. Refírome, por exemplo, aos meus avós polo sincero afecto e o cariño que sempre me mostraron, a familia de Sofía pola súa cálida e continua xenerosidade para connosco, a Tono e Rosita por ser un exemplo e unha referencia para a miña propia

vida, ás miñas tías por crer (cariñosa pero erroneamente) que son “o neno mais listo do mundo”, a todos aqueles que realizaron o esforzo de virnos a visitar nos distintos lugares en que estivemos e, en definitiva, a todas as persoas que, durante estes anos, compartiron xenerosamente o seu tempo connosco (pois, non o esquezamos, o tempo é, sen dúbida, o noso ben mais precioso).

Con respecto á miña nai e Sofía, non hai cualificativos que fagan xustiza ao moito que vos debo. O valor de Sofía ao vir comigo a Italia deixando o seu traballo, familia e amigos. A xenerosidade da miña nai por dar sempre mais do que recibe sen esixir nada a cambio, e por educarme nuns valores que intento aplicar día tras día. Grazas pola vosa paciencia, polo voso apoio e polo voso afecto. Esta Tese vai, con toda xustiza, dedicada a vós.

Español

En primer lugar, me gustaría agradecer a Stefano Cagnoni por su dedicación, paciencia, y continua actitud constructiva, así como por la revisión cuidadosa de esta Tesis y por haberme dado la oportunidad de probar mi valía, al escogerme entre los más de cien candidatos que se presentaron al proyecto europeo $MIBISOC$, una Marie Curie Initial Training Network perteneciente al Séptimo Programa Marco (7PM) financiada por la Comisión Europea (Grant Agreement: 238819), del que formé parte estos últimos tres años. También me gustaría agradecer a los profesores que me acogieron en las distintas estancias realizadas (Dr. Bai Li en la Universidad de Nottingham y Dr. Sergio Damas en el European Center for Soft Computing), así como a los principales organizadores y coordinadores de la red europea en la que he podido formarme como investigador, ingeniero y persona, en especial, Oscar Cordón, Sergio Damas, Carmen Peña y Oscar Ibáñez (éste último, particularmente, maestro y amigo). Agradecer a Ottmar Beucher sus generosos y útiles consejos en lo que respecta al análisis estadístico de los resultados de técnicas de soft computing, y a Roberto Ugolotti por haber compartido conmigo su gran talento y por el estupendo trabajo realizado juntos. Resulta también fundamental citar aquí a las personas cuya colaboración ha resultado decisiva en ciertas partes concretas de esta Tesis: Roberto Ugolotti por su colaboración en el capítulo 5, Davide Valeriani y Alessandro Costalunga por su trabajo en el capítulo 6 y Andrea Valsecchi

en el capítulo 7.

Obviamente, tengo profundas palabras de agradecimiento a todos y cada uno de los Early Stage Researchers implicados en dicha red europea, y muy en especial a mis compañeros de Parma (Youssef, Lara, Pouya, Hamid y Matteo). Ha sido un placer el haber tenido la posibilidad de aprender con todos vosotros durante los últimos tres años y espero que, el hecho de habernos conocido, no se convierta en una mera anécdota en nuestras vidas.

Fue Isaac Newton quién escribió, en una carta enviada a Robert Hooke (creador del término célula) en febrero de 1676, "si he visto más lejos es porque estoy sentado sobre los hombros de Gigantes". Seguramente, entre otros, los gigantes a los que él se refería eran Kepler, Copérnico y Galileo. El origen de tan interesante frase ("a hombros de Gigantes") se encuentra en el filósofo neoplatónico del siglo XII Bernard de Chartres y, según parece, en otros pensadores medievales y renacentistas. Hago conveniente referencia a dicha expresión porque, con total sinceridad, he de reconocer que, si he llegado a algún sitio (sea a nivel personal, profesional o intelectual), ha sido gracias al privilegio y la fortuna de ir, en todo momento, acompañado por unos Gigantes como son mi familia y amigos. Me refiero, por ejemplo, a mis abuelos por el sincero afecto y el cariño que siempre me han mostrado, a la familia de Sofía por su cálida y continua generosidad para con nosotros, a Tono y Rosita por ser un ejemplo y una referencia para mi propia vida, a mis tías por creer (cariñosa pero erróneamente) que soy "el niño más listo del mundo", a todos aquellos que han realizado el esfuerzo de venirnos a visitar en los distintos lugares en que hemos estado y, en definitiva, a todas las personas que, durante estos años, han compartido generosamente su tiempo con nosotros (pues, no lo olvidemos, el tiempo es, sin duda, nuestro bien más precioso).

Con respecto a mi madre y Sofía, no hay calificativos que hagan justicia a lo mucho que os debo. El valor de Sofía al venir conmigo a Italia dejando su trabajo, familia y amigos. La generosidad de mi madre por dar siempre más de lo que recibe sin exigir nada a cambio, y por educarme en unos valores que intento aplicar día tras día. Gracias por vuestra paciencia, por vuestro apoyo y por vuestro afecto. Esta Tesis va, con toda justicia, dedicada a vosotras.

Abstract

This PhD dissertation is focused on the development of algorithms for the automatic segmentation of anatomical structures in biomedical images, usually the hippocampus in histological images from the mouse brain. Such algorithms are based on computer vision techniques and artificial intelligence methods. More precisely, on the one hand, we take advantage of deformable models to segment the anatomical structure under consideration, using prior knowledge from different sources, and to embed the segmentation into an optimization framework. On the other hand, metaheuristics and classifiers can be used to perform the optimization of the target function defined by the shape model (as well as to automatically tune the system parameters), and to refine the results obtained by the segmentation process, respectively. Three new different methods, with their corresponding advantages and disadvantages, are described and tested. A broad theoretical discussion, together with an extensive introduction to the state of the art, has also been included to provide an overview necessary for understanding the developed methods.

Chapter 1

Introduction

All of science is nothing more than the refinement of everyday thinking.
Albert Einstein, Physics and Reality, 1936

Umberto Eco, in [10], considers *scientificity* more as a working model to follow than as a content itself (and, in that case, being only applicable to formal and natural sciences). In particular, he tries to broadly define under what criteria a work can be called scientific, stating that an investigation can be considered scientific if it meets the following requirements:

1. It deals with a recognizable and defined object (*[...] argomento riconoscibile e definito [...]*).

2. It must say things about this subject that have not yet been said, or to examine things that have already been said from a different viewpoint (*[...] dire su questo oggetto cose che non sono già state dette oppure rivedere con un'ottica diversa le cose che sono già state dette [...]*).

3. It must be useful to other people (*[...] deve essere utile agli altri [...]*).

4. It must provide elements for verification and refutation of the hypothesis presented, and therefore has to provide the necessary inputs for its public monitoring

(*[...] deve fornire gli elementi per la verifica e per la falsifica delle ipotesi che presenta e pertanto deve fornire gli elementi per una sua continuazione pubblica [...]*).

A good methodological starting point for this PhD dissertation could be to fill the previous 'template', trying to answer why this research work fulfills each of the mentioned requirements.

With respect to the first one, the *'recognizability'* of the object under investigation, we can affirm that this scientific work deals with **medical image segmentation using soft and bio-inspired computing**. Therefore, the three crucial concepts of this work are: *medical imaging*, *image segmentation* and *soft and bio-inspired computing*. On the one hand, image segmentation is the accurate delineation of an object's boundary within an image or, in other words, the partition of an image into meaningful and non-overlapping regions following some kind of criterion. On the other hand, medical imaging is the production of visual representations of body parts, tissues, or organs, for clinical diagnosis (medical procedures seeking to reveal, diagnose, or examine diseases) or medical science (including the study of normal anatomy and physiology). Finally, soft/bio-inspired computing is an area of artificial intelligence research focused on the design of computational techniques to solve problems (in the presence of uncertain, imprecise and incomplete information) by imitating nature's approaches.

This research is conceived within a broader context: the MIBISOC Marie Initial Training Network ("Medical Imaging using Bio-Inspired and Soft Computing", FP7 PEOPLE-ITN-2008, GA n. 238819). The general area of this european project deals with the application of intelligent systems, constituted by bio-inspired and soft computing techniques, to real-world medical imaging applications. Medical imaging is at the heart of many of today's improved diagnostic and therapeutic technologies, in which computer-based solutions offer the opportunity to obtain quantitative measurement of the medical condition, as well as the pre-processing techniques of filtering, sharpening, and focusing image details, to improve their interpretation by physicians. In fact, bio-inspired and soft computing have already been applied to each of the fundamental steps of medical image processing and analysis with promising results (e.g. restoration,

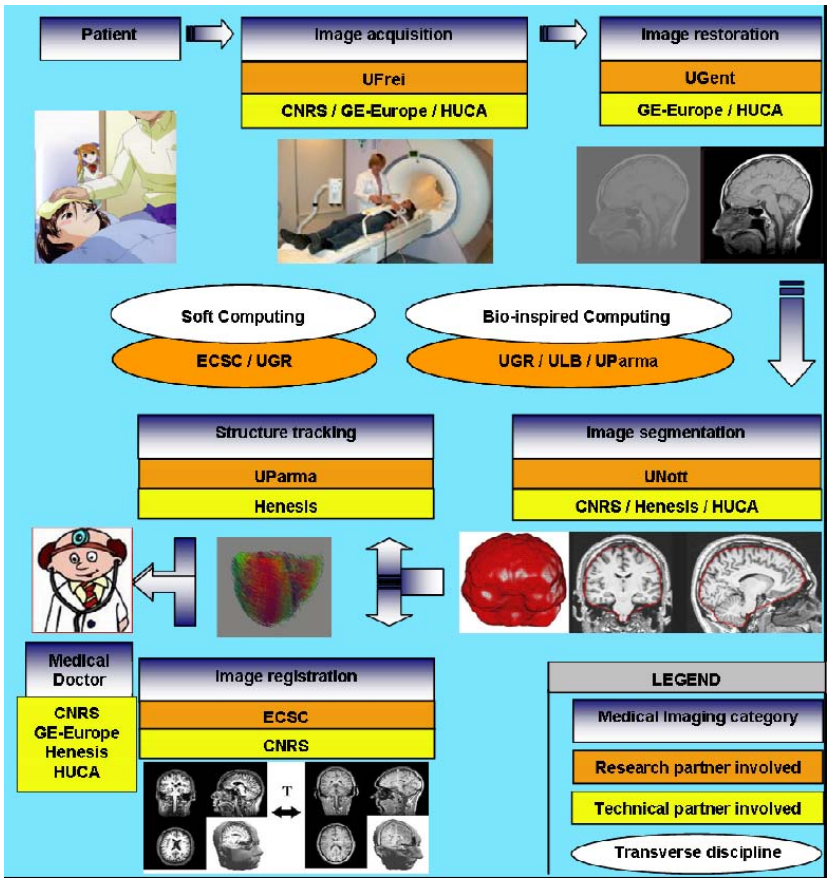


Figure 1.1: MIBISOC partnership [1].

segmentation, registration or tracking). The consortium is composed of world-wide recognized researchers from eight scientific institutions (six Universities, a R&D centre and an enterprise) that are involved as full partners, and four technical partners (a hospital, an enterprise, a medical company and a R&D centre) that provide relevant industrial and medical experience to the Early Stage Researchers (ESRs). Figure 1.1 describes the MIBISOC partnership, including the complete medical imaging pipeline, from the image acquisition process to the physician's final diagnosis [1]. The collaboration of experts from the area of medical imaging with those working on bio-inspired and soft computing applications to computer vision aims at generating new and viable methods and solutions from the combined ideas of the two communities.

In relation to the *'novelty'* of this scientific research, three characteristics should be mentioned:

- this is one of the first, if not the first, systematic and rigorous approach to the hybridization between deformable models (both parametric and geometric) and soft-computing (metaheuristics and classifiers). This PhD dissertation is focused on the segmentation of medical images, as well as on the study of the different possibilites offered by soft computing approaches. This study uses statistical tests and standard quality segmentation metrics extensively, unlike the majority of the work in the field (usually ad-hoc and poorly tested);
- this research represents one of the few cases in which microscopy histological brain images segmentation has been approached using deformable models (both parametric and geometric) and metaheuristics, overcoming in many aspects state-of-the-art methods; and
- finally, this PhD dissertation includes an extensive survey of related work, in the absence of any other 'review' like this in the literature, and introduces three novel automatic segmentation methods, applicable to different medical image modalities, discussing the advantages and disadvantages of their use.

It is important to highlight, with regard to the *'utility'* of improving the existing medical image segmentation systems, the critical role that segmentation algorithms play in image understanding tasks, as they allow one to automatically identify anatomical structures and other regions of interest, or significantly reduce the workload for human operators. Such algorithms are nowadays at the core of multiple tasks, like quantification and measurement of tissue volumes, detection and localization of pathologies or computer-integrated surgery. Furthermore, manual segmentation is not only tedious and time consuming but it can be also inaccurate, hence the importance of developing automatic and accurate segmentation methods.

Finally, with regard to the *'verification and refutation of the hypothesis presented'*, all methods have been extensively compared and statistically tested using standard

segmentation metrics. All algorithms and parameter configurations are described in detail in order to favour repeatability by the scientific community, and the possibility to refute the methods and hypotheses presented here (falsifiability/refutability). It is crucial to stress the importance, and singularity, of this feature since many approaches, in this research field, are only tested using few sample images, introduce extremely ad-hoc methods, do not use standard metrics to evaluate the performance of the algorithms, or do not apply statistical tests to draw accurate conclusions from the results obtained.

This PhD thesis has its origin halfway between engineering and applied research. From a theoretical point of view, the idea is to investigate how hybridizations between soft computing and computer vision techniques can improve methods that are presently used to solve medical imaging problems. From the practical side, this PhD work originates from a joint research project with the Molecular Biotechnology Center of Torino, in which an accurate method to localize the hippocampus in histological images was necessary in order to develop an image analysis pipeline for the identification of synapse-enriched RNAs from in situ hybridization images of the brain.

To end this introduction, it is also necessary to clearly stipulate the conventions followed with respect to language and style.

- This PhD dissertation has been written in English. The only part written in other languages is the acknowledgements chapter, which includes the same text in four different languages: Italian, English, Galician and Spanish.
- All quotations have been included in their original languages, whether in Italian, English, Galician or Spanish. If the original quotation was written in any other different language, such quotation has been translated into English (i.e., English has been the language by default).
- One of the main purposes when writing this PhD dissertation has been to create a self-contained document in which, to understand its content, it would not be necessary to turn to other sources of information. Obviously, this aim can only be fulfilled up to a certain level since, otherwise, there would be the risk of writing an oversized text.

- The content of this dissertation has been mainly extracted from the scientific works published in journals and conferences during the PhD period. An interested reader can also consult such publications (see section 9) in order to deepen his/her knowledge about the methods, approaches and ideas developed and tested.

This PhD thesis is structured in three parts (Fundamentals, Proposed Methods, and Final Remarks) and, in turn, every part is composed by three chapters, except the last one with only two. First, after this introduction, we provide, in chapter 2, the theoretical foundations of this investigation and, in section 3, the image datasets and the kind of medical imaging modalities used in the experiments. In chapter 4, a wide overview of previous related works is introduced. After that, in the second part, three chapters follow with three different approaches to the segmentation of anatomical structures: a parametric approach based on active shape models and ensemble classifiers (chapter 5), a geometric method using eigenshapes (section 6) and, in chapter 7, a level set approach including three different terms (intensity, boundaries and prior shape knowledge) to segment different medical image modalities. Finally, in the last part of this PhD dissertation, a discussion about possible future developments (in chapter 8) and a summary of the conclusions obtained (chapter 9), are followed by four appendices about the scientific papers published during the PhD period, the statistical tests used for analyzing stochastic techniques behaviour, the standard segmentation metrics employed to compare the performance of different segmentation algorithms, and the list of abbreviations used throughout this document, respectively.

Part I: Fundamentals

This section includes three chapters that give an overview of the state-of-the-art of the problem under consideration. First, the theoretical background is introduced, explaining the techniques used in this research and the underlying concepts necessary to understand it. After that, the medical imaging datasets used are presented and explained. Finally, an extensive bibliographical study is presented refering to papers related with this topic.

Chapter 2

Theoretical Background

There is nothing so practical as a good theory.
Kurt Lewin, Problems of research in social psychology, 1951

Instead of his theory being as wide as reality,
his perception of reality may be as narrow as his theory.
Kenneth Craik, The nature of explanation, 1943

2.1 Medical Image Segmentation

Classically, image segmentation (IS) is defined as the partitioning of an image into non-overlapping regions (sets of pixels) that are homogeneous with respect to some visual feature, such as intensity or texture [11]. In order to locate the segmentation process within the broader spectrum of computer vision algorithms, it is important to establish a hierarchy of low-, middle- and high-level tasks. The low-level stage is related to the application of primitive operations, like smoothing, enhancement, and histogram transformations. The middle-level task is focused on image analysis, such as classification, registration and segmentation. Finally, high-level tasks aim to give meaning to the objects recognized in the previous stages and to develop cognitive functions related with the human vision system.

IS algorithms play a crucial role in many medical imaging applications, like the quantification and measurement of tissue volumes, diagnosis, localization of pathologies, and the study of anatomical structures, by automating or facilitating the delineation of such structures and other regions of interest. This task is often challenging because of poor image contrast and frequent artifacts that result in missing or diffuse organ/tissue boundaries. Consequently, a good way of proceeding is to incorporate as much prior information as possible about the problem to be solved, the image acquisition modality or the anatomy, by considering models of the appearance of the structure of interest, based on features such as texture, shape, spatial location of organs, etc. At the same time, it is important to notice that manual segmentation is not only a tedious and time consuming task but, sometimes, it may also be inaccurate, which shows the importance of developing automated methods to accurately carry out such a task.

Using as taxonomic criterion the general principle by which segmentation is based (pixel intensitiy, boundary localization, region detection, or prior knowledge about shape), we could establish the following categories, that can be used in combination with other techniques [12–14]:

- **Thresholding techniques**. A thresholding procedure attempts to determine an intensity value, called the threshold, which separates the image into the desired regions. The segmentation is then achieved by grouping all pixels with greater intensity than the threshold into one class, and all other pixels into another class. These global methods are effective when the intensity levels of the objects of interest fall squarely outside the range of levels in the background. However, since spatial information is ignored, blurred region boundaries can provoke havoc. Image thresholding techniques are well known, and some of the most frequently used methods date from the 70s, as Otsu's method [15], but the problem is not solved yet, and new approaches are required to deal effectively with different signal and image modalities [16].

- **Edge-based methods** are focused on contour detection [17, 18]. These methods make use of various edge operators to calculate the gradient of the image intensity at each point, that provides information about the direction of the largest possible

increase in terms of intensities and its value. The result shows how "abruptly" or "smoothly" image intensity changes at that point, and therefore how likely such a part of the image is to represent an edge, as well as how that edge is likely to be oriented. Difficulties with boundary-based methods occur when objects are overlapping, there is noise or artifacts in the image, or the object has fuzzy boundaries.

- **Region-based methods** [19–21] usually proceed by partitioning the image into connected regions by grouping neighboring pixels with similar features. Adjacent regions are then merged according to some criterion, possibly based on homogeneity or sharpness of region boundaries. Overstringent criteria create fragmentation; lenient ones overlook blurred boundaries and overmerge. However, region-based approaches are generally less sensitive to noise than the boundary-based methods. Many region-based segmentation techniques have been presented in the literature, including region-growing and merging, clustering methods or hierarchical segmentation.

- **Deformable Models** usually start from some initial boundary shape, using prior information about the shape of the object to find, represented in the form of curves, and iteratively modifies it by applying various shrink/expansion operations according to some energy function. These models couple an energy-minimizing approach with the preservation of some "elastic" contour shape. The main risks with such methods are to get trapped into a local minimum and their sensitivity to the initial contour location, which makes the choice of an appropriate model initialization difficult and critical. This category will be described in more detail below.

Anyway, most segmentation techniques are either region-based or edge-based. In general terms, region-based techniques rely on common patterns in intensity values clustering neighboring pixels based on **similarity**. The cluster is referred to as the region, and the goal of the segmentation algorithm is to group regions according to their anatomical or functional roles. Edge-based techniques rely on **discontinuities** in image

intensities between distinct regions, and the goal of the segmentation algorithm is to accurately locate the boundary separating them.

Although the performance of many segmentation techniques is generally good when the contrast-to-noise ratio is high, it decreases rapidly when the structures are insufficiently delineated and have low contrast like the neuroanatomic structures, such as the thalamus or the putamen. Hence, it is really important to upgrade the existing methods in order to improve their accuracy, robustness, or execution time. Also, it is important to notice that most of the IS problems can be posed as optimization problems where the desired segmentation minimizes some energy or cost function defined by the particular application. This possibility is one of the fundamental reasons for hybridizing soft computing technologies with IS algorithms.

2.2 Deformable Models

The term "deformable models" was first used in the late eighties [22, 23] with reference to curves or surfaces, defined within the image domain, that are deformed under the influence of "internal" forces, related with the curve features, and "external" forces, related with the image regions surrounding the curve. Internal forces enforce regularity constraints and keep the model smooth during deformation, while external forces are defined such that the model is attracted toward an object or other features of interest within the image.

There are basically two types of Deformable Models (DMs): **parametric/explicit** and **geometric/implicit** [24]. The former represents curves and surfaces explicitly in their parametric forms during deformation, allowing direct interaction with the model and leading to a compact representation for fast real-time implementation. The latter can handle topological changes naturally, since these models are based on the theory of curve evolution and the level set method, representing curves and surfaces implicitly as a level set of a higher-dimensional scalar function.

Despite the fundamental differences above explained, the underlying principles of

both approaches are very similar. In fact, in spite of the great similarities between the models, it is interesting to note the confusing terminology used. There are multiple terms to refer to practically the same concepts, distinguished in many cases by minor details: Deformable Models [22], Deformable Templates [25, 26], Active Shape Models [27], Active Contour Models/Deformable Contours/Snakes [28], Deformable Surfaces [29–31], Active Appearence Models [32], Statistical Shape Models [33],... In the next two sections some of the best known DMs, as well as the key nuances that differentiate them, will be reviewed.

Parametric Deformable Models

One of the first practical examples, called "snakes" or Active Contour Models (ACMs), was presented shortly after the seminal works of Terzopoulos in [28]. An ACM is a variational method for detecting object boundaries in images. Given n points $C^0 = \{p_1^0, \cdots, p_n^0\}$ that define the initial closed contour, such contour is deformed to lie along the object boundary. The initial contour C^0 is deformed to $C^1 = \{p_1^1, \cdots, p_n^1\}$ by minimizing a certain energy function. Let $X(p)$ be a parameterization of contour C and I be a image intensity. Then the energy is

$$E(C) = \alpha \int |X'(p)|^2 dp + \beta \int |X''(p)|^2 dp - \lambda \int |\nabla I(X(p))| dp \qquad (2.1)$$

The first two terms represent the internal energy while the last term is the external energy. The internal energy is responsible for smoothness, while the external energy is responsible for attracting the contour toward the object boundary, and α (elasticity), β (rigidity) and λ are the free parameters of the system and are determined a priori. Smaller values for λ reduce the noise but can not capture sharp corners while larger values can effectively locate the boundaries but being more sensitive to the noise. Also, α and β prevent the snake from becoming non-continuous or breaking during the iteration process of the optimization problem. The total energy can be written as

$$E(C) = E_{internal}(C) + \lambda E_{external}(C) \qquad (2.2)$$

From the *calculus of variations*, it can be shown that the contour should satisfy the Euler-Lagrange equation:

$$-\frac{d}{dp}(\alpha X') + \frac{d^2}{dp^2}(\beta X'') - \lambda \Delta I(X) = 0. \tag{2.3}$$

Active Shape Models (ASMs) [34] add more prior knowledge to DMs. These shape models derive a "point distribution model" from sets of labelled points (landmarks) selected by an expert within a training set of images: in each image, a point, or set of points, is placed on the part of the object corresponding to its label. The model considers the points' average positions and the main modes of variation found in the training set. While this kind of model has problems with unexpected shapes, since an instance of the model can only take into account deformations which appear in the training set, it is robust with respect to noise and image artifacts, like missing or damaged parts.

Active Appearance Models (AAMs) [32] extend ASMs by considering not only the shape of the model, but also other image properties, like intensity, texture or color. An appearance model can represent both the shape and texture variability seen in a training set, and differ from ASMs in that, instead of searching locally about each model point, they seek to minimize the difference between a new image and one synthesized by the appearance model [35]. ASMs only use data around the model points, and do not take advantage of all the gray-level information available in a whole object as the AAM does.

Topological Active Nets (TANs) are a discrete implementation of an elastic mesh with interrelated nodes [36], that integrates region- and boundary-based features. The model has two kinds of nodes: the external nodes fit the edges of the objects whereas the internal nodes model their internal topology. The advantage of this model is the capability of fitting the edges of the objects while detecting their inner topology. Contrariwise, the model is complex and has limitations regarding topological changes, local deformations, and the definition of the energy functional. A TAN is defined parametrically as $v(r,s) = (x(r,s), y(r,s))$ where $(r,s) \in [0,1] \times [0,1]$. The mesh

deformations are controlled by an energy function defined as follows:

$$E(v(r,s)) = \int_0^1 \int_0^1 [E_{\text{int}}(v(r,s)) + E_{\text{ext}}(v(r,s))]\, drds \tag{2.4}$$

where E_{int} and E_{ext} are the internal and the external energy of the TAN, respectively. The internal energy depends on first and second order derivatives which controls contraction and bending of the shape and the structure of the mesh, whereas the external energy represents the external forces governing the adjustment process (differentiating between both types of nodes: external nodes fit the edges while internal ones model the inner features of the objects).

Finally, Deformable Templates (DTs) represent shapes as deformations of a given prototype or template. Prior knowledge of an object shape is described by a prototype template, usually hand-drawn, which consists of the object's representative contour/edges, and a set of probabilistic transformations on the template. DTs have been successfully applied to object tracking [26] and object matching [25]. To define a deformable template, one needs, firstly, to mathematically define a prototype which describes the prior knowledge about the object shape as the most likely appearance of the object being sought. Secondly, one needs to provide a mathematical description of the possible relationships between the template and all admissible object shapes, that represent the possible transformations which can deform the basic template and turn it into the target object, as appears in the image.

Geometric Deformable Models

Geometric DMs, proposed independently by [37] and [38], provide an elegant solution to address the primary limitations of parametric DMs. These models are based on the curve evolution theory [39–41] and the level set method [42, 43]: curves and surfaces are evolved using only geometric measures, resulting in an evolution that is independent of the parameterization. As in parametric DMs, the evolution is coupled with the image data to recover object boundaries. Since the evolution is independent of the parameterization, the evolving curves and surfaces can be represented implicitly as

a level set of a higher-dimensional function and topological changes can be handled automatically.

Amongst geometric models, the Level Set (LS) method [42] relies on an evolving closed surface defined by a moving interface, the front, which expands from a point out into the image, fitting itself to the region it is released within, and smoothing noise. The interface $\Gamma(t)$ can be characterized as a Lipschitz continuous function:

$$\begin{cases} \phi(t,\mathbf{x}) > 0 & \text{for } \mathbf{x} \text{ inside } \Gamma(t) \\ \phi(t,\mathbf{x}) < 0 & \text{for } \mathbf{x} \text{ outside } \Gamma(t) \\ \phi(t,\mathbf{x}) = 0 & \text{for } \mathbf{x} \text{ on } \Gamma(t) \end{cases}$$

The front, denoted by Γ, is represented by the zero level $\Gamma(t) = \{\mathbf{x}|\phi(t,\mathbf{x}) = 0\}$ of a LS function $\phi(t,\mathbf{x})$. The ϕ evolution can be written in the following general form:

$$\frac{\partial \phi}{\partial t} + F|\nabla \phi| = 0$$

known as the LS equation, where F is called the speed function. This speed can depend on position, time, the geometry of the interface (e.g. its normal or its mean curvature), and the external physics. Among the advantages of LS methods is the natural ability for a single surface to split and merge without losing its identity.

In any case, the definition of the LS function ϕ is essential. One common choice is the signed distance function $d(\mathbf{x})$, which gives the distance of a point to the surface and the sign: generally $d > 0$ if the point $\mathbf{x}$ is outside and $d < 0$ if it is inside the surface (assuming it is a closed surface). This definition is especially interesting to avoid numerical instabilities and inaccuracies during computations. Even with this definition, ϕ will not remain a signed distance function all the time and a reinitialization procedure to keep the LS intact will be needed [44].

Due to its important influence in part of this research work, it seems worth to dedicate a paragraph to explain the "Active Contours Without Edges" (CV) [45] method presented in 2001 by Chan and Vese. This algorithm, based on the Mumford-Shah

functional [46], was designed to detect objects whose boundaries are not necessarily defined by gray level gradients; indeed, it ignores edges completely, making CV a region-based method. The idea is to separate the image into two regions having homogeneous intensity values. More formally, the process minimizes the energy functional shown in Equation 2.5. The functional is used to evolve a LS representing the contour C, using the conventional variational calculus approach. The evolving curve C is the boundary of an open subset w of Ω (i.e., Ω is the image domain, $w \subset \Omega$ and $C = \partial w$), and the regions C and $\Omega \setminus C$ determine the areas inside and outside C, respectively.

$$\begin{aligned} E(\overline{I}_C, \overline{I}_{\Omega\setminus C}, C) = \mu \cdot \text{Length}(C) + \nu \cdot \text{Area}(C) \\ + \lambda_1 \int_C |I(x,y) - \overline{I}_C|^2 dxdy \\ + \lambda_2 \int_{\Omega\setminus C} |I(x,y) - \overline{I}_{\Omega\setminus C}|^2 dxdy \end{aligned} \tag{2.5}$$

In the equation, I is the pixel intensity value of the image to be segmented, and $\overline{I}_C$ and $\overline{I}_{\Omega\setminus C}$ are the average values inside and outside C, respectively[1]. Along with the length of C and its area, there are a third and fourth term representing the variance of the intensity level (i.e., the homogeneity) inside and outside the closed contour. Each term has a weight that determines its influence on the total energy, so that, for instance, the smaller μ, the more the length of the curve can increase without penalizing the minimization. The larger μ, the less freedom is there for the curve to increase in length, thus, it will only be able to detect larger objects. The parameter ν is also a scaling term for the area of the curve, but it can be set to 0 in the Euler-Lagrange derivation, since μ is sufficient to scale the curve according to the objects that need to be detected. Finally, λ_1 and λ_2 are weighting parameters for the forces inside the curve and outside the curve respectively.

Some hybridizations between geometric and parametric DMs have already been presented, like Geodesic Active Contours (GAC) [47], where the proposed approach is based on the relation between active contours and the computation of geodesics or

[1] $\overline{I}_C$ and $\overline{I}_{\Omega\setminus C}$ are also named in many papers as c_1 and c_2, respectively

minimal-distance curves, connecting classic "snakes" based on energy minimization and geometric active contours based on the theory of curve evolution. The technique, introduced in 1997, is based on active contours evolving in time according to intrinsic geometric measures of the image. The evolving contours naturally split and merge, allowing for the simultaneous detection of several objects and of both interior and exterior boundaries.

The partial differential equation of the GAC is the following:

$$u_t = \alpha \cdot div(g\nabla u/|\nabla u|)|\nabla u| + \beta \cdot g|\nabla u| \tag{2.6}$$

where g is an edge indicator function (a positive and strictly decreasing function), u an implicit representation of the curve C, ∇ is the gradient operator, div is the divergence operator (that measures the magnitude of a vector field's source or sink at a given point) [2], and α and β are the contour (internal force) and expansion (external force) weights, respectively. The higher α, the more regularized/smoothed the contour. If β is positive it expands outwards, if negative it expands inwards (moving faster and crossing gradients, the smaller its value).

The problem is formalized as the minimization of the following energy:

$$E(C) = \int_0^{|C|_\varepsilon} g(|\nabla I(C(s))|)ds, \tag{2.7}$$

where $|C|_\varepsilon$ is the Euclidean length of a contour C, s is the arc length of the contour, and ds is equivalent to $|C'(q)|dq$, being $C(q)$ a parameterized planar curve. This method is equivalent to the minimization of the length of the curve C according to a Riemannian metric, which depends on the local gradient of the image I. GAC minimizes Equation 2.7 via a gradient descent scheme and a level-sets representation

[2] The divergence of a vector field is defined as

$$\nabla \cdot \mathbf{V} = \frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y} + \frac{\partial V_z}{\partial z}$$

of the curve. Unfortunately, the method is sensitive to initialization and the global minimum of Equation 2.7 is not always found.

2.3 Medical Image Registration

Image registration (IR) refers to the process of overlaying two or more images of the same scene (taken at different times, from different viewpoints, and/or by different sensors) or, in other words, the process of geometrically aligning multiple images having a shared content [48]. The alignment is represented by a spatial transformation that overlaps the common part of the images. One image, the *scene*, is transformed to match the geometry of the other image, called the *model*.

Three main components characterize an IR method: the *transformation model*, the *similarity metric* and the *optimization process* (see Figure 2.1). The transformation model determines what kind of transformation can be used to align the images. Transformation models vary greatly in complexity, ranging from simple combinations of translation and rotation up to elastic transformations that can represent local deformations and warpings. The choice of the appropriate transformation model for a given application is often crucial.

The similarity metric is the component that measures the quality of an alignment. In medical applications, the most common approach, called *intensity-based*, compares the joint distribution of intensity values between the scene and the model once a transformation has been applied. The degree of matching can be computed from the intensity distributions using measures such as the mean square error, the correlation coefficient or the mutual information [49]. In an alternative approach, called *feature-based*, the alignment is measured only on salient and distinctive features of the image, such as lines, corners and edges, ignoring the rest of the image contents. This can make the problem easier and speed up the registration provided these features can be reliably detected automatically. This is rarely the case in medical imaging because great precision and consistency is required; in the remainder of this article we focus on intensity-based methods.

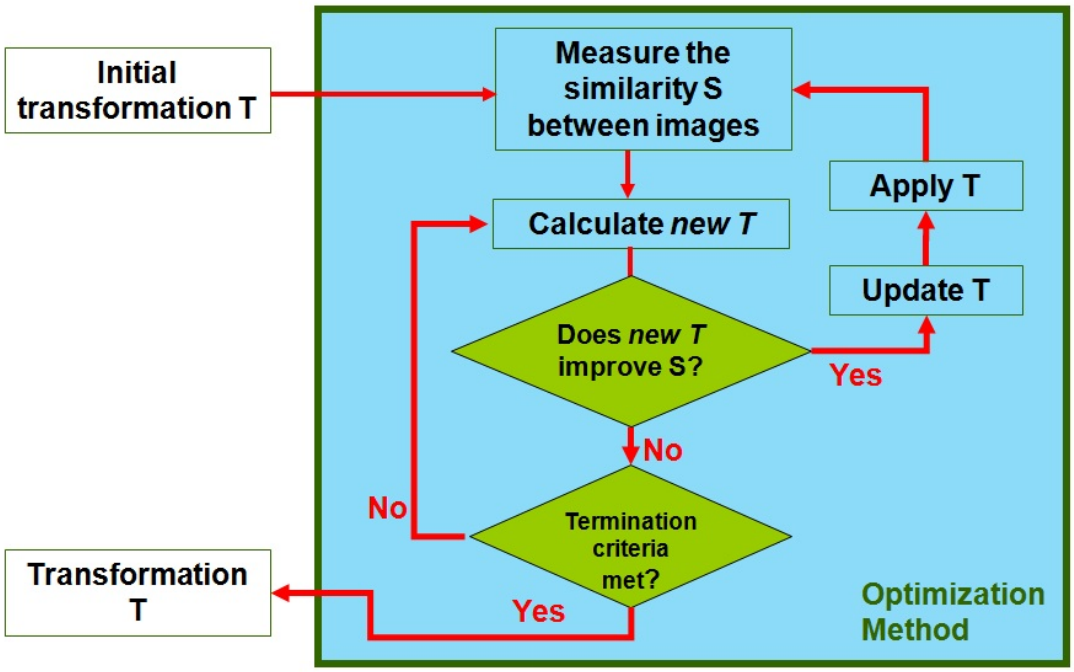


Figure 2.1: Image registration general overview.

The optimization procedure is the component responsible for finding an appropriate transformation to carry out the registration. A transformation is specified by a series of parameters (e.g. a translation vector and a rotation angle), which turns the registration into a continuous optimization problem. Classic numerical optimization algorithms such as gradient descent, Newton's method, Powell's method and discrete optimization [50, 51] are among the most common choices for the optimization component, as well as approaches based on EAs and other metaheuristics [52].

There are cases in which image registration is used as a preliminary step in a segmentation process. An example of segmentation through registration is the atlas-based segmentation, where the availability of an atlas in which the target region has been already labeled is required (i.e., a typical or average image of the anatomical region to be segmented). The atlas-based segmentation process [53] begins by registering the atlas to the input image. Then, the region of the target image that overlaps the labeled region in the atlas is the result of the segmentation process. Figure 2.2 shows a slice of a brain magnetic resonance image and the corresponding deep brain structure

segmentation obtained once the atlas is registered to the input image. The quality of atlas-based segmentation depends closely on the accuracy of the registration step, although the anatomical variability in the target region can limit the effectiveness of the method.

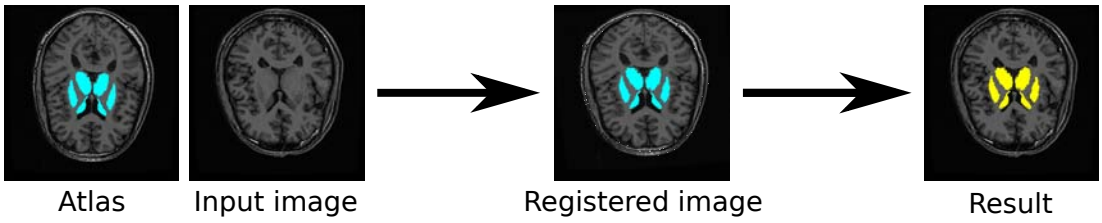


Figure 2.2: An example of atlas-based segmentation [2].

2.4 Texture and Gray Level Co-Occurrence Matrix

A texture is a set of visual elements occurring in some regular or repeated pattern. First order texture measures are statistics calculated from the original image values, like variance, and do not consider pixel neighbour relationships. Second order measures consider the relationship between groups of two (usually neighbouring) pixels in the original image. Third and higher order textures (considering the relationships among three or more pixels) are theoretically possible but not commonly implemented due to their calculation time and difficult interpretation.

Gray Level Co-Occurrence Matrix (GLCM), also called Gray Tone Spatial Dependency Matrix, was introduced by Haralick [54] in 1973, and considers the relation between two pixels at a time, called the reference and the neighbour pixel. It is a feature-based method that characterizes a texture as a homogeneous distribution of feature values. A co-occurrence matrix describes how often a gray level appears in a specified spatial relationship to another gray level. The entry at (i, j) of the GLCM indicates the number of occurrences of the pair of gray levels i and j which are a distance d apart along a given direction θ. The values of d (offset) and θ (direction) are parameters for constructing the GLCM (starting in the upper left corner). A different

co-occurrence matrix exists for each spatial relationship. For instance, if the neighbour pixel is the one to the east (right) of each reference pixel, this can also be expressed as a (1,0) relation: 1 pixel in the x direction, 0 pixels in the y direction. Another example: the (1,1) spatial relationship is equivalent to 1 pixel in the x direction and 1 pixel in the y direction or, in other words, $\theta = 315^\circ$ and d=1 pixel.

A practical example of the GLCM calculation can be found in `http://www.fp.ucalgary.ca/mhallbey/the_glcm.htm`. Let consider the following matrix representing an image gray levels:

$$\begin{matrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 2 & 2 & 2 \\ 2 & 2 & 3 & 3 \end{matrix}$$

If we use the (1,0) spatial relationship the GLCM would be as follows:

$$\begin{matrix} 2 & 2 & 1 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 1 \end{matrix}$$

Therefore, twice the reference pixel and its eastern neighbour are both 0 in the input image. Also, twice the reference pixel is 0 and its eastern neighbour is 1, and three times the reference pixel is 2 and its neighbour is also 2. With this example, one can easily understand the computational burden of this approach. The input image of the previous example had four gray level values (0, 1, 2 and 3), but using eight bit data (256 possible values) it would yield a 256×256 square matrix, with 65,536 cells.

The GLCM is not the only texture measure that has been proposed, but it is the most commonly implemented one. Instead of, or in addition to, this approach, we could employ Laws texture energy measures or Wavelet texture analysis.

2.5 Soft Computing

As complexity rises, precise statements lose meaning
and meaningful statements lose precision
Lofti Zadeh

Opposite to traditional computing methods (hard computing), *soft computing / computational intelligence* (SC) [55] techniques are not based on closed-form solutions (e.g., a formula which solves a specific equation), but mostly on search/optimization procedures (heuristics) and other *approximate* techniques which are able to solve problems also in the presence of qualitative, uncertain, imprecise or incomplete data. Such techniques try to emulate humans' ability when dealing with uncertainty, learning from examples, generalizing knowledge, improving their performances based on experience, cooperating to perform difficult tasks, or evolving (where evolution is considered as a natural/stochastic process driven by "results", and a conscious process driven by experience or intelligence). Since Lotfi Zadeh coined the term "Soft Computing" in 1991, this technological area has developed rapidly both in its theoretical aspects and in its business applications. These methods applied to real-world problems offer more robust, tractable and less costly solutions than those obtained by more conventional mathematical techniques.

The main constituents of SC are fuzzy logic, neural networks, evolutionary computing and probabilistic reasoning:

- **Evolutionary Computation** (EC) include a number of computational models that reproduce natural evolution processes to optimize a goal which is generally represented as a function: **Evolutionary Algorithms** (EAs) [56] and **Swarm Intelligence** (SI) [57, 58]. The name of Evolutionary Computation comes from the use of Darwinian-like *evolutionary* processes to solve difficult *computational* problems, while SI is the discipline that deals with natural and artificial systems composed of many individuals that coordinate using decentralized control and self-organization. In fact, it comprises any attempt to design algorithms or

distributed problem-solving devices inspired by the collective behavior of social insect colonies and other animal societies.

- **Artificial Neural Networks** (ANN) [59–61] simulate a real nervous system formed by a set of neural units coupled by means of synaptic connections. Therefore, ANN is a mathematical/computational model inspired by the structure and functional aspects of biological neural networks, mainly used in classification and regression problems.

- **Fuzzy Systems** (FS) [62, 63] are interpretable models defined in natural language and mainly focused on regression, classification and data mining. Fuzzy systems, including fuzzy logic and fuzzy set theory, provide a rich and meaningful continuous extension to standard binary logic, facilitating the opportunity for modeling conditions which are inherently imprecisely or qualitatively defined.

- **Probabilistic Reasoning** (PBR) [64, 65] deals with knowledge representation and reasoning and encompasses belief networks and parts of learning theory. A belief network or Bayesian network is a directed acyclic graph with nodes including probability information that represents the dependence between variables.

It is necessary to properly contextualize the term SC with other related words as Artificial Life (AL), Artificial Intelligence (AI) and Machine Learning (ML). On the one hand, SC is an area of AI[2] research focused on the design of intelligent systems to process uncertain, imprecise and incomplete information. On the other hand, AL [3] "investigates the scientific, engineering, philosophical and social aspects of our technological ability of synthesizing behaviors similar to life, starting from scratch, in computers, machines, molecules or other alternative means" [66]. Thus, from this

[2] See Table 2.1 for a broad outlook of AI [9]

[3] The first international conference on this field took place in Santa Fe, New Mexico, in 1987. Chris Langton, organizer of such an event, coined the following famous definition: "AL is the study of man-made systems that exhibit behaviors characteristic of natural living systems. It complements the traditional biological sciences concerned with the analysis of living organisms by attempting to synthesize life-like behaviors within computers and other artificial media. By extending the empirical foundation upon which biology is based beyond the carbon-chain life that has evolved on Earth, Artificial Life can contribute to theoretical biology by locating life-as-we-know-it within the larger picture of life-as-it-could-be."

viewpoint, biologically inspired computing approaches like ANN, EC, cellular automata and autonomous robotics would directly belong to this field of knowledge. Finally, ML [67, 68] allows computers to learn, i.e. to gradually optimize their performance, using example data or past experience, and Pattern Recognition (PR) is a branch of AI devoted to the study of ANN, kernel methods, bayesian networks, gaussian mixture models, continuous latent variables and Markov Models, among others, for solving regression, classification and clustering problems.

Table 2.1: Definitions of AI organized according to four categories [9].

Systems that think like humans
"The exciting new effort to make computers think... machines with minds, in the full and literal sense" [69] "The automation of activities that we associate with human thinking, activities such as decision-making, problem solving, learning ..." [70]
Systems that think rationally
"The study of mental faculties through the use of computational models" [71] "The study of the computations that make it possible to perceive, reason, and act" [72]
Systems that act like humans
"The art of creating machines that perform functions that require intelligence when performed by people" [73] "The study of how to make computers do things at which, at the moment, people are better" [74]
Systems that act rationally
"A field of study that seeks to explain and emulate intelligent behavior in terms of computational processes" [75] "The branch of computer science that is concerned with the automation of intelligent behavior" [76]

Metaheuristics

Any organized scheme to try to compose
an accurate model of human life
should contain some degree of anarchy.
Bertrand Russell, Sceptical Essays, 1928

As mentioned in previous sections, the segmentation task can be turned into an optimization problem, in which some kind of energy functional needs to be minimized/maximized, and solved by means of search/optimization methods. Following [77], these optimization techniques can be broadly classified as *calculus-based*, *enumerative*, and *guided stochastic search*. While the calculus-based methods assume the existence of derivatives, and are local in scope, restricting severely their application, the enumerative techniques fail when the size of the search space is large, as in medical IS. Guided stochastic search techniques are based on enumerative methods, but use additional information about the search space. These can be further divided into *single-point search* (like Simulated Annealing) and *multiple-point search* (like population-based algorithms), depending on whether they search considering just one point or several points at a time. Another taxonomy for optimization algorithms can take into account the *direct* or *indirect* nature of the solution search, depending on the use of derivatives. According to this criterion, EAs would be located in the first class (direct methods), while gradient methods (first order optimization algorithms) and the Newton method (second order optimization algorithms) would be located in the second one.

The gradient descent and many other classic optimization techniques (like Newton's method, Levenberg-Marquardt algorithm or Nelder-Mead method) are effective when the problems under consideration satisfy tight constraints (differentiable, continuous, well-defined and convex functions). But when the search space is discontinuous, noisy, high dimensional, non-convex and multimodal, then stochastic algorithms have been found to consistently outperform traditional methods [78, 79]. Among the stochastic approaches to continuous optimization, EAs and SI algorithms, as well as other **metaheuristics** (MHs) [80], offer a number of attractive features: no requirement for a

differentiable or continuous objective function, robust and reliable performance, global search capability, virtually no need of specific information about the problem to solve, easy implementation, and implicit parallelism. In particular, these techniques can achieve very good results in continuous optimization problems, achieving a trade-off between global exploration (to localize promising regions) and exploitation of the located regions (using previously found solutions to obtain more accurate results).

MHs are approximate algorithms, in the sense that they do not guarantee to find the optimal solution but a good approximation in reasonable time. They are not problem-specific (permitting an abstract level of description) and usually non-deterministic algorithms. The main objective of these optimization/learning procedures is to effectively explore the search space, achieving a trade-off between intensification (exploitation of the accumulated search experience) and diversification (global exploration of the search space).

In this work, MHs are going to be organized according to the following taxonomy:

- Trajectory methods. The search process is characterized by a trajectory in the search space, and can be seen as the evolution in (discrete) time of a discrete dynamical system. Examples of this category could be Tabu Search [81], Simulated Annealing [82], Iterated Local Search [83] or Variable Neighborhood Search [84].

- Population-based methods. These techniques deal in every iteration of the algorithm with a set – a population – of solutions. In this case, the search process can be seen as the evolution in (discrete) time of a set of points in the search space. Paradigmatic cases in this regard are Evolutionary Algorithms [56], Swarm Intelligence techniques [85], and Memetic Algorithms [86].

The most commonly used MHs in combination with computer vision techniques in medical imaging problems are described in the next subsections. By far, the most used metaheuristics in registration and segmentation are Genetic Algorithms, Particle Swarm Optimization, Simulated Annealing, Differential Evolution, and Scatter Search. Other approaches, even if not very often present in medical imaging literature, like

Evolutionary Programming and Genetic Programming, are briefly introduced here for historical reasons, in order to give a short overview of the field.

Genetic Algorithms

Genetic Algorithms (GAs) are stochastic, parallel search algorithms based on the mechanics of natural selection [87–90]. GAs were designed to efficiently search large, non-linear, poorly-understood search spaces where expert knowledge is scarce or difficult to encode and where traditional optimization techniques fail. They are flexible, robust, and try to exhibit the adaptiveness of biological systems.

These algorithms encode a potential solution to a specific problem into a chromosome-like data structure and apply recombination operators to preserve critical information. The main features of a GA are the encoding of individuals as strings of symbols, the individuals selection policy, and the use of both the mutation and recombination operators. The basic working of a GA is shown in Algorithm 1.

Algorithm 1 Genetic Algorithm Pseudocode

```
Generate a random population of chromosomes
while stopping criterion is not met do
    Decode each chromosome into an individual
    Evaluate each individual's fitness
    Generate a new population, partly by cloning (copying), partly by recombining,
    partly by mutating the chromosomes of some selected individuals (based on their
    fitness)
end while
```

GAs process a population of solutions by three operations: selection, crossover and mutation. In the initial formulation, the solutions were binary encoded; however, other encoding types have been taken into account for the representation. In particular, real encoding seems particularly natural when optimizing parameters in continuous domains. In that case, a chromosome is a vector of real numbers, each of which is one of the parameters to be optimized. GAs based on real-number representations are called real-coded GAs (RCGAs).

There is a compelling evidence indicating that classic discrete crossover operators (DCOs), i.e., all the crossover operators used for binary encoding which are directly applicable to real coding (like the simple, two-point and uniform crossover operators), are ineffective for RCGAs [91, 92]. Since the crossover operators that exploit the numerical nature of the real coding (aggregation-based operators, like arithmetic or geometric, and neighborhood-based crossover operators, like BLX-α or SBX) consistently outperform the classic DCOs, these kinds of operators are strongly recommended.

Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a bio-inspired optimization algorithm introduced by Kennedy and Eberhart [93]. It is based on the simulation of the social behaviour of bird flocks, so it is clearly located within the SI techniques. In the last fifteen years PSO has been applied to a very large variety of problems [94] and many variants of the original algorithm have been proposed [95].

During the execution of PSO a set of particles moves within the function domain searching for the optimum of the function (best fitness value). The motion of the i^{th} particle can be described by the following two simple difference equations which regulate the particle's position and velocity:

$$\begin{aligned} P_i(t) &= P_i(t-1) + v_i(t) \\ v_i(t) &= w \cdot v_i(t-1) \\ &+ c_1 \cdot rand() \cdot (BP_i - P_i(t-1)) \\ &+ c_2 \cdot rand() \cdot (BGP - P_i(t-1)) \end{aligned}$$

where $P_i(t)$ and $v_i(t)$ are the position and velocity of the particle in the present iteration, c_1, c_2 and w (inertia factor) are positive constants, $rand()$ returns random values uniformly distributed in $[0, 1]$, and BP_i is the best-fitness position visited so far by the particle. In the basic algorithm (global-best PSO), BGP is the best-fitness position visited so far by any particle of the swarm. However, there is a set of variants, termed

local-best PSO, where the swarm is subdivided into particle neighborhoods which can assume different topologies. In that case, BGP becomes BGP_i and represents the best-fitness position visited so far by any particle in the i^{th} particle's neighborhood.

The PSO variant used in this PhD presents two main features that differentiate it from the original one. The first has been suggested by Liu et al [96]. Instead of using a static inertia factor w, they adapt its value to the fitness function of each particle. In particular, if the objective is to minimize the fitness value, the so-called adaptive inertia weight factor (AIWF) is determined as follows:

$$w = \begin{cases} w_{min} + \frac{(w_{max} - w_{min}) \cdot (f - f_{min})}{f_{avg} - f_{min}} & \text{if } f \leq f_{avg} \\ w_{max} & \text{if } f > f_{avg} \end{cases}$$

where w_{max} and w_{min} denote the maximum and minimum possible values of w, f is the current fitness of the particle, f_{avg} and f_{min} are the average and minimum fitnesses of all particles of the swarm, respectively. This way, w varies depending on a particle's fitness so that good particles tend to perform exploitation to refine results by local search, while bad particles tend to further explore the search space.

The second change with respect to the original algorithm is the re-initialization of a particle in case of stagnation. When a particle can not improve its best position in a preset number of iterations, it moves to a random direction with very high velocity:

$$\begin{aligned} v_n(t) &= k \cdot randn() \\ P_n(t) &= P_n(t-1) + v_n(t) \end{aligned}$$

where $k > 1$ is a constant and $randn()$ returns random values from a Gaussian distribution. This way, the search performed by PSO privileges exploration of the search space to reduce the probability of falling into a local optimum.

Differential Evolution

Differential Evolution (DE), first introduced by Storn and Price [97], has recently been shown to be one of the most successful EAs for global continuous optimization [98]. Unlike traditional EAs, DE perturbs current generation individuals by the scaled differences of other randomly selected and distinct individuals. Therefore, no separate probability distribution has to be used for generating the offspring [99].

In DE, new individuals that will be part of the next generation are created by combining individuals that are already members of the current population. Each individual acts as a parent vector, for which a new solution, called donor vector, is created. In the basic version of DE, the donor vector for the i^{th} parent (X_i) is generated by combining three random and distinct elements X_{r1}, X_{r2} and X_{r3}. The donor vector V_i is calculated as follows:

$$V_i = X_{r1} + F \cdot (X_{r2} - X_{r3})$$

where F (scale factor) is a parameter that strongly influences DE's performances and typically lies in the interval $[0.4,\ 1]$. Recently, several mutation strategies have been applied to DE, experimenting with different base vectors and different numbers of vectors for perturbations. For example, the original method explained above is called DE/rand/1, which means that the first element of the donor vector equation X_{r1} is randomly chosen and only one difference vector (in this case $X_{r2} - X_{r3}$) is added. See Table 2.2 for a listing of some classic mutation methods.

After mutation, every parent-donor pair generates a child (called trial vector) by means of a crossover operation. Two kinds of crossover are typically used: binomial (also termed uniform) and exponential. The crossover is applied with a certain probability, defined by a parameter Cr (crossover rate) that, like F, is one of the control parameters of DE. Then, the trial vector is evaluated and its fitness is compared to its parent's: the best survives and will be part of the next generation.

DE shares some features with SI techniques, mainly in the interaction among par-

Table 2.2: Mutation methods in DE.

Name	Donor Vector
DE/best/1	$V_i = X_{best} + F \cdot (X_{r1} - X_{r2})$
DE/target-to-best/1	$V_i = X_i + F \cdot (X_{best} - X_i) + F \cdot (X_{r1} - X_{r2})$
DE/best/2	$V_i = X_{best} + F \cdot (X_{r1} - X_{r2}) + F \cdot (X_{r3} - X_{r4})$
DE/rand/2	$V_i = X_{r1} + F \cdot (X_{r2} - X_{r3}) + F \cdot (X_{r4} - X_{r5})$
DE/rand-to-best/2	$V_i = X_{r1} + F \cdot (X_{best} - X_i) + F \cdot (X_{r2} - X_{r3}) + F \cdot (X_{r4} - X_{r5})$

ticles and in the selection scheme. In particular, both DE and PSO are stochastic, population based, real-valued algorithms, and designed for challenging continuous optimization problems (non-differentiable, nonlinear and/or multimodal functions) using few control parameters. DE can also be considered as an EA, but differs from traditional EAs in the aspect of generating new vectors by adding the weighted difference vector between two population members to a third member.

Memetic Algorithms

Memetic Algorithms (MAs) [86, 100] are hybrid global-local search methods in which a local improvement procedure is incorporated to a traditional EA. MAs are also known as Hybrid GAs, Lamarckian learning GAs, and Baldwinian learning GAs. The idea is to imitate the effect of learning and social interaction during the life span of an individual by some kind of (local) improvement mechanisms (memes [101]) applied to the offspring created by the usual operators of an EA.

Regarding design issues, the frequency and intensity of individual learning directly define the degree of evolution (exploration) against individual learning (exploitation) in the MA search. Furthermore, when only a portion of the population undergo learning, the issue of choosing which subset of individuals to improve need to be considered to maximize the utility of MA optimization. Finally, the individual learning procedure/meme used also favors a different neighborhood structure, hence there is the need to decide which meme or memes should be used for the optimization problem at hand.

Scatter Search

Scatter Search (SS) [102] is a MA based on a systematic combination (instead of randomized, as is usual in EAs) between solutions taken from a subset of the population, named the "reference set", that is usually significantly smaller than a typical EA population. SS is composed of five structural "blocks" or methods:

1. Diversification Generation: a population of solutions P is built with a certain degree of quality and diversity. The reference set R is then drawn from P, and is composed of the $|R_1|$ solutions with best fitness, and the $|R_2|$ solutions from P (hence, $|R|$ = $|R_1|$ +$|R_2|$) that are farthest, based on a particular metric (usually the Euclidean distance), from the reference set; the evolution process acts only on R;

2. Solution Combination: in most problems a specific solution combination method is needed, which can be applied to all solutions or only to selected ones (e.g., the best solutions, and/or randomly selected ones). In many cases an existing crossover operator, borrowed from other EAs, can be employed;

3. Subset Generation: the procedure deterministically generates subsets of R, to which the combination method is applied.

4. Improvement: to obtain high-quality solutions, an improvement method (typically a local search method) is applied to the original solutions and/or to combined solutions;

5. Reference Set Update: once a new solution is obtained (applying the combination method) it replaces the worst solution in R only if it improves the quality of the reference set in terms of fitness and/or diversity;

As improvement method in a MA, several well-known continuous local search methods, like Solis&Wets [103] or Nelder&Mead's simplex [104], have been commonly used [105–108]. For instance, Solis&Wets local search method is a randomized hill-climber with adaptive step size. Each step starts at a point x. A perturbation p is

randomly chosen from a Gaussian distribution with standard deviation ρ. If either $x + p$ or $x - p$ has a better fitness than x, a move to the best point is performed and a success is recorded, otherwise the position does not change and a failure is recorded. After N^+ consecutive successes ρ is increased, for getting faster to the local optima, while after N^- failures in a row, ρ is consequently decreased.

Simulated Annealing

Simulated Annealing [82] (SA) tries to mimic the physical annealing process, where a material is heated and slowly cooled into a uniform structure. SA may even perform bad moves (i.e. changes which leads to worse fitness) accordingly to a probability distribution dependent on the temperature of the system: a move is selected at random and, then, as the temperature decreases, the probability of accepting a bad move decreases (when temperature is zero, no bad moves are accepted, i.e. it behaves like hill climbing). The main building blocks of the algorithm are three: the candidate solution generation method (sometimes referred to as the neighborhood function), the acceptance probability function mentioned earlier, and the cooling schedule.

Evolution Strategies

Evolution strategies (ES) [109–111] are typically applied to continuous optimization problems, where an individual is a vector of real-valued parameters, mutation is normally performed by adding a normally distributed random value to each vector component, and the selection is deterministic. The replacement strategies are called (μ, λ) and $(\mu + \lambda)$, being μ the parents and λ the children created using the evolutionary operators. The survivor selection procedure of the former is based on the set of children, while the latter uses the set of parents and children (elitism). The simplest version ((1 + 1)-ES) operates on a population of size two: the current point (parent) and the result of its mutation. Only if the mutant's fitness is at least as good as its parents', it becomes the parent of the next generation. The main particularity of ES lies in the self-adaptation of the standard deviation of the Gaussian distribution used in the mutation [112].

Evolutionary Programming

Traditional Evolutionary Programming (EP) [113] was concerned with evolving finite state automata for machine learning tasks. Representation and operators were specialized for this application area. Each parent gave birth, by mutation, only to one offspring, and a replacement strategy was used to remove half of the individuals. Contemporary EP [114] has evolved to use any representation and different evolution engines, and nowadays differs from ES by using a stochastic form of replacement strategy, while it also uses self-adaptation of Gaussian mutation in the case of real-valued genotypes.

There has long been a strong debate about the usefulness of crossover. The GA community considers crossover to be the essential variation operator, while mutation is only a background necessity. On the other hand, the historical ES and EP researchers did not use any crossover at all, and even claimed later that it could be harmful. The general agreement nowadays is that the answer is problem-dependent: If there exists a "semantically meaningful" crossover for the problem at hand, it is probably a good idea to use it. But otherwise mutation alone might be sufficient to find good solutions— and the resulting algorithm can still be called an EA [115].

Genetic Programming

Genetic Programming (GP) is the youngest technique within this family of methods [116, 117], and has a specific application area in machine learning and modeling tasks. Parse-trees of formal expressions describing a model or a procedure is its natural representation. Crossover and mutation operators are adapted so that they work on trees (with varying sizes). The evolution engine is "inherited" from GAs (GP has long been seen as GA with a tree-based representation). On the other hand, syntactic expressions – for instance LISP – can be viewed as programs, which makes GP the branch of EC concerned with the automatic evolution of programs.

Classification problems

Learning denotes changes in the system that are adaptive in the sense that they enable the system to do the task or tasks drawn from the same population more effectively the next time.
Herbert Simon

As explained in section 2.5, ML is concerned with the development of algorithms and techniques that allow computers to learn. Learning is used when human expertise does not exist (navigation on Mars), humans are unable to explain their expertise (speech recognition), solution changes in time (routing on a computer network, junk email), solution needs to be adapted to particular cases (user biometrics), humans are expensive to train (zipcode recognition), or the problem size is too vast for our limited reasoning capabilities (calculating webpage ranks, discovering astronomical objects). The main idea is to learn general models from particular examples of data, trying to build good and useful approximation to the data.

We can distinguish three main different kinds of problems in the ML framework: supervised learning (regression and classification) and unsupervised learning (clustering). In supervised learning, a "teacher" presents inputs and desired outputs to the system and, after a training process, the machine is theoretically able to generalize its knowledge and classify/identify novel sets of data. In contrast, with unsupervised ML, the system learns to properly execute a function on its own, through reinforcement and/or automatically detecting patterns and regularities in the input data.

- *Regression* is related to predicting unknown or missing values. In other words, to find a function that fits the input patterns.
- *Clustering* methods try to group the input data into classes before knowing a priori the existence of any class. Therefore, clustering is a mathematical tool that attempts to discover structures or certain patterns in a data set, where the objects inside each cluster show a certain degree of similarity.

- Finally, a *classification* algorithm tries to find the class to which a new instance belongs to. The problem can be mathematically defined as a mapping $f : P \rightarrow C$, where $P = \{p_1, p_2, \ldots p_n\}$ represents the input patterns to be classified and $C = \{c_1, c_2, \ldots c_m\}$ is the set of classes. After a training process, in which patterns and regularities in the data have been detected, the model created by training is tested using samples which were not included in the training set. Some of the most commonly used classification methods (classifiers) are ANN, support vector machines (SVM), and decision trees (DTrs).

Ensemble Classifiers

In statistics and ML, Ensemble Classifiers (ECL) apply multiple classifiers in order to obtain better predictive performance than could be obtained from any of the constituent models [118–120]. An ensemble consists of a set of individually trained classifiers, such an ANN or DTrs, whose predictions are combined when classifying novel instances. Thus, the basic idea is to learn a set of classifiers (experts) and to allow them to vote. The main advantage is the improvement obtained in predictive accuracy, while the principal disadvantage is the difficulty of perfectly understanding its working and a higher computational load.

Essentially, there are four methods for constructing an ECL. All of them are based on manipulating one of the following aspects:

- Training set. Multiple training sets are created by resampling the data according to some sampling distribution. The sampling distribution determines how likely an example is to be selected for training. Then, a classifier is built from each training set using a learning algorithm.

- Input features. A subset of the input features is chosen to form each training set: such subset can be chosen randomly or based on inputs given by domain experts. This approach is really interesting for data that has redundant features.

- Class labels. When the number of classes is sufficiently large, training data

is transformed into a binary class problem by randomly partitioning the class labels into 2 disjoint subsets. Later, re-labelled examples are used to train a base classifier and, by repeating the class labeling and model building steps several times, an ensemble of base classifiers is obtained.

- Learning algorithms. Learning algorithms can be manipulated in such a way that applying the algorithm several times on the same training data may result in different models. Example: ANN can produce different models by changing network topology or the initial weights of links between neurons.

Two of the most succesful ECLs are Boosting (each classifier works on the incorrectly classified instances of the previous one) and Bagging (each classifier is built using a randomly drawn sample of the data). In particular, a Random Forest (RF) [121] is an ECL that consists of a bagging of DTrs (usually CART [122]) with a randomized selection of features at each split: a first randomization through bagging, and a second randomization through predictor subsets; i.e. in addition to constructing each tree using a different bootstrap sample of the data, in RF each node is split using the best among a subset of predictors randomly chosen at that node. This strategy has been shown to perform very well compared to many other classifiers: RF produces a highly accurate classifier, it is able to handle a very large number of input variables, and is robust against overfitting.

Chapter 3

Datasets

In this chapter, we describe the datasets and medical image modalities used in our experiments, as well as we provide a short and general overview about the most common medical imaging modalities.

3.1 Medical Imaging

Medical Imaging (MI) is the production of visual representations of microscopic biological entities and body parts, tissues, or organs, for use in clinical diagnosis (medical procedures seeking to reveal, diagnose, or examine diseases) or medical/biological science (including the study of normal anatomy and physiology); it encompasses computed tomography (CT), magnetic resonance imaging (MRI), positron-emission tomography (PET), single-photon-emission (SPECT), microscopy and ultrasound (US) imaging, among others. MI refers to a number of techniques that can be used as non-invasive methods, thus the body does not have to be surgically operated by medical practitioners to examine various organs and areas. According to the *New England Journal of Medicine*, MI is one of the top developments that "changed the face of clinical medicine" during the last millennium.

In Figure 3.1, a general overview of MI modalities can be seen. The figure shows a

Medical imaging Modalities

Lower numbers means better

	RX	Fluoro	MI	US	CT	MRI
Resolution	1	2	4	3	2	2
Contrast	2	2	1	3	1	1
Noise	2	3	4	3	1	2
3D*	No	No	Yes	Yes	Yes	Yes
Anatomical†,♦	Yes	Yes	No	Yes	Yes	Yes
Functional†	No	No	Yes	No	No	Yes
Dynamic**	No	Yes	No	Yes	No	Yes
Dose‡	2	4	3	1	5	1

(*) 3D vs projection: Ability to adquire (not to display!!) volumetric regions of interest (ROI).
(†) Functional vs. Anatomical: Ability to represent methabolic features (where pathology is not associated to anatomical changes)
(♦) Anatomical information are adquired via combining two devices in a single machine (SPECT – CT, PET – CT, PET – MR)
(**) Dynamic vs. static: Ability to display real time images.
(‡) Radiation dose imparted to patient (potentially harmful!!)

Figure 3.1: Major medical imaging modalities. Figure originally developed by Miguel Ángel Peinado Montes (Service of Medical Physics and Radiation Protection, Central Universitary Hospital of Asturias). From left to right: X-ray (RX), Fluoroscopy (Fluoro), Molecular Imaging (MI), US, CT and MRI.

comparative table with the characteristics of every modality. The next sections will be devoted to a brief explanation of the three image modalities used in the experiments.

MI is a research field closely related to other scientific and technological disciplines like:

- eHealth. A relatively recent term for healthcare practice supported by electronic processes and communication [183].
- Health/Medical/Biomedical informatics. A discipline at the intersection of information science, computer science, and health care, dealing with the resources, devices, and methods required to optimize the acquisition, storage, retrieval, and use of information in health and biomedicine.
- Computational biology. It involves the development and application of data-analytical and theoretical methods, mathematical modeling and computational simulation techniques to the study of biological, behavioral, and social systems.
- Bioinformatics. An interdisciplinary field that develops and improves upon methods for storing, retrieving, organizing and analyzing biological data. The primary goal of bioinformatics is to increase the understanding of biological processes by means of computer science and information technology.

3.2 Microscopy Images

Microscopy imaging techniques are employed by scientists and researchers to improve their possibilities to study the microscopic world. Indeed, advances in microscopy enable visualization of a broad range of biological processes and features in cell structure. An example of a publicly available database containing microscopy images is the Allen Brain Atlas (ABA), a huge database that contains high-resolution images mapping the expression patterns of most genes that are present in the genomes of the analyzed organisms [1]. The first release of the ABA was focused on mouse, and contained the expression patterns of about 20,000 genes obtained by In Situ Hybridization (ISH)

[1]Gene expression is the process by which information from a gene is used in the synthesis of a functional gene product (usually proteins). In non-protein coding genes such as ribosomal RNA (rRNA), transfer RNA (tRNA) or small nuclear RNA (snRNA) genes, the product is a functional RNA.

of mouse brain sections [184]. In a more recent release, a similar resource based on spatially mapped microarray data has also been provided for the human brain [185].

There is wide availability of brain images containing morphological and functional information on the hippocampus in different organisms. Thus, it has become extremely important to design image analysis methods that accurately, robustly, and reproducibly identify the hippocampus region, to automatize any relevant analytic procedure. In the ABA, for each gene under consideration, several images are provided, corresponding to different sections of the brain. Each image is labelled according to the corresponding images of a reference atlas comprising 132 coronal and 21 sagittal sections spaced at 100 μm and 200 μm intervals, respectively.

In order to understand how gene-expression images are obtained, it is fundamental to know that ISH is a technique that allows for precise localization of a specific segment of nucleic acid within a histologic section. The underlying basis of ISH [186] is that nucleic acids, if preserved adequately, can be detected through the application of a complementary strand of nucleic acid to which a reporter molecule is attached[2]. Visualization of the reporter molecule allows to localize DNA or RNA sequences in heterogeneous cell populations (like tissue samples). ISH is a powerful technique for localizing specific nucleic acid targets within fixed tissues and cells, allowing one to obtain temporal and spatial information about gene expression.

Completing the ABA required imaging more than 1 million individual brain sections at cellular resolution labeled via colorimetric ISH (cISH) or Nissl staining at a resolution sufficient to visualize individual cells. To perform such a task, a high-throughput, fully scalable, automated microscopy platform capable of imaging more than 3000 brain sections daily was developed. This system runs 24 h per day and 7 days per week with minimal operator supervision. The platform comprises a barcode scanner, a fully-automated microscope, stage, slide loader, digital camera, and custom software for managing the microscopy process. When operating at full capacity, the imaging system generates approximately one terabyte of uncompressed image data daily. Image data

[2]A reporter molecule is a molecule, whose presence is readily detected (for example, a fluorescent molecule), that is attached to a DNA sequence we wish to monitor.

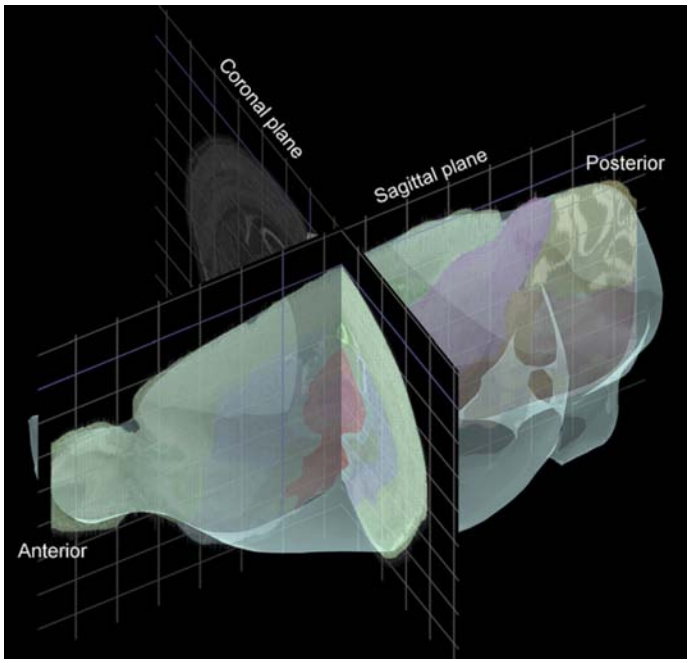


Figure 3.2: A screenshot image from the Brain Explorer application [7].

from the platform are freely available online as part of the ABA [7]. In Figure 3.2, a screenshot image from the Brain Explorer application, that can be used to display ABA data in three dimensions, is displayed. This shot shows the orientation of sagittal and coronal slices through the brain. Brain Explorer may be downloaded for free from the ABA Web site at `http://www.brain-map.org`.

The key features of the Allen Mouse Brain Atlas are: open, public online access; image-based data comprising genome-wide coverage; comprehensive anatomic coverage of the adult mouse brain; microscopic resolution down to the cellular level (see Figure 3.3); sophisticated data search and viewing tools; interactive Brain Explorer 3-D viewer (see Figure 3.2); and detailed anatomic reference atlas of the adult mouse brain that can be viewed with the data. From now on, since the microscopy images used in this research represent tissues, the terms ‘microscopy images’ and ‘histological images’ will be equally used.

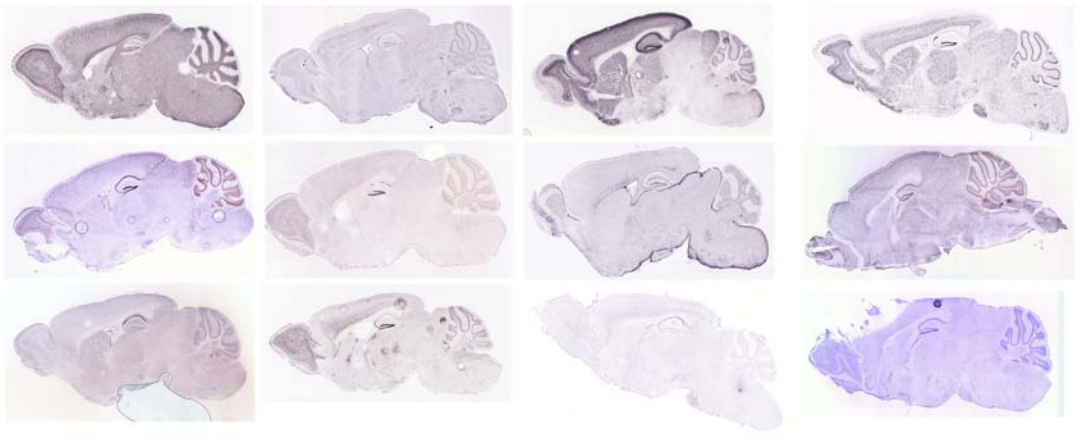


Figure 3.3: Examples of sagittal slices from the Allen Mouse Brain Atlas.

3.3 Computed Tomography Images

X-ray CT is a medical imaging technique in which a three-dimensional image of a body structure, created using X-rays, is constructed by computer from a series of plane cross-sectional images acquired along an axis. These cross-sectional images are used for diagnostic and therapeutic purposes in various medical disciplines. In fact, CT has proven to be a valuable tool for medical diagnostic and, in 1979, the Nobel Prize in Medicine was awarded to the inventors: Allan M. Cormack and Godfrey N. Hounsfield.

CT is based on the equation that quantifies the attenuation of the radiation beam passing through a given material (fat, muscle, water, bone, ...), called Beer-Lambert law [8, 187]:

$$I_x = I_0 \cdot e^{-\mu x}$$

where I_x is the intensity value of the radiation X after passing an object of thickness x (number of emergent photons), I_0 is the intentisty value of the radiation incident on the same surface (number of incident photons), and μ is the linear attenuation coefficient, whose value depends on the atomic number, the density of the material and the wavelength of the incident radiation beam. In optics, the Beer-Lambert law relates the absorption of light to the properties of the material through which the light is travelling, stating that there is a logarithmic dependence between the transmission

of light through a substance, the substance absorption coefficient, and the distance the light travels through the material.

Both CT and conventional X-rays take pictures of internal body structures. In conventional X-rays, the structures overlap: they are often obscured by other organs or bones, making diagnosis difficult. In a CT image, a full 3D dataset is acquired which can be processed to show any structure within it. During CT imaging, an X-ray tube rotates around the patient so that multiple images are collected from many angles and, integrating over all the different views, it is possible to compute the density of any voxel within the 3D structure to be analyzed (Radon transform[3]).

Figure 3.4 shows a radiation beam passing through a region with different densities, and another radiation beam passing through a region with uniform density. Although the total attenuation is similar in both cases, the attenuation profile through the thickness is different. In this case, a conventional radiography only offers the average μ value of the path traversed by the X-ray beam. However, CT gives the average attenuation of each one of the voxels of the body structure, i.e. it provides the different μ values (μ_{1A}, μ_{2A}, μ_{3A}, and μ_{4A}). Having said that, it is not possible to have all this information using a single beam, but it is necessary to apply different beams to each voxel from different angles (projections).

The experiments performed with CT in this PhD dissertation are described in Section 7, and include the 10 CT images (representing lungs and knee) used in [134].

3.4 Magnetic Resonance Images

Magnetic Resonance Imaging (MRI) is an imaging modality which uses external magnetic fields and non-ionizing radiation to provide cross-sectional images from the human body (therefore, with no exposure to harmful radiation). The main principle on

[3]The Radon transform is widely applicable to tomography: the creation of an image from the scattering data associated with cross-sectional scans of an object. If a function f represents an unknown density, then the Radon transform represents the scattering data obtained as the output of a tomographic scan. Hence the inverse of the Radon transform can be used to reconstruct the original density from the scattering data.

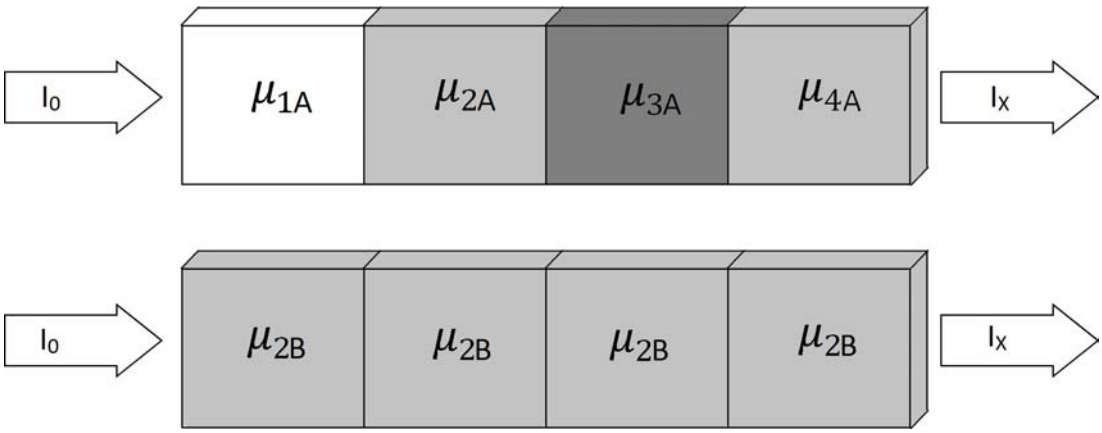


Figure 3.4: Similar total attenuation in two different situations. At the top, a beam passes through an area with different densities while, below, it passes through an area with uniform density. CT is able to discriminate between one case and the other, while X-rays are not. Example taken from [8].

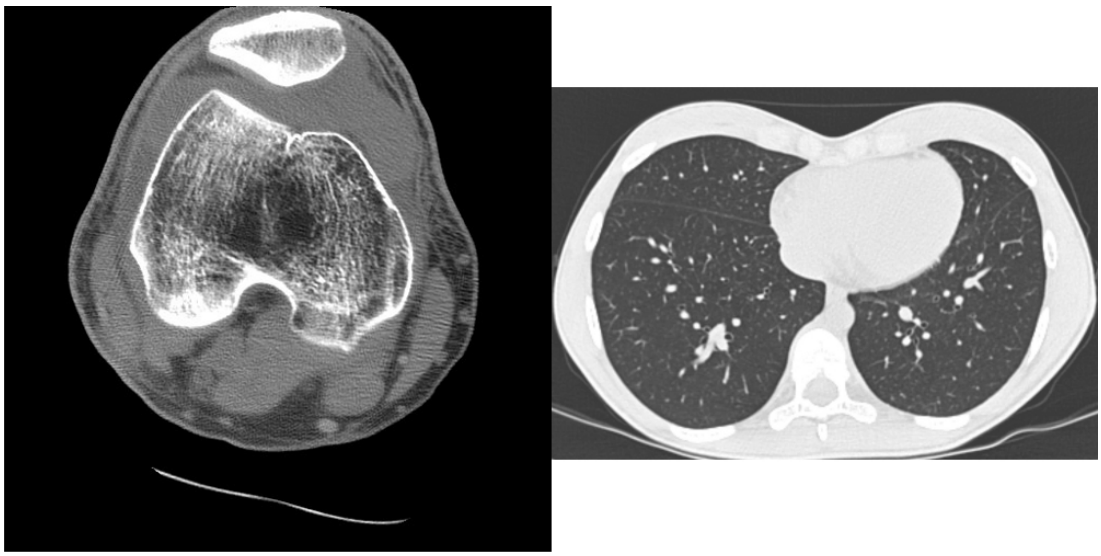


Figure 3.5: Examples of CT images: knee (left) and lung (right)

which MRI is based was published by Felix Bloch and Edward Mills Purcell in 1946: they demostrated that certain nuclei, subjected to the action of an intense magnetic field, are able to absorb energy from radio frequency (RF) waves and emit, in turn, RF signals that can be received by an antenna. Thanks to this discovery, they received the Nobel Prize in Physics in 1952. The first experiences applying MR to the biomedical science took place in the 1950's, and the first MR machine was created and patented in 1972 by Raymond Damadian. In the 1970's, Paul Lauterbur and Peter Mansfield studied how to generate pictures from MR scans and how such a scan can be done. In 2003, both researchers obtained the Nobel Prize in Physiology and Medicine "for their discoveries concerning magnetic resonance imaging".

MR is based on the interaction of magnetic fields and RF waves with the physical matter. The physics fundamentals of this technique are summarized in Figure 3.6. Essentially, an MRI scanner is a device in which the patient lies inside a large magnet where the magnetic field is used to align the magnetization of some atomic nuclei in the body; RF waves are applied to systematically alter the alignment of this magnetization, and the signal emitted by the patient, after the magnetic impulse ends, is received and used to construct the image.

The basis of MRI is the behavior of the hydrogen nuclei under the influence of magnetic fields. A hydrogen nucleus is made up of a single proton, and protons have a spin[4]. Moreover, since these particles are electrically charged, when spinning, they produce a tiny magnetic field whose direction is the axis of rotation (magnetic dipole). Hydrogen is the base element of the MRI diagnostic techniques because it is the most abundant one in the human body (60-90%) and, from the technical point of view, it is the easiest atom to deal with magnetic fields. However, the principles and concepts are applicable to any atomic nucleus that has a non-zero magnetic moment: nuclei with odd number of nucleons (protons and/or neutrons), since, otherwise, the magnetic moments are canceled. Thus, although the hydrogen nucleus (^1H) has a significantly higher magnetism due to the value of the gyromagnetic constant, also different nucleus

[4]The spin is an intrinsic form of angular momentum carried by elementary particles, composite particles (hadrons), and atomic nuclei. The spin is the source of the MR signal: a nucleus with spin is magnetic.

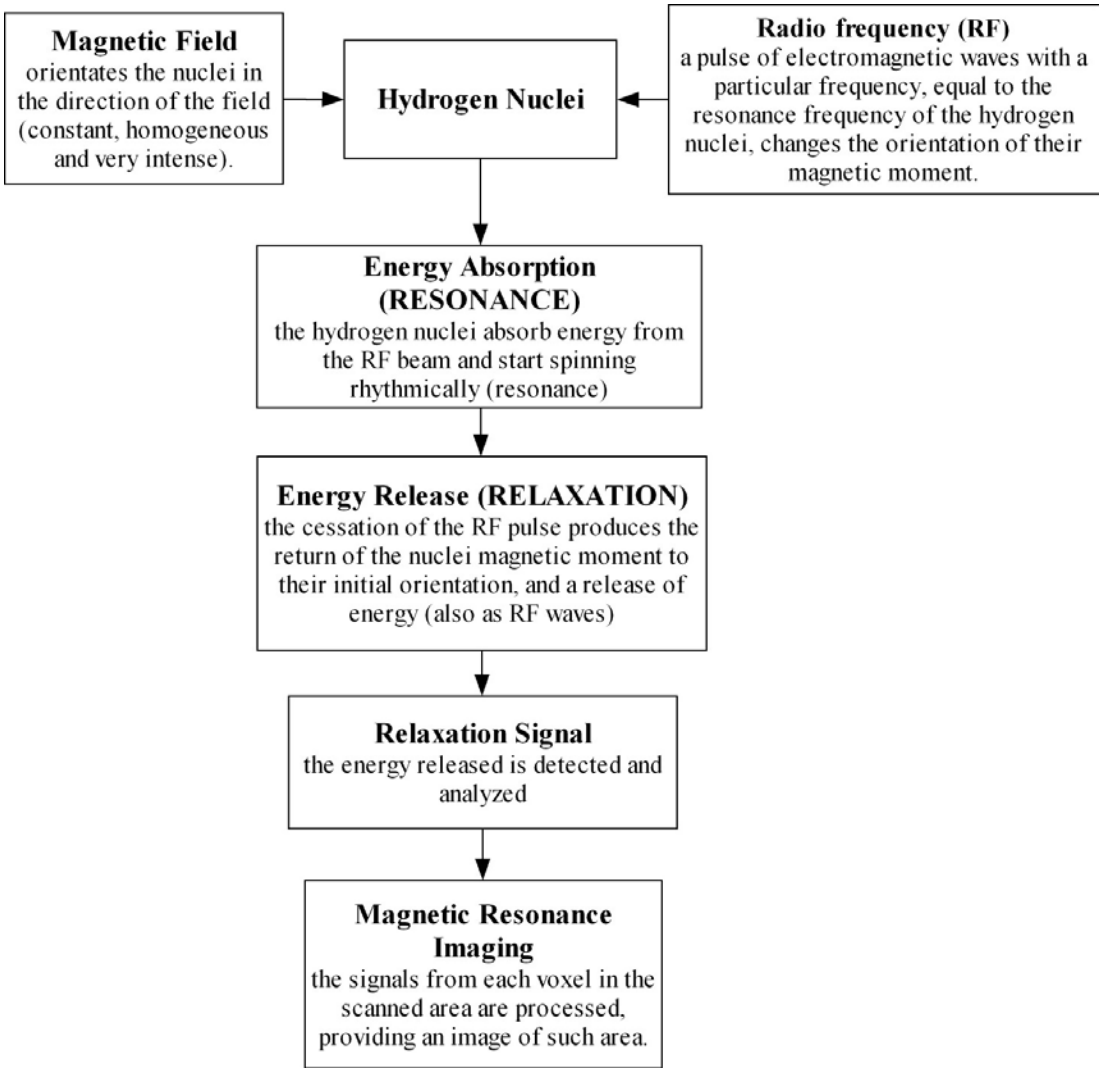


Figure 3.6: General Overview of the MRI physics fundamentals

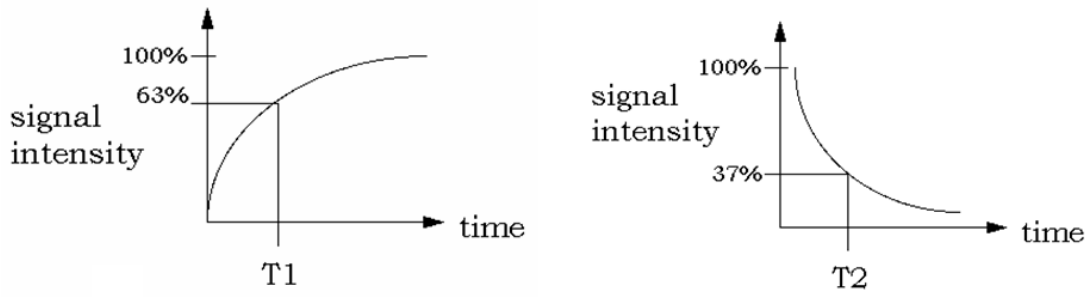


Figure 3.7: Left: T1 relaxation. Recovery of signal in the Mz direction. T1 is the time in ms for recovery of the 63% of the original M. Right: T2 relaxation. Decrease in signal in the xy plane. T2 is the time in ms for 63% of the transverse magnetization to be lost.

have great interest from the medical point of view (^{31}P or ^{23}Na).

The magnetization, that is generated when the patient is placed within a MRI machine, is oriented in the sense and direction of the external magnetic field (longitudinal magnetization, Mz). After that, a RF pulse, equal to the right Larmor frequency[5], is applied to the anatomical area under study. Such RF pulse is "received" by the protons, producing an energy transfer by "resonance" to the precessing protons[6]. During the RF pulse, the protons are all oriented in the same direction at the same time, and their vectors are added, resulting a vector oriented in the transverse direction (transverse magnetization, Mxy): therefore, the RF pulse decreases the longitudinal magnetization and establish a new transverse magnetization.

The relaxation process starts by disconnecting the RF pulse. There are two relaxation processes, related to every magnetization, which are characteristics of each tissue: T_1 (longitudinal) and T_2 (transverse). T_1 is the time for longitudinal magnetization to recover the 63% of its equilibrium state. T_2, instead, is the time necessary for the transverse magnetization to be reduced by 63% of its maximum strength (see Figure

[5]The Larmor frequency is a specific frequency of each type of nucleus and depends on the intensity of the external magnetic field. For instance, the Hydrogen Larmor frequency in a magnetic field of 1 Tesla is 42.58 MHz.

[6]The presence of the external magnetic field causes, in addition to the orientation of the moment, a rotational movement of the protons around the field called precession movement.

3.7). The RF signal emitted in the process of relaxation of the protons, that form each elementary volume, is very weak. To intensify it, consecutive RF pulses separated by pauses are used, forming a sequence that is repeated several times. The use of appropriate parameters allows one to study the relaxation times of different tissues and, therefore, to differentiate them (see Figure 3.8): echo time (TE) and repetition time (TR). TE is the time that passes since the pulse is sent until the signal is received. TR refers to the time interval between successive pulses of RF. The spin echo sequence is made up of a series of events : 90° pulse - 180° rephasing pulse at TE/2 - signal reading at TE. This series is repeated at each time interval TR, eliminating the constant inhomogeneities of the magnetic field (due to the MR equipment) and allows one to obtain the true T2 value (instead of T_2^*[7]).

Basic MRI scans include T_1-weighted MRI ($T1WI$), T_2-weighted MRI ($T2WI$) and T_2^*-weighted MRI ($T2^*WI$). T1WI is a MRI where the contrast is predominantly dependent on T_1, and can be created by using short TE and TR times in conventional spin echo sequences. $T1WI$ are very anatomic, in the sense of facilitating the morphological study of the different anatomical regions. The shorter TE, the stronger the signal obtained from a tissue, but the difference in intensity between a tissue A and B would be very small. However, differences in T_2 curves are more pronounced with a long TE, and therefore, the contrast between the different tissues is better ($T2WI$). The T2-weighted images are more physiopathological and they further facilitate the diagnosis (see Figures 3.9 and 3.10).

In chapter 7, 17 T_1-weighted brain MR images retrieved from a NMR database with their associated manual segmentations [188] were used in the experiments.

[7] T2 relaxation corresponds to inhomogeneities of molecular origin that involve phase difference in protons (spins). The sum of such inhomogeneities with the constant ones inherent to the magnetic field is called T_2^*.

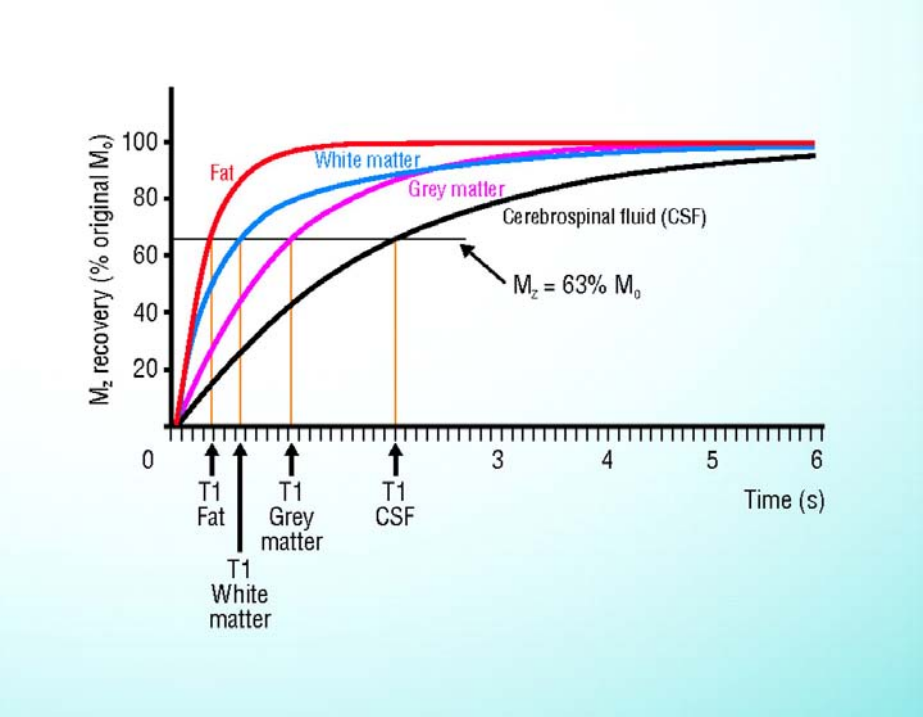


Figure 3.8: Different tissues in the body interact with RF energy in different ways. For example, fat will give up the RF energy it received during the 90 degree RF pulse very quickly (short T1 and an even shorter T2 time). Conversely, water takes longer to give up the absorbed RF energy (relax), and therefore has longer T1 and T2 times. This image has been taken from `http://mri-2010.blogspot.it/2010/10/october-lecture-notes-1-image-density.html`

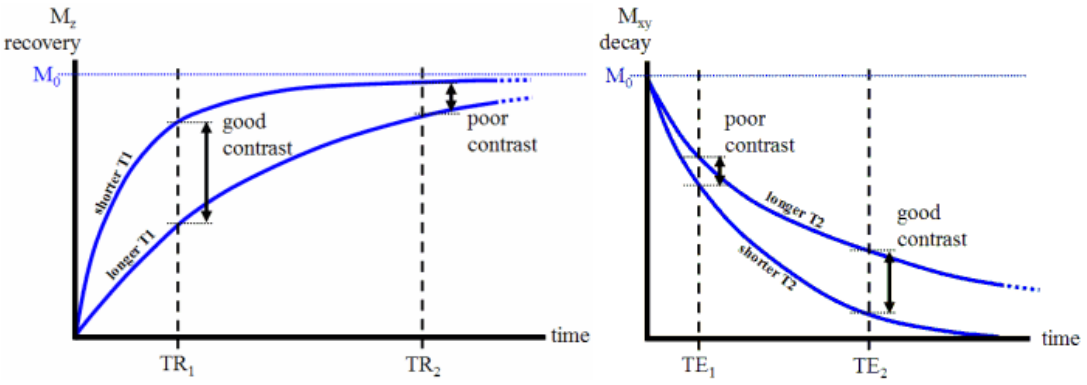


Figure 3.9: T1 and T2 decay curves. The TR enhances or minimizes the difference in signal due to differences in the T1 of tissues. The TE enhances or minimizes the difference in signal due to differences in the T2 of tissues. Source of this figure: `http://www.revisemri.com/questions/basicphysics/`

Figure 3.10: Examples of MR images. T_1-weighted (left) and T_2-weighted (right) images of brain, cervicothoracic spine and right shoulder.

Chapter 4

Medical Image Segmentation using Deformable Models and Soft Computing

It is fundamental to understand that the combination of internal and external forces in a DM determines a target function to minimize, whose minimum is theoretically located at the boundary of the object to segment. This target function can be very complex (noisy, highly-multimodal) and most classic algorithms fail at minimizing it [79, 123]. Hence, the global search capabilities of metaheuristics can be very beneficial to optimize this function. Furthermore, the automatic learning of DM parameters is also possible using these intelligent techniques [5]. In fact, automatic parameter configuration is desirable since it is known that *manual* parameter tuning is a time-consuming task and may introduce a bias in comparing an algorithm with a reference, when there is better knowledge of the algorithm under consideration with respect to the reference or, possibly, different time is available for tuning them.

In the literature several examples can be found which hybridize parametric DMs and metaheuristics. In [124] and [125], snakes are combined with an optimization procedure based on GAs. In [126] a GA evolves a population of medial-based shapes extracted from a training set, using prior shape knowledge to produce feasible deformations while

also controlling the scale and localization of these deformations. In [127] an ACM is applied to the automatic segmentation of PET images of liver, and a GA is used to find optimal parameters values for the edge detection step. In [128], the authors have described a method for interactive segmentation of 3-D medical images that relies on an edge detector and an elastic contour model for both of which the parameters are optimized by a GA. The method uses a small set of manually traced contours of the structure of interest and two cascaded modules: a nonlinear edge detector and an interpolator based on an elastic contour model. PSO has also been successfully used in conjunction with various types of DMs. Asl and Seyedin [129] apply the technique proposed in [130] using PSO instead of a GA, obtaining similar results in terms of precision but in shorter time. In [131], a multi-population PSO is used to drive an active contour model, emphasizing the capability of the model to adapt to shapes with strong concavity. With respect to standard and extended TANs, the minimization of TANs energy to segment CT images is carried out by means of GA and memetic algorithms [132], Differential Evolution [133] and SS [134].

In relation to geometric DMs, much fewer proposals of hybridization have been presented. In [135] a GA is used to perform LS curve evolution using texture and shape information to automatically segment the prostate in CT and MRI pelvic images. In [5], a GA is used to find an optimal set of parameters that characterize the LS method in CT and MRI segmentation. Finally, in [136], the initial segmentation based on the LS method is refined using swarms of intelligent agents.

MRI, due to the higher percentage of medical imaging papers that deals with it, is the modality that has most benefited from the use of MHs. This is probably because of its resolution, greater than ultrasound and molecular imaging, good contrast, low noise levels and its non-invasive nature, since it uses only magnetic fields and radio frequency waves. In relation to the intelligent techniques used, there are many successful MHs that have not been applied yet to this research field. For instance, no work has been published using the CHC binary-coded evolutionary algorithm or Ant Colony Optimization (ACO) when dealing with medical IS problems. In general terms, MHs have been focused on optimizing the parameters of the DM, avoiding problems with the initialization of the

model and local minima, and learning the system parameters automatically.

To understand the growing importance of this research field, the histogram at the top of Figure 4.1 displays the scientific production in medical IS, by executing the Scopus query:

```
"image" AND ("medical" or "biomedical")
AND ("segmentation" OR "localization" OR "partition")
```

Regarding the scientific production about medical IS using MHs, shown at the bottom of Figure 4.1, the following query was run in Scopus:

```
"image" AND ("medical" or "biomedical")
AND ("segmentation" OR "localization" OR "partition")
AND ("differential evolution" OR "swarm"
OR "chc" OR "evolutionary"
OR "genetic algorithm" OR "cma-es"
OR "PSO" OR "annealing" OR "tabu search"
OR "metaheuristic")
```

Finally, in order to know the volume of publications on evolutionary medical IS using DMs, the same query was executed adding a reference to terms related to DM (*AND ("deformable model" OR "template" OR "topological" OR "statistical shape" OR "active models" OR "active nets")*). The result included 76 publications, and an h-index of 14, related to this topic since 1995 (see Figure 4.2)[1].

Looking at the data globally, one can say that both medical IS and MHs-based medical IS are fields of growing interest, where publications and citations are dramatically increasing. Given the extremely beneficial properties of MHs, a more intense application to medical imaging problems seems inevitable, even more if the use of

[1]A careful revision of the literature showed the existence of, at least, 70 more papers (mainly conference papers or publications in which 'strange' kinds of DMs were used, so they did not appear in the results of the queries). The queries were also run in the ISI Web of Knowledge but the number of resulting papers was even less.

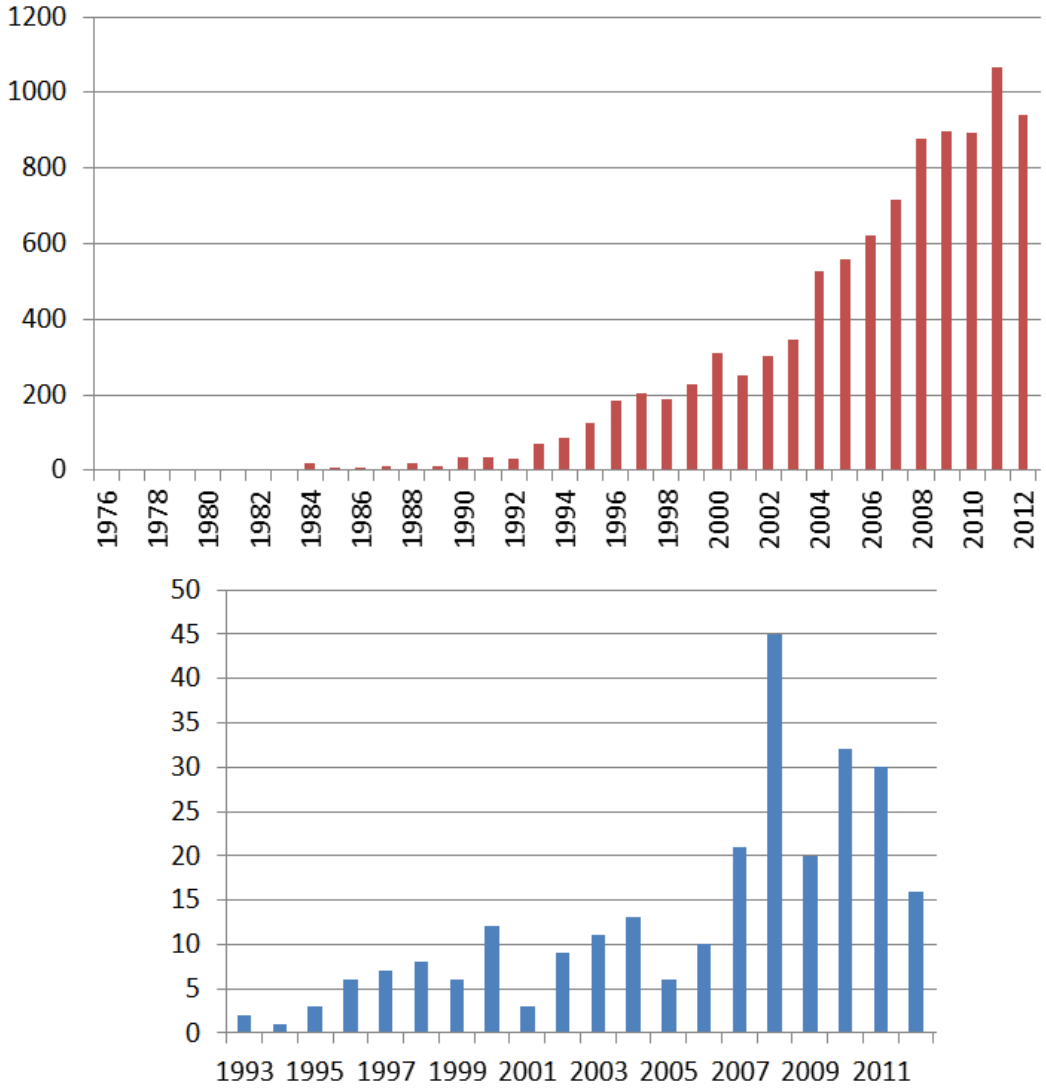


Figure 4.1: Scientific production related to medical IS (top) and medical IS using MHs (down). The queries, run in Scopus on the 10th of July (2013), displayed 9942 and 269 papers with a growing tendency.

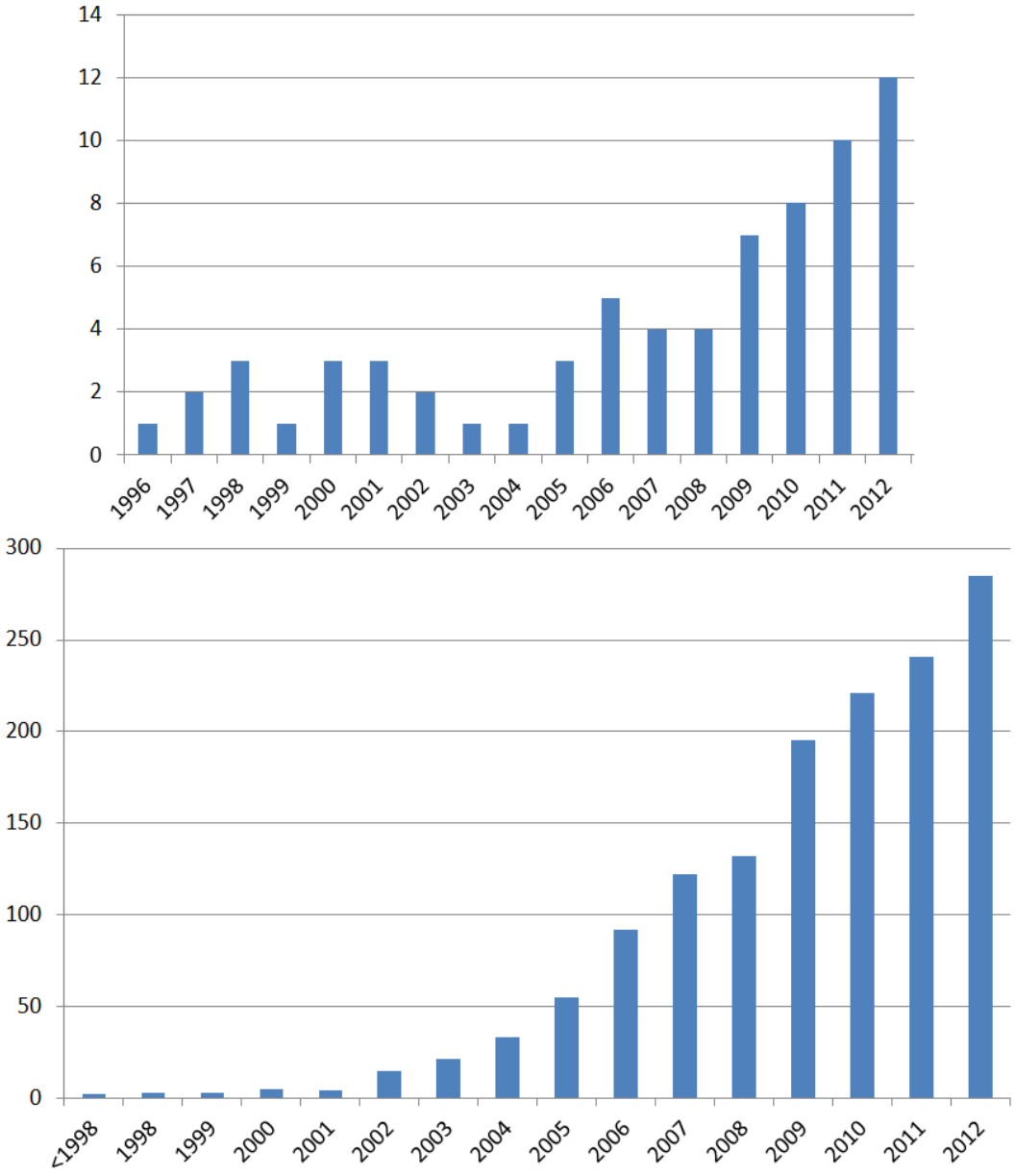


Figure 4.2: Number of publications (top) and citations (down) related to medical IS using DMs and MHs. The queries, run in Scopus on the 10th of July (2013), displayed 76 papers and 1429 citations also with a growing tendency.

DMs is taken into account (a standard within the image processing). Also, it seems a safe bet to explore the possibilities their combination offer, trying new evolutionary approaches or improving the existing ones, reducing the execution time of algorithms (progressively viable with the increase of CPU/GPU performances), and introducing novel approaches to medical image analysis fully based on SC algorithms or in which SC algorithms play an essential role [137].

Since the main kinds of DMs used in this thesis have been ASMs and LS, the following two subsections summarize all publications related to image segmentation using these two paradigms and MHs (including our own proposals published in different journals and conferences).

4.1 Statistical Shape Models

One of the most populated categories is the one related to DMs that use some sort of statistical information about the objects to segment (instead of using only generic constraints, like in ACMs). Such DMs can be created by running some mathematical transform (like PCA) over a number of training shapes or including more information specific to the deformation shape limits. In this case, ASMs, AAMs, DTs and similar approaches, even if the terminology used in the papers does not explicitly refer to such categories, are here globally called Statistical Shape Models (SSMs).

The potentiality of this kind of DM was discovered rather early. In the ‘90s five papers were already presented, of which two are journal publications. From the seminal papers by Hill et al. [138, 139], GAs have been used several times, alone or in hybrid approaches, as optimizer to overcome the problems of classic methods [123, 126, 140–153]: DM initialization and existence of local optima.

In terms of applications, there is a broad variety of topics, ranging from face/body recognition [141, 143–145, 154–157] and car/road-sign localization [3, 142, 158–160] to medical image segmentation [123, 126, 138–140, 147–153, 157, 161–168], or mobile robot navigation [146].

Table 4.1: Statistical Shape Models (Active Shape Models, Active Appearance Models, Deformable Templates). From left to right: author name, reference to the paper, DM name, dimensions (2D/3D), type of metaheuristic, type of imaging technique, and year of publication. $*$ refers to ISI indexed journals.

Author	Reference	DM Name	Dims	Metaheuristic	Image Modality	Year
Hill et al.	[138]	Flexible Templates (like ASM)	2D	GA	Medical (ultrasound) Heart Left Ventricle 20 and 25 images	1992
	[139] / $*$			GA vs SA		
Jolly et al.	[3] / $*$	DT	2D	SA (Metropolis alg.)	393 image seq. moving vehicles	1996
Rueckert & Burger	[169] [161, 162]	Geometrically DT	4D	SA + ICM	1 MRI seq.	1996 1997
Sum et al.	[154]	ASM	2D	Greedy local search	1320 lip images	2001
Mignotte et al.	[140] / $*$	DT / ASM	2D	GA + steepest ascent	echographic seq. (50 frames) & synthetic images	2001
Pitiot et al.	[163] / $*$	SSM	2D	Evolutionary SA	Medical (MRI) 3 brain structures T1-weighted 94 images	2002
Lalonde & Gagnon	[164]	Geometrically DT	2D	Variable Neighborhood Search (VNS) using SA as local search	optic disc in 2 ophthalmic images	2002
Betrouni et al.	[165]	AAM	2D	SA	30 abdominal ultrasound images	2004
Karungaru et al.	[141]	DT	2D	real-coded GA	54 faces	2004
Vavilin & Jo	[142]	DT	2D	GA	95 images with 119 signs	2006
Heimann et al.	[166]	SSM + Deformable Surface	3D	ES	Medical (CT) liver in 54 CT volumes	2007
Sattar et al.	[143, 144] [145]	AAM	2D/3D	GA vs Nelder-mead simplex vs NSGA-II Gradient-based GA	face images 32 subjects 1405 images real & synthetic images	2008
Mata et al.	[146] / $*$	DT	2D	GA	real-time video seq.	2008
Yin & Collins	[158]	DT/ASM	2D	SA	aerial video motorcycle zebrafish bladder MRI human walking seq.	2009
Chen et al.	[147]	ASM	2D	GA	20 lateral knee X-ray images	2009
Szilagyi et al.	[148]	AA motion M	4D	multipopulation GA + Nelder-Mead simplex	ultrasound seq. Number not specified	2009
Takahashi et al.	[149]	ASM	2D	GA	5 CT images cardiac region	2010
Colutto et al.	[167] / $*$	3-D medial axis shape model (3-D-MReps)	3D	CMA-ES (vs PSO and DE)	1 3D MRI scan & synthetic data	2010
Ben Fredj et al.	[168]	DT	2D/3D	GA	synthetic heart 3D MRI	2011
McIntosh et al.	[126, 150] / $*$	Medial-based shape	2D	GA	Medical (MRI) corpus callosum 50 images	2010 2012
Mussi et al.	[155, 159] [156]	DT	2D	PSO	Road signs human body pose	2010 2012
Ugolotti et al.	[160]			DE vs PSO	Road signs	2012
Liu te al.	[151]	AAM	2D	GA	200 tongue images	2012
Mesejo et al.	[123] / $*$ [152]	ASM	2D	DE vs PSO vs LM vs SA vs SS vs GA	Medical (histological) brain (hippocampus) 320 real images + 20 synthetic images + 4 human video-seq.	2011 2012 2013
Ugolotti et al.	[157] / $*$ [153]					

DM representation and encoding

In general, the parameters to be optimized by the metaheuristic are the coordinates of the parametric DM [123] or the coefficients of a parameterized curve (like a B-Spline) [163]. Such coordinates are usually described in a cartesian system but polar coordinates can also be found [153].

In [138, 139], ten parameters control the shape and transformations of the model: six shape parameters and four transformation parameters (translationX, translationY, rotation and scale). In the former, a point in the search space is encoded as a binary chromosome while, in the latter, gray-code binary integers are used. In [147, 149], a similar approach is carried out in terms of representation/encoding.

An interesting aspect of [163] is that the number of control points dynamically changes. The larger the number of control points in the deformable template, the better the local fidelity of the match is likely to be and the higher the computational cost (inducing many local minima and complicating the search space). The system determines the segment (whose shape is parameterized by four control points) of the B-spline that achieves the lowest match by computing the partial match of each segment over the diffused edge image (to avoid generating a large number of null matches). Then, that segment is locally refined, inserting new control points.

In [126, 150], medial-axis-based 2D shape representations describe the object's shapes in terms of an axis positioned in the middle of the object along with thickness values assigned to each point on the axis that describe the shape of the boundary. Each of the N medial nodes (being nodes 1 and N the terminal nodes) have associated 4 scalar values (length, orientation, and left and right thickness profiles), and the anatomical structure to be segmented is reconstructed from these 4 medial profiles and a set of affine parameters that describe the objects pose and scale (position, scale and base angle). Each shape is a chromosome with genes encoding affine and statistical shape deformation parameters. The affine transformation parameters are those encoding the global rotation, scale, and position of the shape, while the statistical shape deformation parameters are those representing weights of the principal components, as obtained from

the hierarchical regional principle component analysis, for a particular deformation, location, and scale.

Also, a sort of medial-based representation can be found in [123, 152, 153, 155–157, 159, 160] where, even comprising very different examples, the common factor is the use of 'skeleton' representations, that are employed to compose a simple model of the object to localize, segment or track. Obviously, this kind of minimal representation has many advantages, like a low computational cost allowing for real-time execution, the simplification of the operations with the coordinates and their constraints, or the possibility of performing a fast initial coarse localization and, later, a fine segmentation. In particular, in [123, 152, 153], the encoding includes lengths and angles from one point in the model to another. In [155–157], an articulated 3D model of a human body is matched against a set of images of a human performing some action, taken from different perspectives, to estimate the subject's posture in space. Given the significant computational burden imposed by this approach, PSO and DE were implemented as parallel algorithms within the nVIDIA^{TM}CUDA computing architecture. In all these cases, real-valued encoding was used by the metaheuristic.

A classic representation of the model can be found in [165, 166], where the DM is represented by a point distribution model (PDM), i.e. a dense collection of landmark points on the surface of the object, and each training shape is described by a single vector of concatenated landmark coordinates. In some approaches, the SSM is created by means of PCA, but there are many examples where the limits for the deformation, i.e. the feasible shape deformations, have been simply obtained by extracting the average and standard deviation of the positions in the parametric examples in the training set [123]. In fact, PCA can be applied to different image features, like in [143–145], where PCA is applied not only to the shape but to the texture, to obtain shape and texture parameters accounting for 95% of the variation to perform face alignment application in embedded systems.

In [3], the authors define a polygonal template to characterize a general model of a vehicle and derive a prior probability density function to constrain the template to be deformed within a set of allowed shapes. They propose a bayesian formulation which

combines motion information and edge directionality to ensure that the deformable template is contained within the moving areas in the image and its boundary coincides with strong edges with the same orientation in the image.

Another bayesian formulation is presented in [161, 162] and solved, as in the previous case, using SA. The authors present an energy-minimization framework based on geometrically deformable template (GDT) [170], whose degree of deformation, from its equilibrium shape, is measured by a penalty function associated with the mapping between the two images. A GDT consists of a set of vertices describing the equilibrium shape (the undeformed prototype shape), a set of vertices representing the deformed prototype shape (result of the external forces over the model), and a penalty function which measures the amount of deformation of the template with respect to the equilibrium shape. This penalty function is invariant to scaling, rotating, and translation of the template. Such a model can incorporate not only information about the mean shape and the variability of objects, but also information about the mean location, orientation and size of objects, and their variability, being able to segment several objects simultaneously. Therefore, to solve the segmentation problem it is necessary to estimate the optimal values for the following parameters: translation, scaling, rotation, and N parameters for the non-affine deformation. The affine ones are calculated by an exhaustive search method (Iterated Conditional Modes). The non-affine ones are estimated using SA. In [169], a multiscale approach to GDT to contour fitting is used: the segmentation algorithm starts by constructing a linear scale-space of an image through convolution of the original image with a Gaussian kernel at different scale levels, where the scale corresponds to the standard deviation of the Gaussian kernel.

A cubic B-spline shape representation involving a binary encoding is used in [140]. To take into account the natural variability of the object under consideration, the authors introduce a set of admissible affine transformations, ensuring a first crude registration of the shape, and a set of non-affine local and global transformations. In [146], a three-stage algorithm is used. First, regions of interest (ROIs) are extracted. Then, the extracted ROIs are used to initialize a GA for the landmark search through the image. Each individual of this GA encodes a deformable model, and the fitness of an individual

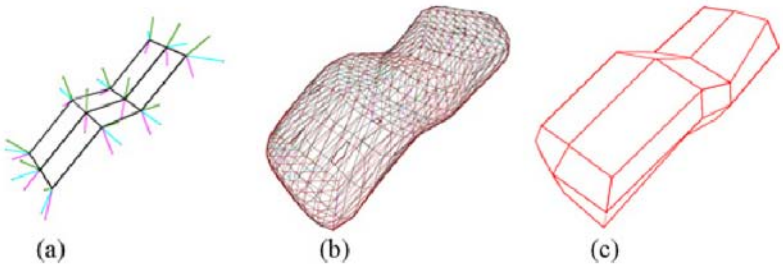


Figure 4.3: (a) Instance of 3-D-MRep shape space, (b) its implied boundary, and (c) initial interpolation mesh that is used to construct the surface. Note that the interpolation mesh is an irregular quadrilateral mesh, since it has vertices with valence 3 at each of the 4 corners of the figure.

is a measure of the matching between the deformed model it encodes and the landmark searched for. Finally, if a landmark is found, symbols are extracted and identified with a classic backpropagation neural network. Thus, a GA is used to confirm or reject the ROI hypothesis. Each individual's genome is made up of five variables: the deformed model's cartesian coordinates (x, y) in the image, its horizontal and vertical size in pixels, and a measure of its vertical perspective distortion.

In [167], as a shape model, they use a parametric 3-D medial axis representation (3-D-MRep). The parameter domain of 3-D-MReps constitutes a finite-dimensional Riemannian manifold, and each element of this shape space defines a surface. 3-D-MRep (see Figure 4.3) uses a discretization of the continuous medial axis of an object instead of boundary representations.

Every instance of an MRep in three dimensions is represented by a regular $n_1 \times n_2$ quadrilateral mesh, and the $n_1 n_2$ atoms centered on the mesh vertices. The following information is provided for every atom of a shape instance:

- the position $x \in \mathbb{R}^3$ of the atom,
- the radius $r \in \mathbb{R}^+$ of the atom,
- two boundary vectors $b_1, b_2 \in S^2$, pointing to the two implied boundary points of the surface. For border atoms (i.e., atoms on the boundary of the mesh), an

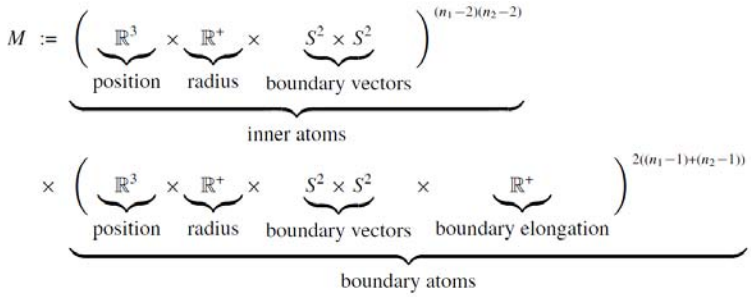


Figure 4.4: Shape space of 3-D-MReps.

additional boundary vector $b \in S^2$ is defined as the bisector of the two given boundary vectors.

- a parameter for border atoms, called $\eta \in \mathbb{R}^+$ specifying the elongation factor along the additional boundary vector.

Each of the components of M (see Figure 4.4) is a Riemannian manifold, and this product space is called the shape space of 3-D-MReps.

Optimization procedure

In terms of genetic operators, it is important to highlight that, since these approaches take advantage of a 'preliminary statistical study' of the objects to segment, in many cases there are no special operators to manage the constraints of the models. In general terms, this is because the SSMs are usually already constrained to a particular region of the search space depending on a training stage. The evolutionary operators are not usually designed ad-hoc and they do not present a great level of sophistication because of this reason.

In [163], a Hybrid EA inspired by Guided Evolutionary Simulated Annealing (GESA)[171] is used. The authors consider a population of individuals, each of which consists of a pose and shape parameter (with scale, rotation and translation). The population is made up of f families: each family F^i consists of a parent P^i and c_i children $C^{i,j}$. The pose parameter of each of the parents is selected at random to cover

the entire input image, and is projected onto the shape space in order to adjust the pose and shape parameters of the template to the required deformation while insuring that the deformed template be a valid instance of the shape model (this seems to be the way they implement the restrictions). To favor the children whose shapes are closer to the first eigenmodes, the temperature of the Boltzman distribution that controls the acceptance number is varied. At a local level, the children of the same family compete with one another to generate the parent for the next generation. At a second level, there is a competition between the families themselves since the number of children allocated to each family depends on the combined fitness of all the children, and is biased toward the first eigenmodes to favor the most likely shapes. The number of children actually reflects the interest in a given area of the search space: the better candidate solutions in a given area, the more attention we devote to it. Therefore, the entire algorithm can be viewed as parallel SA with competition.

Even if a quite simple GA is used, in [138, 139], the authors compare SA with GA in optimizing their models. As selection strategy they use the Remainder Stochastic Independent Sampling (RSIS) algorithm, and the initial population from which the search progresses is purely random. Firstly, preliminary experiments showed that GA outperforms SA in all cases in terms of minimization error. Secondly, and applying only GA, the results obtained by this method were compared with the manual segmentation of an expert. Niching was also applied to avoid premature convergence to suboptimal solutions: the fitness of an individual is weighted by the number of neighbours (the more the neighbours of an individual the worse its fitness value), the size of the population is increased to prevent extinction caused by sampling errors, and a restricted mating strategy is implemented in order to promote speciation, i.e., prefering neighbours to distant individuals for crossover. In [148], a robust solution of such an optimization problem implies a multi-population GA approach, which creates subpopulations within the niches defined by multiple potential optima. The employment of multiple groups of individuals favours population diversity as subpopulations that occupy different niches do not need to outperform each other in order to propagate, resulting in a proper exploration of the search space. Subpopulations only interact by means of a migration process, where the best individuals from one subpopulation are copied

into another subpopulation, replacing the worst individuals in the destination search niche. Intensification is achieved by allocating a separate portion of the search space to each subpopulation. After convergence of the global search, a Nelder-Mead simplex algorithm is further employed starting from the best solution.

In [123, 152, 153], the restrictions, in terms of deformation, are managed by the polar representation and the limits for the deformation calculated from the training set, thus the development of ad-hoc operators is not necessary, and the procedures used in GA, SS, DE, SA and PSO are the conventional ones.

A global search based on an EA is employed in [166] to detect suitable initial parameters for the model, which are subsequently optimized by a local search (ES is run to find a rough initialization in a strongly down-sampled version of the image). After that, a deformable mesh with the same topology as the SSM is used for the final segmentation. The solutions are stored as real-valued vectors, there is no cross-over operator, and solutions are modified by adding a random vector from a multivariate zero-mean gaussian distribution (gaussian mutation). After initialization, the deformable model used for the final segmentation is defined as a triangular mesh, having the same topology as the SSM. Its evolution is controlled by the Lagrangian equation of motion, using a directed graph and a min-cut/max-flow algorithm for energy minimization.

In [144], a GA searches faces globally whereas gradient descent helps GA to search a face locally. In other words, the exploitation properties of gradient descent and the exploration properties of GA are combined into an effective optimization system. For this reason the authors propose a new gradient operator for the GA, which functions in conjunction with the mutation operator already available in the GA. In [145], the authors present a face alignment algorithm in multiple images based on AAMs and Pareto multi-objective genetic optimization using NSGA-II. This multiobjective EA is used to optimize the appearance and pose parameters. The target is to find the best possible values of these parameters causing the minimum error pixelwise, between the model and the query images of both cameras.

In [151], the shape model is adjusted to fit the image in an iterative procedure, which starts from the mean shape. In each iteration, the position of the points is slightly

modified by examining the region around each current model point to find the best match. The model parameters are then updated, until convergence, to fit the shape model to the new location of points. This procedure is useful in cases where the mean shape is a good aproximation for the final segmentation, and a GA is used more as a local optimizer than as a global-search method.

The energy function to segment moving vehicles [3] has many local minima, and the authors have chosen a SA procedure based on the Metropolis algorithm to solve the minimization problem. Such an algorithm is a SA procedure that, using a geometric temperature schedule, minimizes the energy function $E(\theta, Z)$ by constructing a sequence of template deformations starting from a prototype template. In each epoch, the Metropolis algorithm is started with an initial prototype template belonging to the vehicle class being considered and it iterates while constraining the deformations to remain in the same class. The authors examine the results of all five epochs, each epoch corresponding to a particular vehicle class (sedan, pickup truck, hatchback, station wagon, van), and pick the deformation parameters whose energy is smallest (see Figure 4.5). This approach has two advantages: 1) the five epochs can be executed in parallel, and 2) in addition to segmenting the vehicle in the lane closest to the camera, the algorithm is also able to classify the vehicle into one of five classes.

In [140], when segmenting ecographic sequences, 100 individuals (or 100 ellipses: circle + affine transforms) are initially randomly generated in the image (first generation or first step of the genetic search). Each individual represents a candidate solution (a candidate contour) for the optimization problem. The genetic procedure then acts in a iterative way by creating a new population with genetic operators. At each generation, 5% of the best individuals are selected for the hybridization with the local optimization technique (gradient ascent).

The segmentation of MR images in [161, 162] requires the estimation of the optimal values for the following three parameters: translation, scaling, rotation, and N parameters for the non-affine deformation. The affine parameters are calculated by an exhaustive search method (Iterated Conditional Modes), while the non-affine ones are estimated using SA.

In [164] two modifications to the GDT model are proposed to perform the segmentation of the optic disc in ophthalmic images. GDTs are based on the use of thin-plate splines and have the appealing property of being invariant under affine transformations. However, the proposed search method is computationally costly because it relies on SA. As well, such an invariance to affine transformations may cause the search algorithm to retain invalid solutions (e.g. ellipses when searching for circles). This paper addresses these concerns by exploring the use of a metaheuristic called Variable Neighborhood Search, that treats simulated annealing as a local search tool, and also by redefining the shape energy so that affine transformations are taken into account to improve the serch quality.

Finally, in [167], the main challenge of the segmentation of MR volumes using CMA-ES is to determine the mean value (on the manifold M) for the next iteration and to transport the covariance to the new location on the manifold. For a fair comparison, the authors tried to choose the parameters as recommended in the relevant literature and modified them only if this led to a significant improvement of the results.

Target Function

The most common approach in these works is to maximize the overlap of the DM with the object of interest, taking into account intensities, boundaries, or textures. The prior knowledge about every particular case can be very easily included in the fitness function, as happens, for example, in [123]: since the hippocampus is slightly darker than the immediately surrounding area, this information is encoded in the fitness function through the maximization of the difference in intensity between the inner model points (ideally corresponding to the hippocampus) and the outer model points (ideally corresponding to the external, and clearer, regions surronding the hippocampus). Since the candidate polar coordinates are constrained within a preselected range of values computed from the training set, the development of complex ad-hoc operators and fitness functions is not necessary.

In [155–157], the body model consists of two layers, the skeleton and the skin.

The skeleton layer is defined as a set of homogeneous transformation matrices which encode the information about the position and orientation of every joint with respect to its parent joint in a kinematic tree hierarchy. The matching function compares the silhouettes extracted from the original images to the silhouettes generated by the model in its candidate pose.

In [143–145], when tackling the face recognition problem, each chromosome corresponds to a 3D AAM. Each shape is deformed, rotated and translated according to the appearance, rotational and translational parameters in the chromosome, respectively. This deformed shape is placed on the test image to warp the face into the mean frontal shape. The error of the target function is then calculated pixelwise between this warped image and the frontal view image of the database obtained by the appearance parameters of each chromosome. The newly-born children are checked to verify wether they are still inside any of the Gaussian clusters identified in the face space; if not, they are initialized to the nearest gaussian distribution.

The cost function in [154] is defined following a region-based approach. The RGB representation of a lip image is transformed into the uniform CIELAB and CIELUV color spaces. Color feature vectors consisting of color features (a, b, u, v, hue_{ab}, hue_{uv}, $chroma_{uv}$) are then generated for each pixel. Applying fuzzy clustering to the color feature vectors, a probability map is obtained, such that a pixel with higher probability value is more likely to be within the lip region.

In [3], the segmentation problem is stated in a Bayesian framework (probabilistic DM) and is reduced to an energy minimization problem: 1) a template defined by a set of deformation parameters which denotes a prototype of the objects present in an image, 2) a prior probability density function which specifies what deformations of the prototype template are more likely to be present in the target image, 3) a probability density function which discriminates between different hypothesized deformations in terms of how “close” they are to the actual objects in the image, and 4) an algorithm for finding the template which ”best” characterizes the objects in the target image (let us call it Z); more specifically, the deformation parameters θ which maximizes the a posteriori probability $p(\theta|Z)$. The probability density function is modeled by a Gibbs

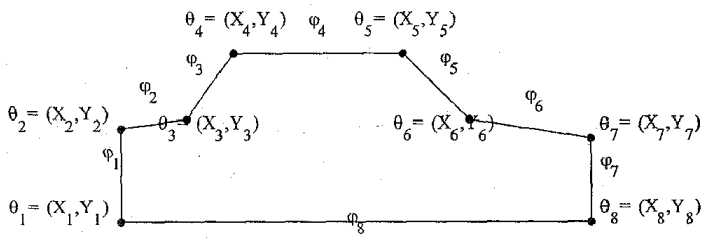


Figure 4.5: Deformable template defining a generic model of a vehicle from [3].

distribution whose exponent comprises two terms. The first term is a function which derives from the motion of the vehicle of interest. It attains its maximum value when the deformed template encompasses only pixels that are moving. The second term is a directional edge-based function. It attains its maximum value when the contours of the deformed template coincide with underlying image edges that have a strong gradient magnitude and whose gradient orientation is perpendicular to the contour.

In [140] the main idea, as in other papers in this section, is to find the optimal deformation parameter vector to fit the original prototype template to the image. The fitness function in this case is composed by a likelihood energy term that is minimum when the deformed template delimits exactly two homogeneous regions with gray level distributions corresponding to blood and muscle, and a prior energy term that penalises the deviation of the deformed template from the original prototype.

A model of the heart, in [161, 162], is composed of 3 different objects which are represented by the endocardial contours of the left ventricle and right verticle as well as the epicardium. Authors use GDT, whose deformation is controlled by an energy penalty function that measures the degree of non-affine deformation of the template from its equilibrium shape which represents the a-priori information. At the same time, the penalty function does not penalize affine transformations. To do so, bending energy of a thin-plate spline (TPS) mapping is used. The internal energy measures the amount of non-affine transformation of the template with respect to the equilibrium shape. The external energy is a form of potential that attracts the template towards image features such as edges.

In [167], the authors investigate two appoaches to detecting objects in voxel images

based on the 3-D-MRep shape models:

- A Simplified Mumford-Shah Functional: used to segment synthetic 3D images.

  The authors search for sets of voxels representing volumes, which are characterized by a significant difference between the mean image intensity inside the volume and the mean intensity of the background. They want to include, in the reconstruction energy, statistical information about the expected solution. Assume training shapes $p_1, ..., p_S \in M$, being M a Riemannian manifold. The approximated principal geodesic analysis (aPGA) of this data is given by its mean $\mu \in M$ and covariance Σ. After that, the authors derive the Mahalanobis regularization $d_{M,\mu,\Sigma}$ based on the aPGA. Combining this regularization with the Simplified Mumford–Shah Functional leads to the definition of the regularization functional: $I_\alpha^{SMS}(p) = I(p) + \alpha d^2_{M,\mu,Sigma}(\mu, p)$, being I a mapping of a shape p to the simplified Mumford-Shah energy of its boundary $\psi(p)$, and $I(p) = I'(\psi(p))$.

- Edge-Based Segmentation: used to segment the cerebellum.

  If the objects of interest have nearly the same intensity as the surrounding volume, region-based segmentation cannot be applied. To overcome this problem, an edge-based segmentation method is introduced. In this case, the first term of the functional forces a Jordan submanifold to be at locations where the gradient of the voxel data is high, and the second term penalizes the surface area of such a submanifold. In this case, the functional for the 3-D-MReps shape model is $I_\beta^{Snakes}(p) = I'_\beta(\psi(p))$

Therefore, the main idea here is to minimize I (equivalent to I_α^{SMS} when $\alpha = 0$) or I_β^{Snakes} depending on the case.

Critical Discussion

There are a number of strengths and weaknesses in the works here presented. With regard to the former, MHs give flexibility in the DM design, since it allows one to

ignore the nature of the objective function (differentiability, convexity) and to use the method as a black-box optimizer. Also, MHs are used to move the DM as well as to optimize its parameters, although this last option is not too common in SSMs. In relation to weaknesses, we could mention the following:

- In general terms, the papers published so far do not show a good justification of the use of MHs, and it appears that, in some cases, a local optimizer or a classical method would have been good enough.
- Comparisons with traditional methods are rare, even if they would be really useful and illuminating: in relation to the previous point, those comparisons could demonstrate the interest of introducing these sophisticated techniques instead of using classic/traditional ones.
- Also, in the vast majority of the papers, there is not a detailed description and presentation of the parameters and operators used, and no information about the tuning procedure followed is provided.

Among the most interesting works of this section, we could cite the four Bayesian approaches introducing probabilistic DMs [3, 140, 161, 162], the paper that presents the DM online definition for mobile robot navigation [146][2], the approaches presented in [163, 167] focused on the segmentation of objects using sophisticated MHs (a generalization of the CMA-ES on vector spaces to Riemannian manifolds, and an evolutionary version of SA), and the appropriate use of GAs and DMs in the resolution of a difficult biomedical problem [126].

4.2 Level Set Methods

The first issue that should be noted in this section is the shortage of scientific works where geometric/implicit approaches are used in conjunction with MHs. Only 16

[2]Instead of receiving the detailed model definition from the user, the algorithm extracts and learns the information from each object automatically.

papers have been found on this topic, four of which apply practically the same basic design [4, 135, 172, 173] while, in other cases, the use of metaheuristics has not been adequately justified or explained [174–176]. Generally, classic optimization methods like gradient descent have been (and are being) used. However, in some cases, like GAC, such techniques have been shown to be ineffective and likely to produce suboptimal solutions. In any case, these approximate solutions are acceptable in certain scenarios, where extreme accuracy is not required, while, on the other hand, many DMs are based on partial differential equations, which can be solved by well-consolidated numerical methods. However, MHs can be tremendously useful to learn the parameters of the model [2, 5, 177–179], to refine the results obtained by the geometric approach [136], to obtain the initial contour and/or the prior information to be used by the LS method [2, 175, 180] or, directly, to guide the optimization process avoiding local minima [4, 135, 173, 174, 181]. Also, an important advantage of using MHs is that they can optimize the LS function without the need to compute derivatives, thereby permiting a straightforward introduction of new curve-evolution terms [2].

Two of the most interesting and adventurous approaches can be found in [2] and [181]. The former describes a hybrid geometric DM, combining region- and edge-based information with the prior shape knowledge introduced using deformable registration. Such an approach implies the learning of the LS parameters by means of a GA, and the use of SS to derive the shape prior. In the latter, the computation of a global minimum of the CV model (multiphase piecewise constant Mumford-Shah model) is performed combining gradient-based steps with a stochastic optimization phase hopping from a local minimum (basin) to another, while the computational cost is alleviated using a multiresolution approach.

Another very interesting research work is presented in [182] (a continuation of the work presented in [176]), where the authors take the viewpoint that region information can be introduced as extra constraints within the contour energy-minimization framework. With this in mind, the contour energy minimization problem is formulated as a search for a minimum-energy contour with its interior satisfying a region-based constraint. Such a constraint can be any function characterizing the contour interior

structure, but in this particular case they use region homogeneity as criterion. The introduction of the constraint is aimed at limiting the search space of contours, focusing on the ones with desirable interior properties; and an ES is used to solve this constrained contour energy minimization problem.

Table 4.2: Level Set Method. From left to right: author, reference to the paper under study, dimensions (2D/3D) of the images segmented, type of MH, imaging technique, and year of publication. $*$ refers to ISI indexed journals.

Author	Reference	Dims.	Metaheuristic	Image Modality	Year
Wang et al.	[176]	2D	ES	cell images	2002
	[182] / $*$			MRI and CT MRI, CT, ultrasound, histology	2004
Xiao et al.	[175]	2D	GA	Marrow Cells Mammograms	2005
Ghosh et al.	[4, 135]	2D-3D	GA	prostate CT and MRI	2006
Afifi et al.	[172]	2D	PSO	liver in abdominal CT	2010
Mesejo et al.	[173]	2D	DE vs PSO vs real-coded GA	Histological Images	2013
Kan et al.	[174]	2D	PSO	tree images	2007
Iakovidis et al.	[177] / $*$	2D	GA	Ultrasound images	2007
Law et al.	[181] / $*$	2D	Basin Hopping vs Gradient Descent vs SA	Natural scenes Brain MRI Zebrafish Image breast cancer cell image	2008
Heydarian et al.	[5] / $*$	2D	dynamic GA	Medical (CT and MRI) CT: 2 kidney + 2 lung MRI: 2 kidney	2009
Oliveira et al.	[178]	2D	GA	liver CT	2010
	[179] / $*$				2011
Feltell & Li	[136]	3D	swarm agents	Medical (MRI) 1 synth. brain seq. (181 img)	2010
Mesejo et al.	[2] / $*$	2D	SS and GA	microscopy, MRI and CT	2013
Mesejo & Cagnoni	[180]	2D	DE	Histological Images	2013

DM representation and encoding

As usual in most of the papers reviewed here, GAs have been used in a large percentage of works (9 out of 16). In some cases, they rely on a binary encoding [5, 177], when the features of the problem may suggest using a real number representation, or they use operators which are not particularly well-adapted to the nature of the chromosomes, like in [4, 135] where, despite having a real-valued genome, the single-point crossover is used (see Section 2.5).

Weight EigenShape_1	Weight EigenShape_2	Weight EigenShape_3	Weight EigenShape_4	Weight EigenShape_5	Weight EigenShape_6	Translation X	Translation Y	Scaling Factor	RotationAngle
λ	ν	ϵ	τ	κ	ρ	σ			

Figure 4.6: Two GA encoding examples using LS. In the upper row, an example of a GA chromosome in an optimization problem [4]; in the lower, a GA chromosome used in the parameter tuning approach [5] (in this case, the LS parameters to be learned are derived from Li et al. [6]).

In the works of Ghosh et al. [4, 135], Afifi et al. [172], and Mesejo et al. [173] the parameters to optimize are the coefficients of a linear combination of the eigenshapes and, in [4, 135, 172], also the pose of the object to segment (i.e. the parameters corresponding to an affine transformation). In other cases, for example, the parameters represent the location (in polar coordinates) of the points of a parametric model that is used later as initial boundary for the LS [180], the weights for the different terms in the LS equation [2, 177], or the parameters needed to extract the prior knowledge [2, 175].

In [136], the segmentation problem is seen as a fuzzy voxel classification along the LS interface, but the presence of anisotropic voxel intensities within the input image causes an unacceptable number of incorrectly classified voxels. To solve this problem, after the user identifies the area of interest, intelligent semiautonomous agents can move across the zero-level surface modifying it: the LS representation allows the authors to locate the surface in space and calculate the normal to the surface at any point very easily. Therefore, the representation used by the swarms of intelligente agents is the same as the one used by the LS: they inhabit the surface modifying the sparse field, and update the values at all zero-level points within a given range r of the agent's position x by a value a, $a = A(x)$, weighted by the distance from the agent, according to a normalized Gaussian distribution.

In [182], individual contours are represented by states and an ES is used to deform the actual contour according to the region-based constraints introduced. Figure 4.6 displays two paradigmatic examples of encoding.

Optimization procedure

In general, the operators used with LS methods are quite classic and not too customized. The model constraints are usually managed by all the underlying maths inherent to the LS formulation: the curves are split and merged naturally adjusting the object topology. Furthermore, in some cases, the existence of a training set already constraints the possibilities of generating new shapes.

In [136], as explained before, agents inhabit the zero-level surface, sensing and modifying it as necessary using straightforward interpolation routines, and are allowed to modify the surface at their location effectively, whilst maintaining the structure of the sparse field. Basically, every agent move independently using two rules/operators: a) navigation across the surface, and b) modification of the sparse field. For the former, we need to define a movement potential function P, while for the latter it is necessary a surface update function A.

In [181], authors have designed a method based on two operations (hopping and local optimization). In turn, the hopping phase consists of a stochastic split and a deterministic merge, in which a large area of the image can be split and merged in a single step, thus reducing the number of hops needed to search for a global minimum. To split the selected segment into two subsegments, the ISODATA thresholding method is used.

In [182], since the recombination of contours requires a much higher computational complexity, only mutation by addition of a Gaussian perturbation is used. As selection scheme, a modified (μ, λ) is applied. The notation (μ, λ) indicates that μ parents create $\lambda > \mu$ offspring by means of mutation, and the best μ offspring individuals are deterministically selected to replace the parents. However, the acceptance of temporary deterioration might also make (μ, λ) selection drift away from the contour energy minimum. To avoid this, they select the state with lowest energy from μ survivors in (μ, λ) selection and compare it to the lowest energy state selected from previous selections. The state with lowest energy found in all the selection processes when the algorithm is over represents the output contour and the solution to the problem.

Finally, in [6], a dynamic GA is used, that adaptively changes the probability of crossover, the probability of mutation, and the number of crossover points.

Target Function

In [5], the main aim is to select the seven ideal parameters, introduced in [6] to eliminate the need for a re-initialization, for different kinds of image modalities and organs (CT and MRI in particular). To do so, authors developed an evaluation function which includes four measures to calculate geometric differences between "the object boundaries as determined by the level set method" and "the desired object boundaries". Something similar is used in [2], where a GA is in charge of tuning the weights and the parameters of each term based on training data. In general, the quality of a solution is defined as the average quality of the segmentations obtained using the parameters values encoded by such a solution during the training phase.

In [182], the problem is to find a closed contour $C(s,t)$ enclosing a region Ω_C such that

$$E(C(s,t)) = \frac{1}{\oint_C ds}\{\oint_C \frac{1}{1+|\nabla G * I(x(s),y(s))|^p} ds\} \tag{4.1}$$

is minimized with the region characterizing constraint

$$D(x,y) = \frac{1}{1+|\nabla G * I|^2} e^{-\frac{|I(x,y)-I_0|}{\sigma}} \geq T_V \tag{4.2}$$

for all $(x,y) \in \Omega_C$, and where $p = 1$ or 2, T_V is the similarity threshold, and $|\nabla G * I|$ is the absolute value of the gradient of the image intensity ($I(x,y)$) smoothed by a Gaussian filter $N(0,\sigma_0^2)$. The solution to the constrained optimization problem is to use ES to deform $C(s,t)$ until an optimum $C(s)$ is reached.

In the several papers using eigen-shapes, despite their differences (kind of textural measures used and calculation of the affine transformation), the fitness function and the general pipeline is essentially similar: all employ texture and shape information to evolve the contour using a training and a test phase, and the fitness of an individual is

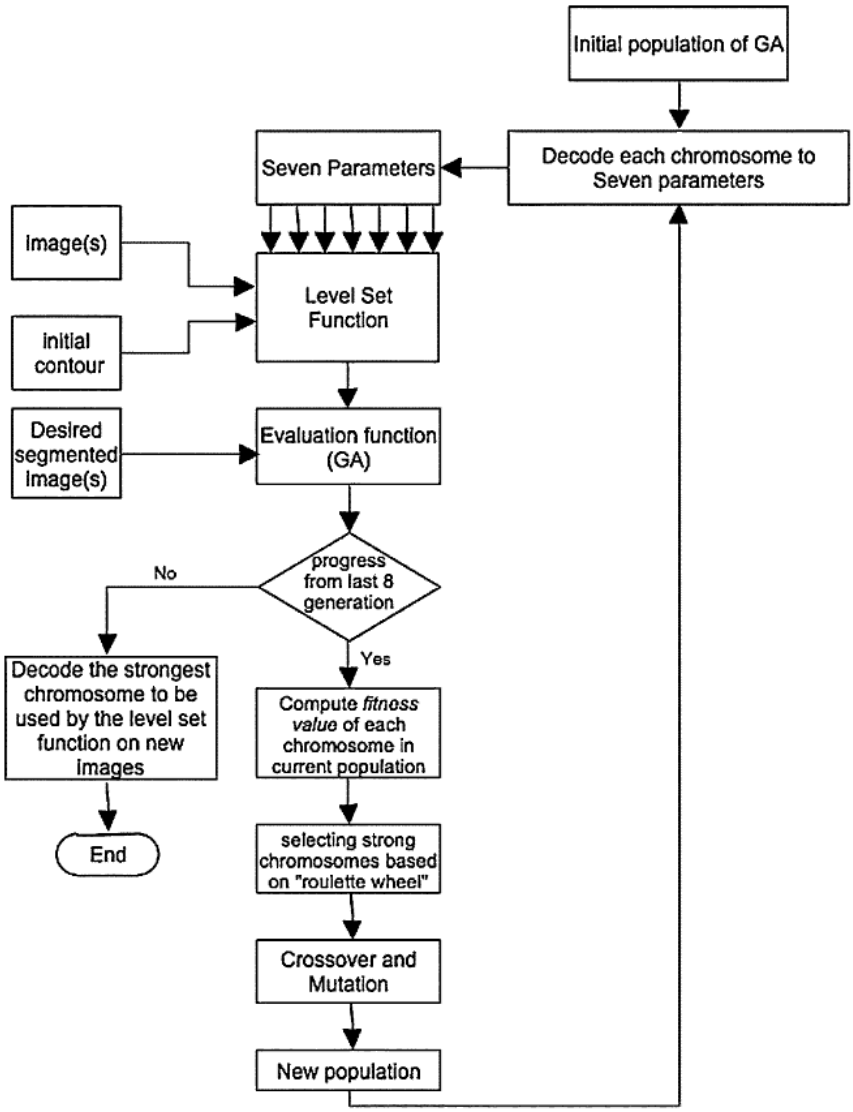


Figure 4.7: Main workflow diagram for [5].

measured by comparing the textural difference between the regions inside and outside the contour. In other words, the fitness of a given shape is determined by the match between the texture of its enclosed region and the mean texture of the target object calculated in the training phase.

In [181], each point is attracted to a local minimum through a gradient descent process to find local minima and, after that, basin hopping between local minima (basins) simplifies the energy landscape. If the new local minimum corresponds to a lower energy, then the new solution will be taken without reservation. Otherwise, a coin may be flipped to determine if the new solution is accepted. In such a work, the updates are global i.e., instead of each point on the LS function moving in its normal direction at a speed related with the gradient descent, the hopping step is region-based, allowing to escape from local minima effectively. In general terms, for a given image u_0, the piecewise constant Mumford-Shah model seeks for a set of curves C and a set of constants $c = (c_1, c_2, \ldots, c_n)$ which minimize an energy functional given by

$$\begin{aligned} F^{MS}(C, c) = \sum_{i=1}^{n} \int_{\Omega_i} |u_0(x, y) - c_i|^2 dxdy + \\ + \mu \times Length(C) \end{aligned} \tag{4.3}$$

In the previous equation, the curves in C partition the image into n mutually exclusive segments Ω_i for $i = 1, 2, \ldots, n$. The idea is to partition the image so that the u_0 intensity of each segment Ω_i is approximated well by a constant c_i. The goodness-of-fit is measured by the fitting term $\int_{\Omega_i} |u_0(x, y) - c_i|^2 dxdy$.

In [136], the movement potential function P is defined as $P(x) = ||\mathbf{v} + \gamma H(x) + \lambda||\mathbf{u} - x||||$, where x is an agent's $\mathbb{R}^3$ location, $\mathbf{v}$ is the previous velocity, $H(x)$ is a steering function (moving the agents automatically toward areas of interest, for example, using the gradient H can lead the agents away from areas of high voxel intensity gradient, i.e. toward more homogeneous areas), $\mathbf{u}$ is a user-specified location, and γ and λ are constants. In this work, the agent behaviour algorithm is entirely deterministic. When the agents are in an area that does not trigger any surface modifications (when the occupied area of the image is such that it does not stimulate agents to modify the level

set surface), all the agents simply move together, along a near-circular path orbiting the user-specified point whilst being pulled toward the path of lowest image gradient.

Critical Discussion

When a set of training images is available, one interesting option would be to use an approach similar to the one by Ghosh et al.[4, 135] and Mesejo et al.[173]. Nevertheless it is noteworthy that these approaches are quite slow due to the need to compute the average shape and the main forms of variation, as well as the evaluation of the texture that characterizes the evolving contour for every iteration and individual.

The hybridization of LS with MH can generate interesting findings in the future. First, because, not all the possibilities have been fully explored up to now. Second, because the combination of two techniques with so many positive aspects, it cannot be more than satisfactory. On the one hand, MHs provide learning and global search capabilites, thus avoiding local minima, making the initialization of the initial contour robust and the introduction of new evolution terms straightforward, as well as avoiding the need of computing derivatives and speeding-up evolution. On the other hand, the LS method provides easy management of topological changes and adaptation to solve any dimensional problem, as well as the areas inside and outside the evolving contour can be quickly determined[3]. In particular, interesting examples can be found when trying to optimize the parameters of a LS model [2, 5], to include different terms in the formulation [2], to use swarms of agents that refine the obtained results [136], to take advantage of a training set of shapes to segment a difficult structure [4, 135, 173], as well as to solve the Mumford-Shah functional [181] or to quickly initialize a LS [180].

[3]In fact, all papers under consideration but one are related to medical applications due to the significance and importance the LS method has in the segmentation of biomedical images.

Part II: Proposed Methods

The results presented in this PhD have as starting point a Multicentre Project funded by the Compagnia di San Paolo (Torino, Italy), under the Neuroscience Programme. In that project, we participate in an interdisciplinary consortium aimed to significantly extend the current understanding of the complex signaling events occurring in dendritic spines (DS)[4], by integrating 'state of the art' computational and experimental approaches. We proposed to dissect the function of DS by integrating high-throughput data generation and mining, experimental validation and theoretical modeling in a coherent, unified view.

Besides traditional statistical techniques which can be applied to real-world data whose properties are not known or are affected by significant noise, a number of SC techniques has also been deployed. Research results have shown that such techniques can be particularly effective in image analysis. On this basis, we aim at developing new recognition algorithms for brain anatomical structures, capable of automatically extract two types of information:

1. the relative abundance of the mRNA of every gene in the main brain areas; and

2. the identity of mRNAs that are particularly enriched in dendrites, which are very likely to be involved in DS biogenesis/functions.

[4]Spines are neuronal protrusions, each of which receives input typically from one excitatory synapse, that contain neurotransmitter receptors, organelles, and signaling systems essential for synaptic function and plasticity. Numerous brain disorders are associated with abnormal dendritic spines, and spine formation, plasticity, and maintenance depend on synaptic activity and can be modulated by sensory experience.

The motivation for developing 'intelligent' computer vision approaches can be found in the great interest that RNA molecules specifically enriched in the neuropil of neuronal cells (and, in particular, in DS) have in neurobiology, in virtue of their involvement in synaptic structure and plasticity. The systematic identification of such molecules is therefore a very important task. In that regard, high resolution images of RNA ISH experiments contained in the ABA represent a very rich resource to identify them and have been so far exploited for this task based on human-expert analysis. However, software tools that may automatically address the same objective are not very well developed.

With this aim in mind, three different segmentation methods (that allow for the accurate delineation of a particular anatomical region) have been designed, implemented and tested, using the ABA as reference dataset for the development of new brain image analysis algorithms.

The first method (section 5) uses a training set of ABA images to extract the limits for the deformation and main modes of variation of a parametric DM, in order to manage very simple templates that can be used as part of a fast approach to perform genome-wide experiments. Such a method is divided into four different stages (DM initialization, localization using DMs and metaheuristics, segmentation, and expansion of the segmentation using ensemble classifiers), and can be seen as an intelligent manner of localizing promising areas where a fast and well-established segmentation technique is going to be applied. The main advantages of this approach are its execution time and accuracy, while its main disadvantages are its ad-hoc nature (it needs a training set of shapes and textures, as well as suitable parametric models of the object to locate), the introduction of prior shape knowledge only in the localization step (and not in the segmentation stage, allowing the existence of problems with very ambiguous regions), the impossibility of managing topological changes, and the manual and tedious creation of the templates of the object to find.

The second method (section 6) is introduced to investigate other uses of a training set of shapes, and to try to overcome some of the disadvantages of the previous approach. The LS method is used to allow topological changes, and the templates are created

automatically using principal components analysis to derive texture, mean shape and shape variability from the training set. After that, a metaheuristic is used to evolve the segmentating contour optimizing the weights of a linear combination of the differente eigenshapes and the mean shape, by means of an intensity- and texture-based fitness function. This method obtained reasonably good results, but worse than the previous approach and with a larger execution time, so it can be seen as an intermediate stage that led us to the development of the third segmentation method.

The third method (section 7) tries to solve all the previous problems found, taking into account the difficulty of developing a general and, at the same time, accurate method. First of all, the LS method is used to manage possible topological changes. Secondly, three different sources of information are taken into account (intensity, boundaries, and shape). Thirdly, the automatic MH-based tuning procedure allows to use the same method over different image modalities and anatomical structures, therefore making this approach general. And finally, only a single previously segmented image is necessary to perform the segmentation instead of, as in previous methods, a complete training set. This is possible thanks to a deformable registration process in which the target image is registered with an Atlas. In terms of generalization [5] and accuracy, this method obtained the best results, but it also presents two drawbacks: it critically depends on the underlying registration step, and it is slower than the first method with ABA images.

The following three chapters describe these methods in detail.

[5] Here, with 'generalization', we refer to the possibility of easily applying this approach to different image modalities and/or anatomical structures of interest without a major and time-consuming modification in the original pipeline.

Chapter 5

Hippocampus Segmentation in using Active Shape Models and Random Forests

5.1 Histological Images and Hippocampus

Among the different anatomical structures which make up the mammalian brain, the hippocampal formation (HPF) (see Figure 5.1) is particularly interesting. From an anatomical viewpoint, the HPF, composed by the Hippocampus and the Subiculum (SUB), is located within the medial temporal lobe. In turn, the Hippocampus is composed by the Dentate Gyrus (DG) and Ammon's Horn (CA), which is further composed by three different regions (CA1, CA2, and CA3).

The hippocampus has long been known for its crucial role in learning and memory processes [189]. Moreover, it has recently been demonstrated that the volume of the hippocampus is an early biomarker for Alzheimer's disease. Therefore, there is a great interest in understanding the cellular and molecular events that take place in this structure, under both normal and abnormal conditions. From this point of view, a precise gene expression map at the cellular and subcellular level within this region

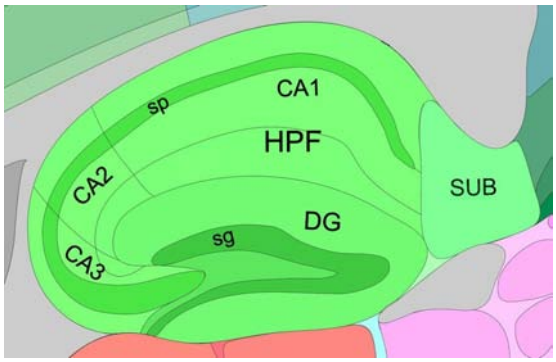


Figure 5.1: Regions in Hippocampal Formation (HPF)

can provide crucial information for understanding such biological mechanisms. A very promising data source to derive this map has recently been provided by the Allen Brain Atlas (ABA), a huge, publicly available database that contains high-resolution images mapping the expression patterns of most genes contained in the genomes of the analyzed organisms (see 3.2).

In this chapter, we describe a general model-based method for the automatic localization of anatomical districts in histological images, which could be also applied to other structures, as well as to different imaging modalities. In particular, we describe an application in which such a method is used to locate the hippocampus in ISH images from the ABA. This method allows us to automatically extract image parameters from corresponding regions of a huge number of images, to cluster genes with similar expression patterns and subcellular mRNA distribution. It can be hypothesized that genes whose expression is mapped in images with similar visual features are likely to be also functionally similar.

In our method, the hippocampus is located by detecting, as landmarks, two regions which are usually well distinguishable within the structure (see Figure 5.1): the pyramidal (*sp*) and granule (*sg*) cell layers, which belong to the CA and DG regions, respectively.

As in most medical imaging applications, the problems we have to solve are mainly related with the characteristics of the images involved in the process. The main problematic features of these images in general, and of the hippocampus region in particular,

related with biological features or with the image acquisition process, can be summarized as follows:

- natural variability of brain structure shapes in different subjects;
- fuzziness of the hippocampus boundary;
- limited relevance of color for detecting anatomical structures: regions with similar colors may represent different structures and vice versa, depending on the dye used as well as on local image acquisition settings;
- contrast variability between structures: different genes are not expressed equally in the same anatomical region, making it difficult to construct a consistent model for each landmark in all images. Moreover, grained patterns with many irregularities hamper the classification of individual pixels as belonging to the anatomical structures under consideration;
- orientation issues: the imaged structures may be rotated or displaced on the slice with respect to a "standard" alignment;
- lighting issues: within the same set of images, some are much brighter than others.
- variable resolution even within the same image: high-resolution regions coexist with low-resolution ones;
- presence of artifacts: tears, scraps, bubbles, streaks in tissues, partial cut-off of regions;
- large image size (the typical resolution of ABA images is about 15,000 $\times$ 7,000 pixels, and the ROIs which we work with about 2,500 $\times$ 2,000 pixels).

These problems significantly hamper tasks like localization and segmentation of structures in such images. More specifically, and from a low level point of view, these ROIs have the following average histogram (Figure 5.2), where the darker parts represent, among other things, the hippocampus.

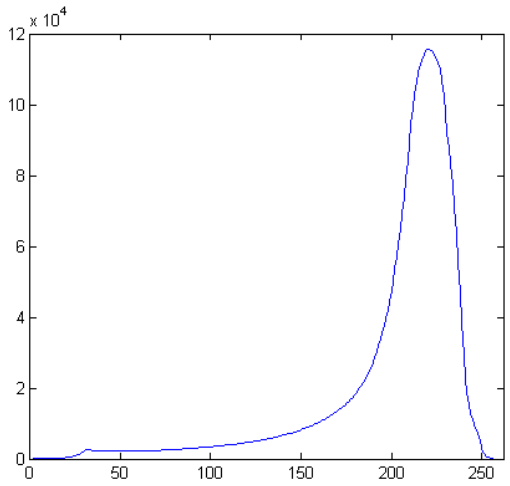


Figure 5.2: Average histogram of the images under study.

The fully-automatic 2D localization method we propose is based on atlas-based registration and on the optimization of the parameters of a parametric deformable model. Our method can be divided in two stages: (i) selection of the corresponding slice in the reference atlas based on a two-step affine registration, and (ii) proper localization of the hippocampus. We compare six real-parameter optimization techniques on the task under consideration; in particular, we show that DE significantly outperforms the other methods taken into consideration: Levenberg-Marquardt (LM) [190], SA, SS, GAs and PSO. In our tests, we evaluate the localization of the hippocampus in real and synthetic sagittal images, but this method could as well be applied to other subcortical structures, image modalities or anatomical planes.

Despite the importance of histology, a branch of biology focused on the study of the microscopic anatomy of cells and tissues, it is surprising to check that most literature devoted to histological image processing and analysis is focused on registration and 3D reconstruction of the whole brain: 3D reconstruction from a sequence of histological coronal 2D slices using a model built by non-linear transformations between the neighbouring slices [191]; image registration combining the high-frequency components of slice-to-slice histology registration with the low-frequency components of the histology-to-MRI registration [192]; a 2D to 3D nonlinear registration

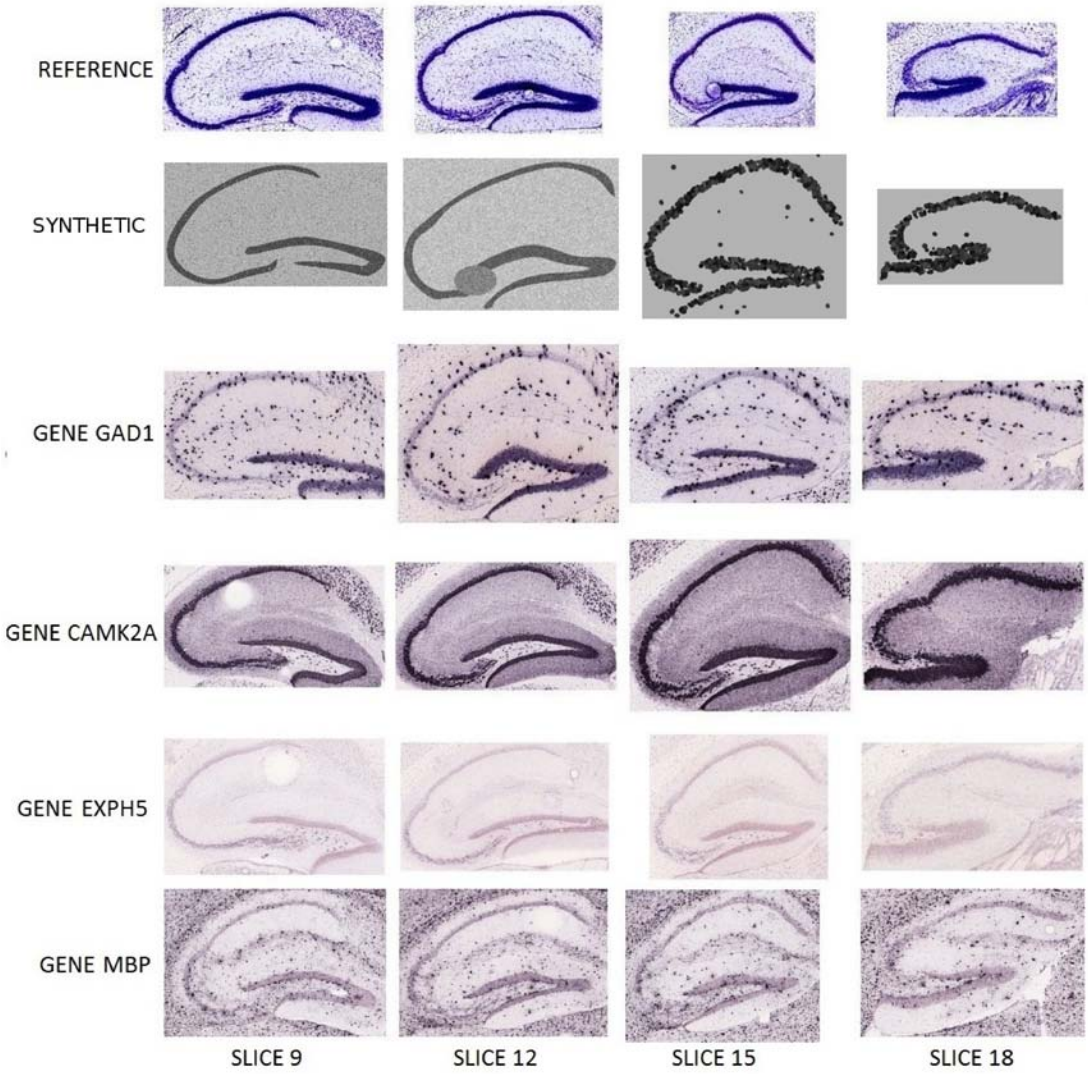


Figure 5.3: Hippocampus variability. Horizontal-wise: hippocampus sections taken at different levels look different, and different genes can as well produce very different visual features. Vertical-wise: corresponding sections from different brains maintain some shape similarity.

using a registration technique, based on partial differential equations, driven by a local normalized-mutual-information similarity measure [193]; slice-by-slice segmentation of anatomical structures where the successful segmentation of one section provides a prior for the subsequent one [194] (they assume that the segmentation of few sparsely sampled slices is done manually, so it is not a completely automatic method). In [195], the segmentation is treated as a classification problem using RF and Markov Random Fields, which refine the results at the pixel level, but the method requires previous knowledge about the reference slice associated to that image. Therefore, one could state that there is a lack of research about automatic segmentation of anatomical brain structures in histological images.

5.2 DE-based hippocampus localization

Our automatic structure localization method consists of two phases:

1. *Best Reference Slice Selection*, implemented as a two-step affine registration method which: (i) determines the position of the section displayed in the target image according to a reference atlas, and (ii) extracts the region of interest (ROI) where the hippocampus is more likely to be located. Hence, it represents the initialization of the deformable model: it determines which template should be used and where it should be applied.

2. *Structure Localization*, in which the point distribution model selected in the previous step is adapted by a stochastic optimization procedure based on DE, to fit the image region where the structure of interest (if any) is located.

Best Reference Slice Selection

The initialization of the Deformable Model, in terms of choosing a model and its starting position, is obtained using an atlas-based affine registration with the reference images of the ABA. The main idea is to find the sagittal reference slice of the atlas

which best matches the target image. This phase produces two results: firstly, based on the information contained in the corresponding reference atlas image, it allows one to extract the ROI where the hippocampus is expected to be located. Secondly, it makes it possible to determine the position, within the brain, of the section represented in the target image. Consequently, one can select the corresponding point distribution model of the hippocampus (derived empirically as described for the ASMs), which is to be applied in the following step within the selected ROI.

To find the best reference slice, we use a two-step method (summarized in Figure 5.4) that considers the similarity between the global shape of the brains and of the two hippocampi. The only piece of a-priori knowledge we use is that the position of the hippocampus is broadly the same in every brain and that all the hippocampi have roughly the same shape.

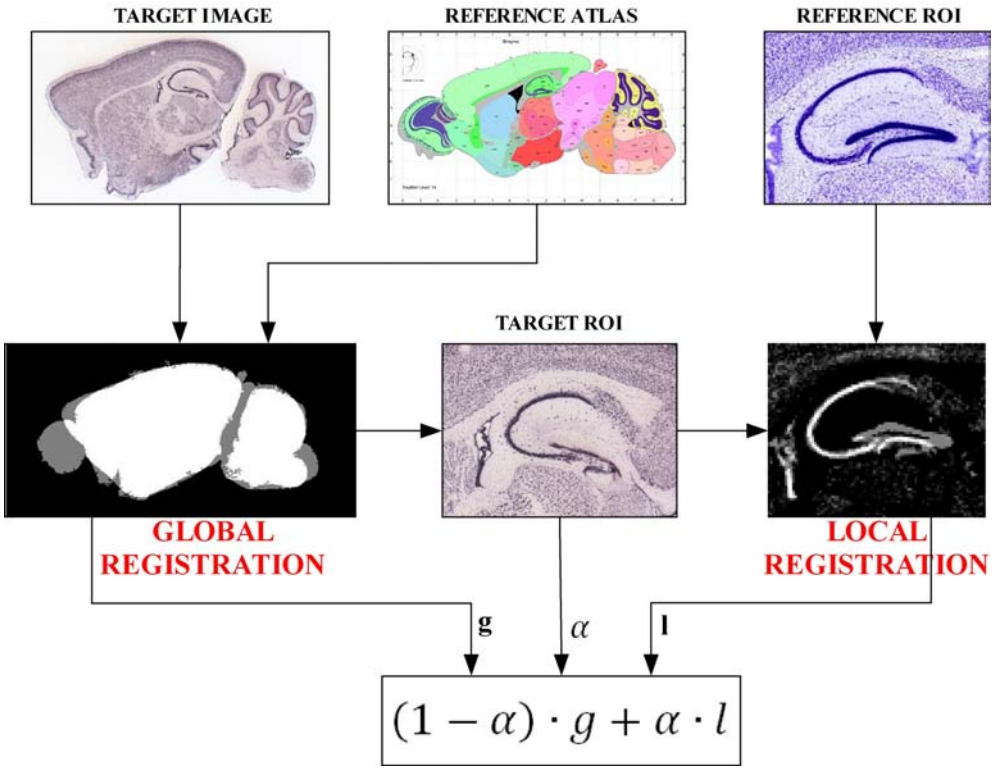


Figure 5.4: Best Reference Slice Selection: Data Flow

The first step performs an affine registration between the target image and each of

the Atlas references. Affine registration is a widely used technique for reconstructing and matching medical images since it can usually reach a good trade-off between quality and computational effort.

As explained before, in this stage, the intensity levels of the images are not very useful; on the contrary, they can sometimes spoil the results of the registration because the algorithm could try to match different anatomical parts of the brains just because they accidentally have the same intensity level. For this reason, we ignore these differences and do the affine registration based only on the shape of the two brains. This means that the images involved in this registration process are not the ones downloaded from the ABA, but the same images after a segmentation and binarization phase aimed at obtaining an image that is white where the brain is visible (foreground) and black elsewhere (background).

We base the affine registration only on the shape of the two brains. This means that the images downloaded from the ABA are first segmented and binarized before registration to obtain a binary image that is white where the brain is visible (foreground) and black elsewhere (background).

After the global registration (in which Levenberg-Marquardt is used as optimizer), we extract the region of the target image that falls within the coordinates of the ROI which contains the reference hippocampus. We assume that, at this point, the hippocampus of the target image is entirely or partially included in the extracted ROI, and that SP and SG are among the brightest regions of the ROI. Then, an equalization process is performed to reduce the differences between the images, and we start a local affine registration process (whose optimization procedure is based on PSO), in which we expect to match the two hippocampi.

After these two registration steps, we have two measures that represent the similarity between the target image and each reference image:

- g_k, is the squared difference[1] between the binarized target and the binarization of

[1]The squared difference is a similarity metric calculated as follows: $\frac{1}{N}\sum_{i=1}^{N}[S(i) - M(i))]^2$, where N is the number of pixels in the scene (S) and model (M) images.

the k^{th} reference brain slice;

- l_k, is the squared difference between the hippocampus regions in the two images[2].

The reference image that minimizes the following fitness function (see Figure 5.4) is selected as best slice:

$$f_k = (1 - \alpha) \cdot g_k + \alpha \cdot l_k$$

where $\alpha \in (0, 1)$ is a parameter that depends on the "quality" of the extracted ROI. A good quality ROI is an image where the anatomical parts of the hippocampus are clearly visible. We observed that "good" images can be distinguished from "bad" ones by simply looking at the standard deviation of the intensity levels of the image: bad images usually have small standard deviations and vice versa. Accordingly, we set α to be proportional to the standard deviation of the ROI. In this way images in which the hippocampus is not clearly visible (and consequently may cause a bad local registration) do not influence the results of slice selection.

The output of this phase consists of the region of the target image where the hippocampus is expected to be found and the estimated position of the image within the brain. This also indicates which are the best models to use for the localization, ranked by similarity.

In order to evaluate the correctness of this two-step approach, we computed the distance (in slices) between the slice selected by our system and the one suggested by the ABA on a test set of 320 images. In 45% of cases the slice selected was exactly the same, while in 43%, 10% and 2% of cases there was a difference of one, two, or three or more slices, respectively. In those cases in which our system selected a reference slice different from the one proposed by the ABA, we also compared the fitness values obtained by the hippocampus localization algorithm. The results of these experiments showed that there are no statistically significant differences between the two choices. Therefore, these results demonstrate that our initialization method is effective and can be used also in problems where no ground truth is available.

[2]It is fundamental to understand that, while g_k is calculated using the complete brain image, l_k is calculated only taking into account the extracted ROIs.

Hippocampus Localization

While in the standard ASMs external forces are driven by the contours in the image, in our approach we parametrically deform the model shape to match as closely as possible the shape of the hippocampus in the region we want to locate. The model is moved and deformed, by altering its parametric representation using an optimization heuristic which maximizes a function which measures the similarity between the model and the object itself. Since, in our problem, color and shapes are greatly variable, the only common information on which we can rely is that all hippocampi have two substructures (*sg* and *sp*) whose shapes have more consistent features, besides being usually characterized by lower intensity values and higher color saturation with respect to the surrounding structures.

To enhance the contrast between *sg*, *sp* and the surrounding parts, we pre-process the ROI containing the hippocampus by stretching its histogram. We invert pixel intensities, to consistently use the convention by which, as for binarized images, the most relevant regions in the image have higher intensities. These operations are not applied to the original images of the ABA we show, to avoid altering their actual appearance.

Templates

Obviously, our main concern is to adapt a model that permits an efficient localization, but we want, as well, to obtain this goal with the simplest possible model, i.e. having the smallest possible number of points in each template. In this work, for every slice of the reference atlas, two templates (one composed of 7 points for *sg*, one with 8 points for *sp*) have been created by manually selecting significant corresponding landmarks in all images in a training set.

A template aims at taking into account all possible positions and deformations with respect to the prototypical shape of the structure it represents. The template we use does not refer to the absolute positions of the points, but describes the relative positions (or shifts) between consecutive points in polar coordinates. This is uncommon for a DM, since the majority of the implementations use cartesian coordinates. Every template is

composed of two parts:

- an "inner set" of points that lies on the anatomical part we want to locate;
- an "outer set" of points that lies just outside it, obtained by rigidly shifting the previous set.

For a proper localization, the "outer set" is fundamental; otherwise, using only the "inner set", a completely dark and large area would always achieve the highest values of the target function (see Figure 5.5). These sets of points are based on the following observation: the hippocampus is always darker, if only slightly, to immediately adjacent regions. Thus, a parametric model that seeks to maximize the difference, in terms of intensity, between an internal/inner model (which supposedly overlaps the hippocampus) and an external/outer model (which should be located in a more clear region surrounding the hippocampus), can help decisively in finding the location of this anatomical structure.

A template is fully described by four $2 \times n$ matrices, where n is the number of points in the template:

$$\mathbf{S} = \begin{bmatrix} \rho_{m1} & \vartheta_{m1} \\ \rho_{m2} & \vartheta_{m2} \\ \vdots & \vdots \\ \rho_{mn} & \vartheta_{mn} \end{bmatrix} \quad \Delta = \begin{bmatrix} \Delta_{\rho 1} & \Delta_{\vartheta 1} \\ \Delta_{\rho 2} & \Delta_{\vartheta 2} \\ \vdots & \vdots \\ \Delta_{\rho n} & \Delta_{\vartheta n} \end{bmatrix}$$

$$\mathbf{L} = \begin{bmatrix} \rho_{l1} & \vartheta_{l1} \\ \rho_{l2} & \vartheta_{l2} \\ \vdots & \vdots \\ \rho_{ln} & \vartheta_{ln} \end{bmatrix} \quad \mathbf{U} = \begin{bmatrix} \rho_{u1} & \vartheta_{u1} \\ \rho_{u2} & \vartheta_{u2} \\ \vdots & \vdots \\ \rho_{un} & \vartheta_{un} \end{bmatrix}$$

$\mathbf{S}$ is the "standard template" and represents the standard coordinates of the inner set, Δ is the displacement of the outer set with respect to the inner set, $\mathbf{L}$ and $\mathbf{U}$ are the minimum and maximum values allowed for every parameter that describes the inner

set. It should be noticed that ρ_1 and ϑ_1 represent the positions of the first point with respect to the upper left corner of the image. After that point, proceeding row-wise, every (ρ, ϑ) pair represents the shifts of the subsequent point in terms of distance and angle with respect to the previous point, respectively. As a consequence, the number of parameters to optimize for each model is twice the number of the points in the template (14 parameters for *sg* and 16 for *sp*).

The matrices $\mathbf{S}$, $\mathbf{L}$ and $\mathbf{U}$ have been computed by manually selecting the reference points in a training set. The first one is the median of the selected shifts and the other two are the minimum and the maximum values observed in the training set, respectively. To improve the templates, a manual refinement of the parameters has been performed. The matrix Δ has been manually built based on the observation of several hippocampi.

From a computational point of view, a model is a vector of $2 \times n$ elements that ranges within the "boundary templates" $\mathbf{L}$ and $\mathbf{U}$; its elements (coordinates) are evolved by a metaheuristic which optimizes the target function described in section 5.2. The aims with which the target function was defined are to match the model with the target while keeping it as close as possible to the shape of the standard template $\mathbf{S}$.

Target Function

As in classic deformable models, we consider the model to be subjected to external forces (driven by the image features) and internal forces (driven by the model itself).

The target function F, which is to be maximized, has three components: external energy E, internal energy I, and contraction factor C:

$$F = E - (I + C)$$

The external forces move (and deform) the model to maximize the intensity of pixels in the inner set, while minimizing the intensity of pixels in the outer set. For both sets, we evaluate the intensity of the image within a 3×3 neighborhood N_3 of all points in the model (Punctual Energy, PE) and in p intermediate points along the segment

between two consecutive points (Continuous Energy, CE), i.e.

$$PE = \sum_{i=1}^{n} [T(N_3(I_i)) - T(N_3(O_i))]$$

where n is the number of points in the model, $I_i = \{x_i, y_i\}$ is the i^{th} point of the inner set (in cartesian coordinates), $O_i = \{x_i + \Delta x_i, y_i + \Delta y_i\}$ is the i^{th} point of the outer set, $T(P)$ is the intensity of the image in P, if P is a point, or the average intensity, if P is a neighborhood; $\Delta x_i, \Delta y_i$ are the elements of Δ in cartesian coordinates. As well,

$$\begin{aligned} CE &= \sum_{i=2}^{n} \sum_{j=1}^{p} T(I_{i-1} + \frac{j}{p+1}(I_i - I_{i-1})) \\ &- \sum_{i=2}^{n} \sum_{j=1}^{p} T(O_{i-1} + \frac{j}{p+1}(O_i - O_{i-1})) \end{aligned}$$

where p is the number of points to evaluate in every segment. In our case we set $p = 20$. The final external energy is computed as

$$E = \gamma_P \cdot PE + \gamma_C \cdot CE$$

where the weights γ_P and γ_C, that can be used to give more importance to the punctual energy or to the continuous energy, have been empirically set to 5 and 1 for all tests, respectively (see Figure 5.5).

The internal energy I, related with the forces that oppose the deformation of the model, is computed as:

$$I = \xi_\rho \cdot \sqrt{\sum_{i=2}^{n} (\rho_i - \rho_{mi})^2} + \xi_\vartheta \cdot \sqrt{\sum_{i=2}^{n} (\vartheta_i - \vartheta_{mi})^2}$$

where ξ_ρ and ξ_ϑ are two positive weights that regulate the deformability of the

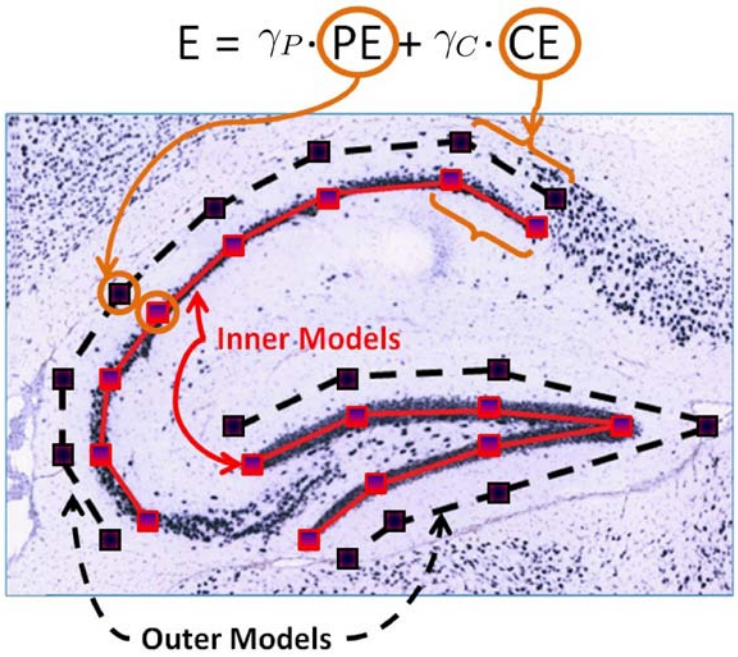


Figure 5.5: Inner and outer models superimposed to a hippocampus image, showing the components of the external energy.

model. The higher their values, the less the model deforms and the more it keeps the shape of the standard template $\mathbf{S}$. Please notice that the index i starts from 2. This way, the first shift (which actually represents the starting position of the model) is not taken into account and the model is independent of its position in the image.

Finally, the contraction factor C (Figure 5.6) also regulates the model's deformability to avoid unfeasible situations (e.g. having the extremes of *sg* too close to each other), and is defined as follows:

$$C = \xi_c \cdot \|I_n - I_1\|$$

If $\xi_c < 0$ the two extremes of the model repel each other; if $\xi_c > 0$ they attract each other. In our case we set $\xi_c > 0$ for the *sp* models and $\xi_c < 0$ for the *sg* models.

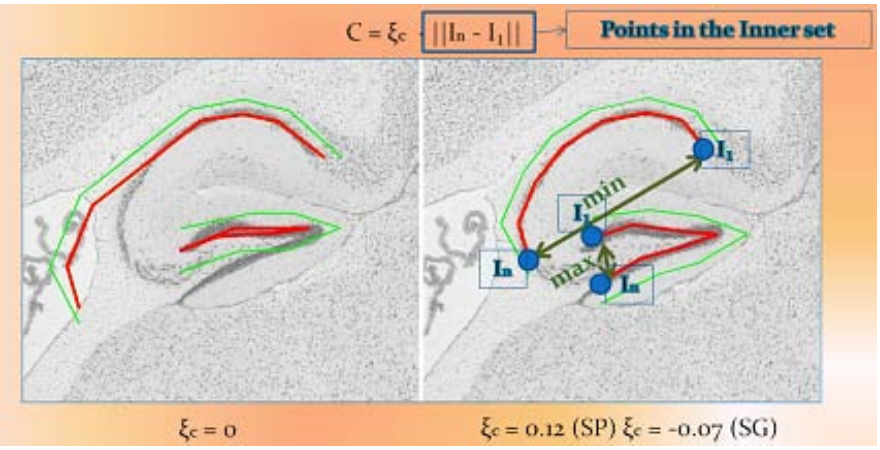


Figure 5.6: Contraction factor tries to avoid unfeasible situations.

Experimental Results

To test the system we used both real and synthetic images. As for the former, we randomly selected 320 images (corresponding to 320 different genes) from the ABA within subsets of representative samples of all possible hippocampi, featuring high and low levels of gene expression, good-quality and low-quality images, and so on. We also created 20 synthetic images that represent simplified versions of the real ones. In these images, representing all reference slices, the hippocampus is made up of small circles having random radius and color; small and large ellipses were also added to simulate cells and, finally, gaussian ($\mu \in [0.0, 0.2]$ and $\sigma \in [0.01, 0.15]$) and salt and pepper noise (density $\in [0.05, 0.25]$) were introduced to add fuzziness to the images (see Figure 5.7).

Tests were run on a computer equipped with a 64-bit Intel® Core™i7 CPU running at 2.67 GHz with 4 Gb of RAM.

We compared the results obtained by DE with those obtained by LM, a classic combination of the gradient descent and the Gauss-Newton methods, and the following stochastic optimization techniques: the previously presented variation of PSO (section 2.5); SS with local search based on the line search algorithm and the BLX-α crossover, as described in [196]; GA and SA. In LM and SA we used, as initial value, the best

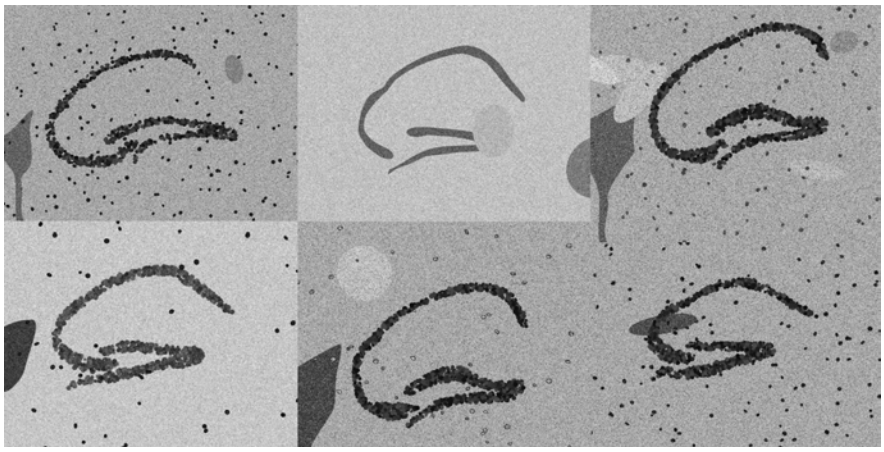


Figure 5.7: Examples of synthetic images used in the experiments.

solution in a randomly generated set of the same size as the other methods' populations.

The parameters (as well as the population topologies and the mutation and crossover operators) set in the tests have been initially chosen based on the most commonly used, and then refined during the development of the system. The values chosen for the most relevant parameters are shown in Table 5.1. For all techniques, a limit of 20000 function evaluations has been set as the only termination criterion, and the limits for the deformation were used as constraints in creating new solutions by crossover and mutation.

Table 5.1: Parameters used in testing different optimization techniques.

DE	PSO	SS	GA	SA	LM
$Cr = 0.9$ $F = 0.7$ Uniform Crossover Mutation: DE/target-to-best/1 Population Size = 80 Iterations = 250	$c_1 = 2.05$ $c_2 = 1.75$ $w_{max} = 1.0$ $w_{min} = 0.2$ Population Size = 80 Iterations = 250	$b_1 = 7$ $b_2 = 8$ $\alpha = 0.5$	$P_{crossover} = 0.6$ $P_{mutation} = 0.09$ Population Size = 80 Iterations = 250 Tournament of size 4 Uniform Crossover	$T_{start} = 1.5$ $T_{end} = 1E-9$ exponential cooling schedule: $T_{k+1} = 0.8 \cdot T_k$	$\lambda_0 = 0.01$

For every image, we first performed the extraction of the ROI and the selection of the best model, as described in Section 5.2. After this, for each optimization technique, we ran 25 tests on every image, for a total of 8000 experiments on real images and 500 on synthetic images.

Table 5.2: Comparative results of *sp* and *sg* localization for synthetic and real images (20000 function evaluations per experiment; the higher the values, the better the localization).

Synthetic Images						
sp localization						
Method	Average	StdDev	Avg. Worst	Avg. Best	Avg. Median	Wilcoxon test
DE	120.41	5.49	106.30	126.37	121.68	-
PSO	105.86	8.67	84.86	118.61	107.71	<1.00E-16
SS	105.70	6.16	91.66	115.69	106.37	<1.00E-16
GA	105.93	15.64	54.93	120.40	110.11	<1.00E-16
SA	101.80	8.19	80.87	110.64	103.76	<1.00E-16
LM	35.43	30.86	19.96	94.24	34.11	<1.00E-16
sg localization						
Method	Average	StdDev	Avg. Worst	Avg. Best	Avg. Median	Wilcoxon test
DE	128.91	2.92	121.18	132.99	129.49	-
PSO	118.29	5.85	105.55	127.63	119.05	<1.00E-16
SS	115.56	6.00	101.37	124.81	116.08	<1.00E-16
GA	115.03	9.67	87.86	127.20	116.49	<1.00E-16
SA	111.23	3.87	102.49	117.91	111.40	<1.00E-16
LM	64.76	19.58	33.79	103.34	61.42	<1.00E-16
Real Images						
sp localization						
Method	Average	StdDev	Avg. Worst	Avg. Best	Avg. Median	Wilcoxon test
DE	141.47	5.59	124.36	146.06	145.21	-
PSO	133.56	8.24	110.94	143.89	138.64	<1.00E-16
SS	131.06	6.04	118.54	141.66	135.81	<1.00E-16
GA	132.00	14.49	86.07	143.70	138.58	<1.00E-16
SA	131.00	3.05	123.83	137.71	134.91	<1.00E-16
LM	79.32	28.26	9.60	126.11	85.02	<1.00E-16
sg localization						
Method	Average	StdDev	Avg. Worst	Avg. Best	Avg. Median	Wilcoxon test
DE	148.24	1.54	143.58	149.73	149.30	-
PSO	144.65	3.69	135.37	148.70	145.52	<1.00E-16
SS	140.07	4.27	129.10	146.23	140.69	<1.00E-16
GA	140.89	6.83	122.56	148.50	143.06	<1.00E-16
SA	138.31	1.70	134.60	141.51	139.28	<1.00E-16
LM	105.58	20.01	65.54	141.61	110.52	<1.00E-16

Algorithm Comparison

Table 5.2 summarizes the results of our tests. In the upper part of the table we report the results for the 20 synthetic images and, in the lower part, the ones related to the 320 real images. The second column reports the average fitness for all tests, while the third reports the average standard deviation of fitness in the 25 experiments performed on each image. The fourth and fifth columns report the average of the worst and best results obtained for each image. Similarly, the sixth column reports the average of the medians.

We focused our analysis on the comparison between the performance of DE and of the other methods. DE achieved the best average fitness as well as the lowest standard deviation, which indicates a more robust behavior, as it is able to produce more consistent results over different trials; the performance of the other algorithms is more dependent on their random initialization. To assess the significance of this result, we performed a statistical test with confidence level of 0.01. We used non-parametric tests because the assumption of normality was not met. The Kruskal-Wallis test [197] was statistically significant in all cases (p-value $<$ 1.00E-16) and proved the existence of differences between the sets of results, where at least one sample median was significantly different from the others. After that, the paired Wilcoxon signed-rank test was performed [198], assuming as null hypothesis that the median of DE results is less than or equal to the median of the other methods. This is a one-tailed test in which the alternate hypothesis is that the median of any other method is less than the median of DE. The corresponding p-values (shown in the last column of Table 5.2) reject, in all cases, the null hypothesis, showing that significant differences exist between the performance of DE and of the other methods. Moreover, the average worst values show that DE avoids local minima and is able to get good results more consistently than the other methods.

Figure 5.9 describes the behavior of the five metaheuristics and a gradient-based local search method by plotting the average fitness value versus the number of function evaluations. It can be seen how DE and GA start better than the other techniques but, while the former continues to improve its performance, the latter is not able to refine

the solution found. At the same time, the evolution of SS is more discontinuous due to premature convergence of the reference set, followed, after each reconstruction, by an abrupt improvement of the fitness values. The worst results were obtained by the gradient-based local search method (LM), most probably because of the multimodality and non-differentiability of the function to optimize, which justifies the choice of using metaheuristics to solve this kind of problems In Figure 5.8, a visual justification of the use of metaheuristics for segmenting the hippocampus in histological images can be seen. Such a figure shows the highly-multimodal nature of the fitness function for hippocampus detection [199]: even varying only two of the parameters that control the model localization in the image, while fixing all the others, one can see that the fitness landscape has many local optima in which a local search method could fall. Also, in that figure one can see how higher values of the fitness function (in red) are associated with better positions of the model in the image.

Overall System Performance

In a second test we evaluated the results of the entire localization method by dividing the outcomes into three quality classes (see Figure 5.10):

1. Perfect Match: all points of the two models overlap the corresponding parts and cover them almost entirely;
2. Good Localization: (i) all points of the two models belong to the regions which must be detected, but they do not cover them entirely or (ii) at most two points are slightly outside of them;
3. Error: all other possibilities, from three or more misplaced points to models which are located in a completely different position of the brain.

If the results obtained by the model chosen in the Best Reference Slice Selection phase were not good enough (using a threshold for the target function values), we repeated the procedure using the model which ranked second in the previous phase, and took the best result.

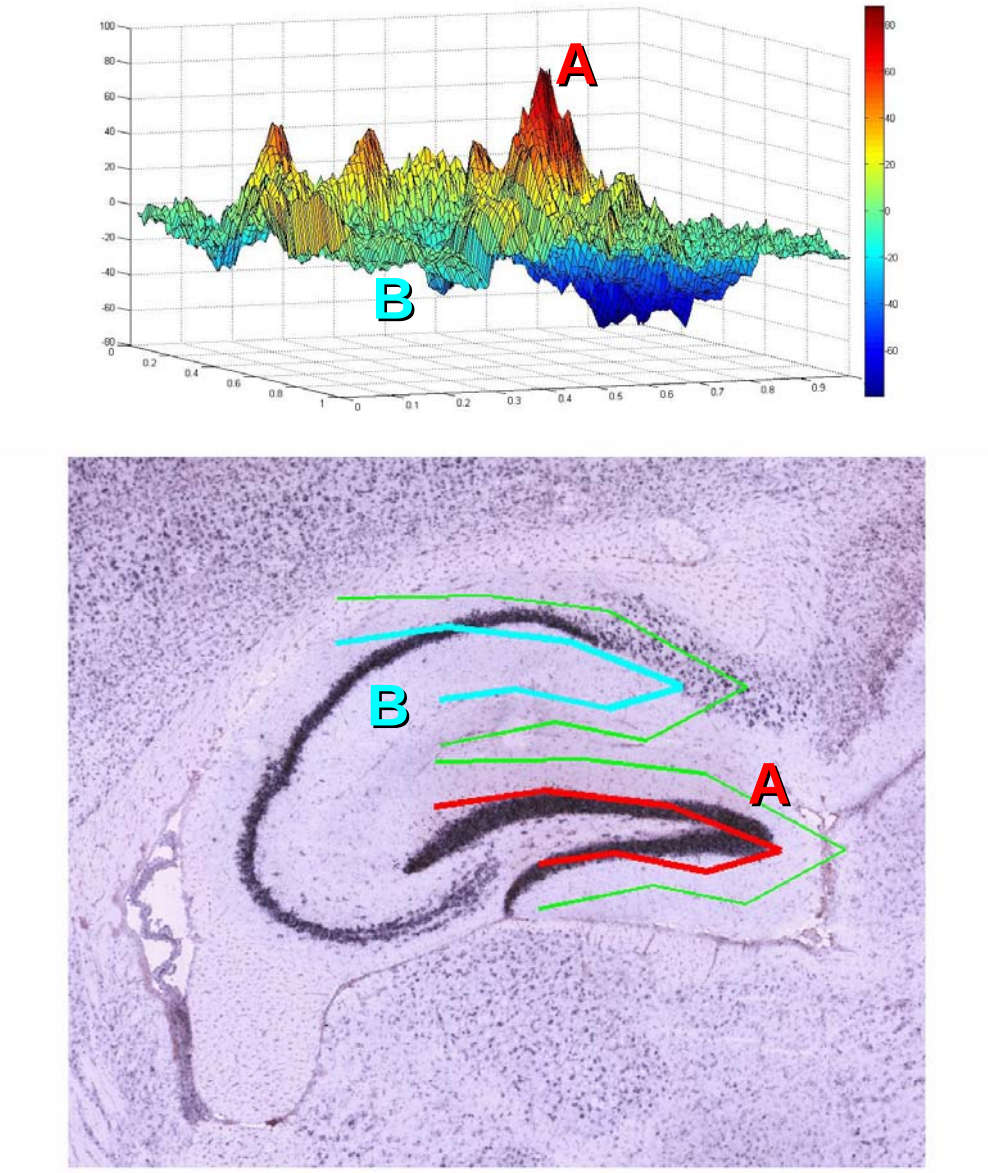
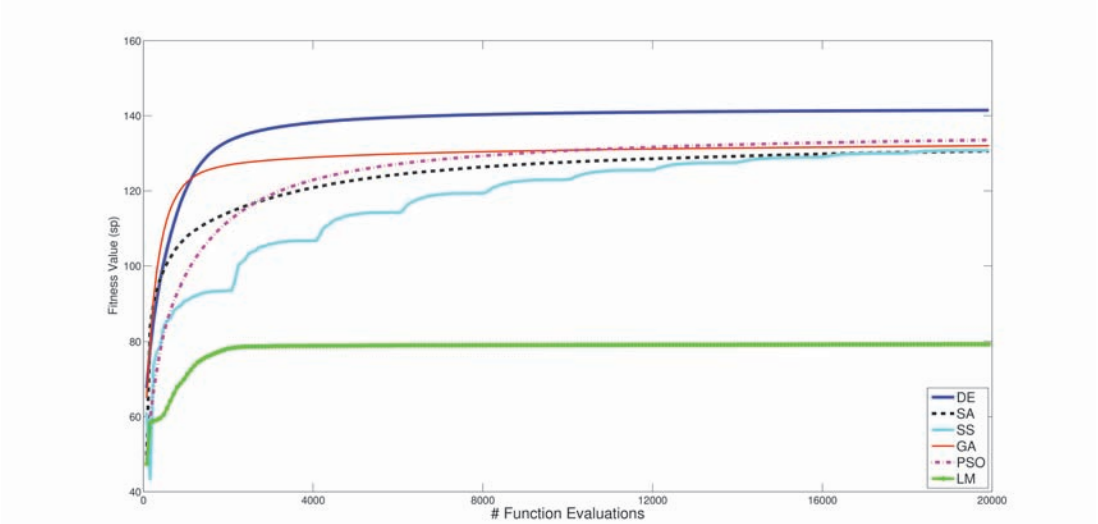
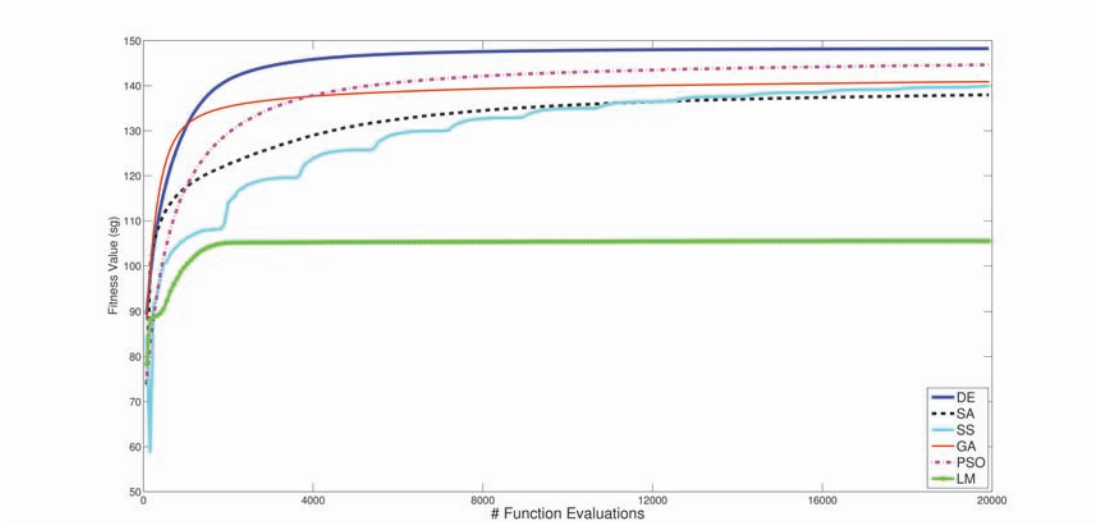


Figure 5.8: Fitness landscape multimodality in the hippocampus localization problem, and visual representation of how better values of the target function are associated with higher fitness values and better hippocampus localizations. The outer models are depicted in green.



(a) *sp*



(b) *sg*

Figure 5.9: Evolution of the fitness values in localizing *sp* (above) and *sg* (below).

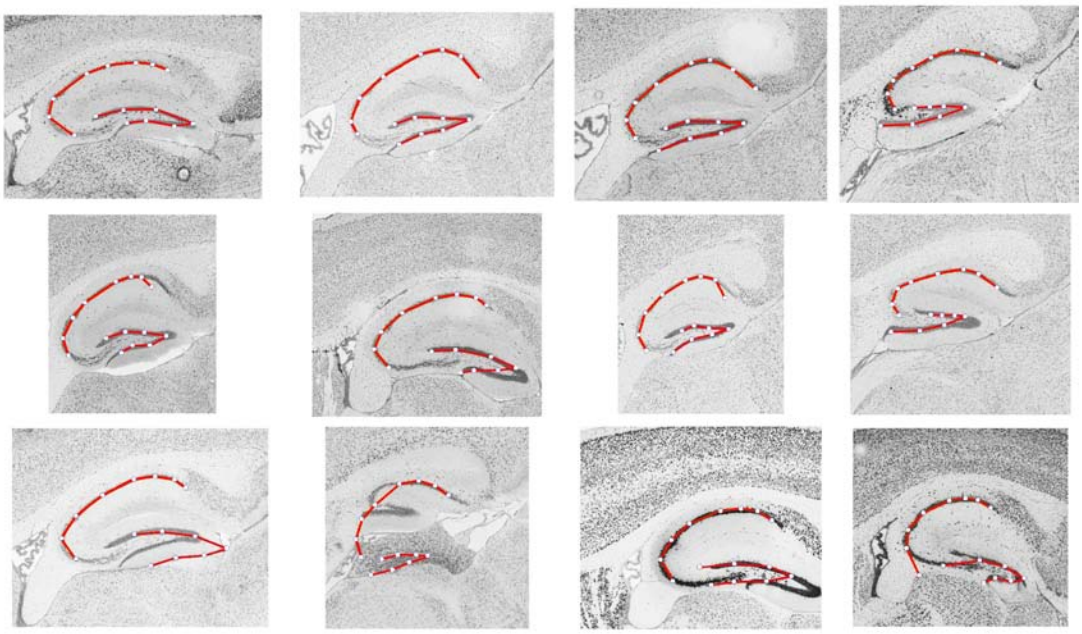


Figure 5.10: Upper row: perfect matches; middle row: good results; lower row: erroneous localizations.

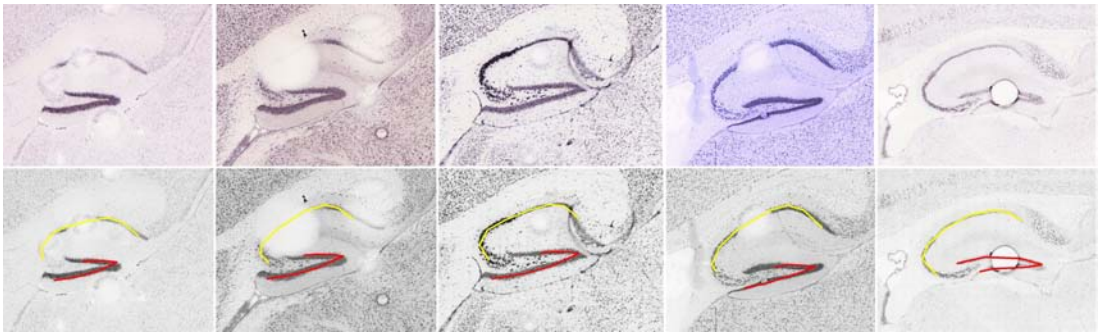


Figure 5.11: Some localization results over spoiled images of hippocampi.

Our method was able to perfectly localize the hippocampus in 58.0% of synthetic images and in 47.8% of real images, and reached a good localization in 35% and 43.1% of cases, respectively. This means that our method was able to localize the hippocampus satisfactorily in 93.0% of the synthetic images (20 images and 25 runs per image) and in 90.9% of real images (320 images and 1 run per image).

The most problematic situations that appear to drive the system to bad localizations are substantially three:

- low quality of images (for instance, images that are damaged or have very low contrast): this can affect the results of ROI extraction or of the Best Reference Slice Selection phase;
- images where the hippocampus has low levels of gene expression and is surrounded by other anatomical parts with higher expression levels;
- hippocampi having shapes which differ substantially from the typical cases included in the training set.

Most of these errors could most probably be avoided by increasing the size of the training set for the point distribution models, or improving the preprocessing phase of the images.

On the other hand, the method we adopted shows very good performances when dealing with corrupted images of hippocampi (see Figure 5.11).

With regard to time complexity, the localization task, implemented in C, employs on average 2.88 seconds per image, applying the Best Reference Slice Selection process on 10 reference slices.

5.3 Segmentation using Iterative Otsu's Thresholding Method and Random Forests

Segmentation

It is well-known that one of the main problems of DM is their initialization. In this work, the initialization, in terms of the choice of a model and its initial position, is solved by an atlas-based affine registration using the ABA reference atlas. The main idea, explained in the previous section, is to find the sagittal reference slice of the atlas which best matches the target image. This phase produces two results: firstly, it makes it possible to determine the position within the brain of the section represented in the target image and, consequently, to choose the right model. Secondly, based on the information contained in the corresponding reference atlas image, it allows us to extract the Region of Interest (ROI) where the hippocampus is expected to be located. As explained before, the parametric representation of the model is moved and deformed, guided by a MH, according to an intensity-based similarity function between the model and the object itself. The aim is to maximize the intensity difference between the hippocampus and the surrounding structures while keeping feasible the shape of the model.

The results of the localization, which represent the input to our segmentation method, are the ROI (whose size is usually around 2.500×2.000 pixels) and the points of the models which overlap the hippocampus. The segmentation, whose pipeline is shown in Figure 5.12, is based on a combination of classic computer vision techniques.

1. Preprocessing: a Median Filter (with a 25×25 pixels mask) is applied to the ROI in order to remove the noise that can affect the segmentation. The median filter in constant time[3] [200] was chosen since it offers a good trade-off between results and complexity.

[3]The use of a median filter has long been hampered by its algorithmic complexity of $O(r)$ (being r the kernel radius). With the trend toward larger images and proportionally larger filter kernels, the need for a more efficient median filtering algorithm becomes pressing. In [200], a new, simple yet much faster algorithm exhibiting $O(1)$ runtime complexity is described and analyzed.

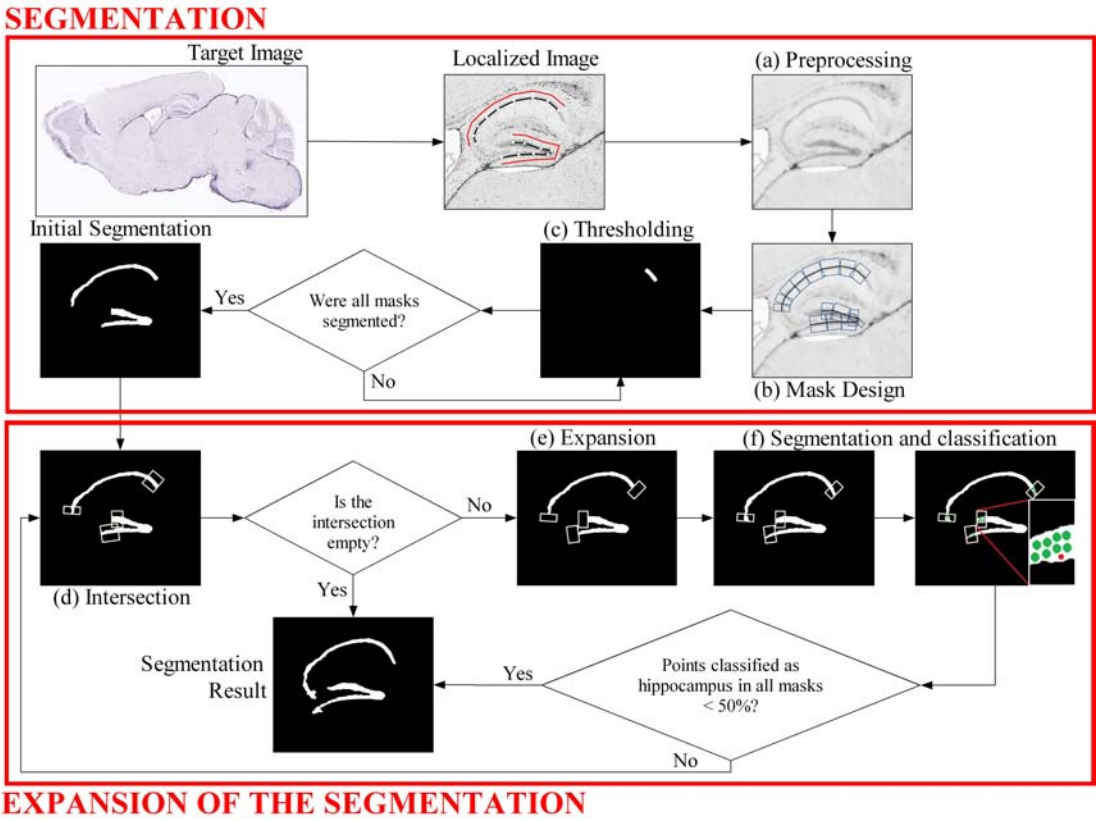


Figure 5.12: Segmentation pipeline. This pipeline starts from the localized image and finishes with the hippocampus binarized.

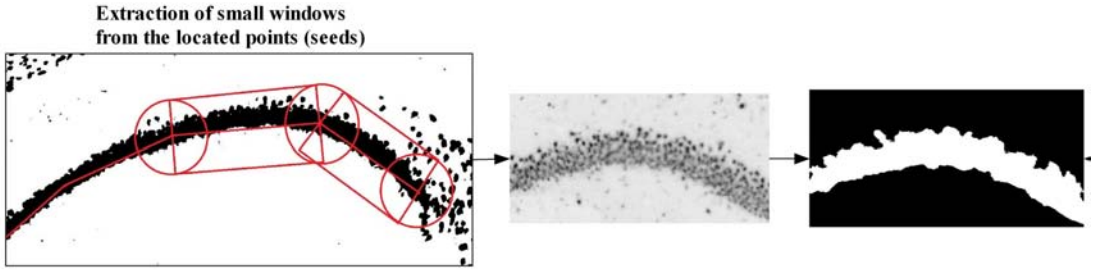


Figure 5.13: Iterative approach to Otsu's thresholding method.

2. Mask Design: we create window masks around the located points, using them as seeds. In this way, we can apply the thresholding technique at a local level, without applying it to the whole image.

   Using the located points as seeds, we can focus our interest on some image regions excluding others. We create window masks where we apply the thresholding technique at a local level, without applying it to the whole image.

3. Thresholding: We apply Otsu's method [15] iteratively on every window mask created from the seeds and, in every iteration, we keep only the largest segmented component. Therefore, the localization process can be seen as an intelligent technique for localizing areas where the segmentation method will be applied (see Figure 5.13). If we applied directly Otsu's on the ROI, or over all the masks at the same time, the results would be clearly unsatisfactory (see Figure 5.14).

One of the main advantages of this method is that, since we are using windows and evaluating a larger space (not only the located points), we can overcome the problem of points lying outside the hippocampus: the points in the model are not necessarily part of the final segmentation.

Expansion of the Segmentation

Two main problems may affect segmentation results: situations in which the located points do not overlap with the hippocampus, and situations in which the hippocampus

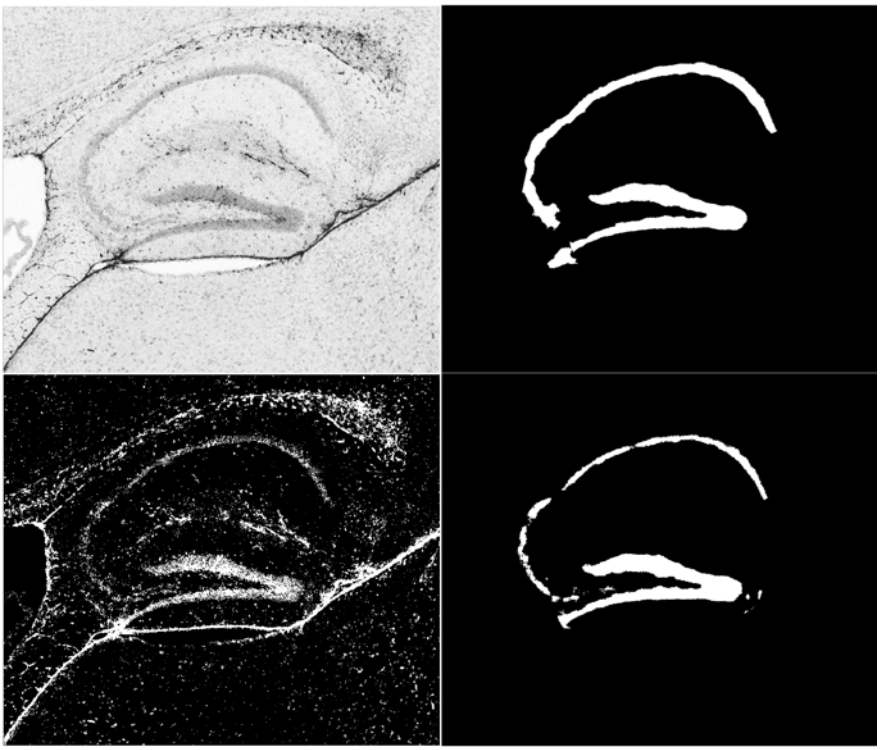


Figure 5.14: Original image (top left); result using only Otsu's over the whole ROI (bottom left); Otsu's over all masks at the same time (bottom right); segmentation after applying the proposed method (top right).

is not completely covered by the final located model. Even if experiments on a test set of 320 images yielded a perfect (all points over the hippocampus and covering it completely) or good localization (the points do not cover it entirely or, at most, two points are slightly outside it) in 90.9% of cases, we can still improve our results by introducing a method to detect/classify which points belong to the hippocampus, in order to extend and refine the segmentation to regions not yet included. Since we would like to know if the point under examination belongs to the hippocampus or not, we are talking about a classification problem.

In order to select the most adequate classifier we tried several of them: starting from the simplest ones (like Naive Bayes, 1-Nearest Neighbour or DT) and following with more advanced techniques (like RF, Support Vector Machines, Multi Layer Perceptrons and Adaboost).

The training set was formed by 189 images from our database. For each image, 20 points over the hippocampus and 20 points outside the hippocampus were selected. In total, 7560 patterns were used during the training process (employing a 5x2 cross-validation). After training, the best models achieved were tested using 1200 patterns. Such a test set was extracted from 30 completely different images, downloaded randomly from the ABA, selecting, for each image, 20 points within the hippocampus and 20 points outside the hippocampus. All these patterns were encoded as a vector of 11 textural features (first order measures: mean, standard deviation, skewness, kurtosis, entropy, coefficient of variation and energy; second order: contrast, correlation, energy and homogeneity from the GLCM using (1,1) as spatial relationship) employing 5 window sizes (30×30, 90×90, 150×150, 210×210 and 300×300).

RF (with 500 trees and 7 variables randomly sampled as candidates at each split) obtained better results than the other methods, achieving an accuracy and a false positives rate of 97.75% and 1.75%, respectively. Once our classifier is trained we can use it to extend the segmentation towards the parts that have not been taken into consideration yet (top right of Figure 5.15). This expansion acts as a region growing method, that first detects the intersection between the boundaries of the window masks and the segmented region (bottom left of Figure 5.15). Then we create a new window

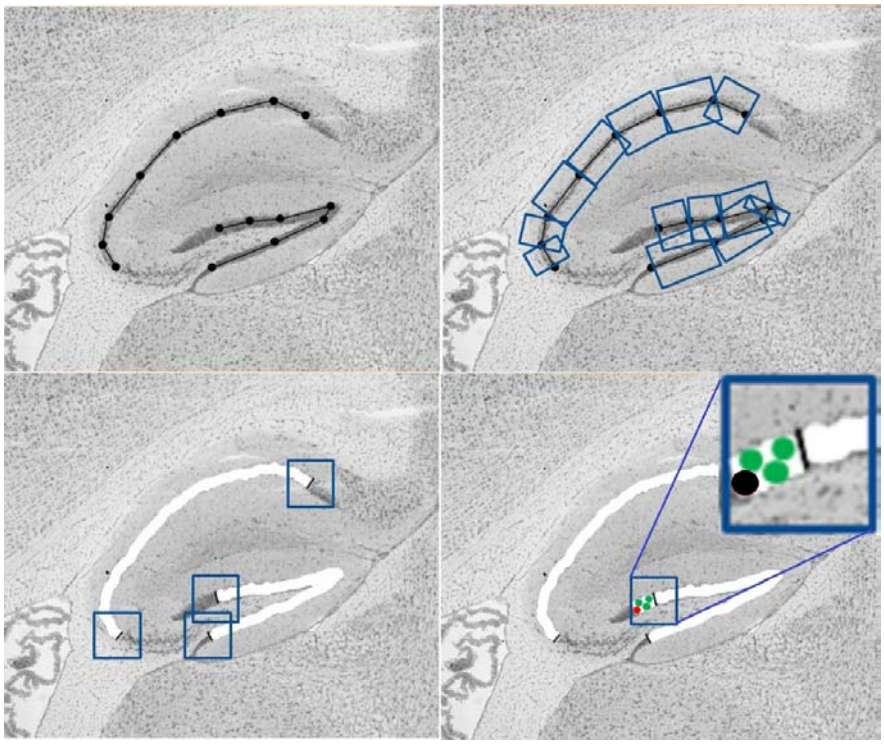


Figure 5.15: Expansion of the segmentation.

mask centered in each intersection. After that, this new window is segmented, and several random points are selected and classified (bottom right of Figure 5.15) in the new segmented part. If the majority of the points are classified as hippocampus, this new segmentation is accepted. The expansion of the window mask will continue until no further step is possible, i.e. there are no intersections between any window mask and the segmented areas, or all new segmentations are discarded by the classifier. It is important to notice that this refinement process is used only for including new segments to an incomplete segmentation, and not for removing bad segments in the original segmentation.

	Synthetic Images						**Real Images**					
	Segmentation			**Expansion**			**Segmentation**			**Expansion**		
	Avg	**Median**	**Std**	**Avg**	**Median**	**Std**	**Avg**	**Median**	**Std**	**Avg**	**Median**	**Std**
	Results obtained using only one reference (recommended by the Atlas)											
TP (%)	82.19	83.37	12.50	87.41	88.59	10.27	84.01	85.39	9.26	88.68	89.93	8.28
FP (%)	8.03	7.82	6.34	8.51	8.37	6.63	18.85	18.35	8.35	19.25	18.74	8.02
Time (sec)	5.85	5.82	0.69	8.24	7.90	3.64	13.05	13.01	3.47	16.17	15.52	7.64
Wilcoxon test	p-value (TP): 9.48E-216, p-value (FP): 4.15E-135						p-value (TP): 7.90E-220, p-value (FP): 4.81E-078					
	Results obtained using all the references in the Atlas											
TP (%)	88.38	89.17	6.31	92.11	92.38	5.13	88.57	89.33	7.30	92.25	93.46	6.98
FP (%)	11.01	10.64	5.88	11.08	10.82	6.09	22.12	22.02	9.51	22.29	21.85	9.70
Time (sec)	36.39	36.49	5.36	7.92	7.89	2.98	51.52	51.56	4.73	15.79	14.65	9.20
Wilcoxon test	p-value (TP): 3.19E-033, p-value (FP): 7.42E-005						p-value (TP): 1.84E-022, p-value (FP): 2.37E-001					

Table 5.3: Results of segmentation with synthetic and real images.

Experimental Results

Influence of the expansion process

In order to study the behavior of this method, tests over real and synthetic images were run on an Intel® Core™ i5-2410M CPU @ 2.30GHz with 4.00 GB of RAM. Table 5.3 reports the results obtained using synthetic and real images employing 25 runs per image (375 independent executions). In the experiments, true positives (TP) refers to the percentage of pixels correctly segmented and the false positives (FP) represent the percentage of pixels that should have not been segmented.

The first two rows in Table 5.3 show statistical information about the results obtained in terms of true positives (TP) and false positives (FP), respectively. Such statistics

are the average on all runs, the mean of the medians computed on each image, and the standard deviation on all runs. The third row reports the same statistics applied to the execution time, while the last one shows the p-value obtained using the Wilcoxon signed-rank test, with a level of confidence of 0.01, for the Null-Hypothesis "There are no differences between the median TP/FP percentages obtained with and without expansion".

We firstly tested our approach on a synthetic version of the problem using 15 images, to evaluate the performance of our system when dealing with noisy images with several artifacts. In these images, the hippocampus is formed by small circles of random radius and color. Small and big ellipses (between 1000 and 3000) were included trying to simulate cells, and gaussian (mean $\in$ [0.0, 0.2] and variance $\in$ [0.01, 0.15]) and salt and pepper noise (density $\in$ [0.05, 0.25]) were added to introduce fuzziness into the images.

The p-values obtained were less than the level of confidence, giving a statistical proof that the results with the expansion were better than the ones obtained using only segmentation, which supports the idea of introducing such a step in the segmentation pipeline.

Regarding real images, a ground truth image was created by manually segmenting the hippocampus in 15 significant images. In order to avoid erroneous or incomplete manual segmentations, these were supervised by an expert in molecular biology. Every image was manually segmented 5 times and, for each group of 5 manual segmentations, the intersection and union images were calculated. The difference between the maximum (union) and minimum (intersection) area segmented in equally biologically valid segmentations was computed and showed an average difference of 25.26% with a standard deviation of 3.01%. Hence, two segmentations of the same image can differ in around 25% of the segmented area while being both "correct". In all these experiments, the image considered as reference (ground truth) was the intersection of the five manual segmentations.

As in the previous case, the best results were obtained including the expansion phase. Although the percentage of FP may seem very high (around 20%), it is necessary

to consider the intrinsic uncertainty mentioned in the previous paragraph and that the ground truth used was the intersection image, that considers a smaller area as gold standard. Furthermore, the best results were obtained giving freedom to the system to choose the reference to use, instead of using the one indicated by the ABA, proving the effectiveness of our method to localize the hippocampus and its ability to select the best references and models to be used.

Comparison with other segmentation methods

A wide and representative image set was collected (see 5.16) to run experiments on the hippocampus, and compare the segmentation performance of ASM+RF with respect to other segmentation methods. Seven approaches, from deterministic to non-deterministic ones, and from recent trends to classic computer vision techniques, were compared using different standard metrics.

The methods included in the comparison were the following:

- Active Shape Models (and Iterative Otsu Thresholding Method) refined using Random Forests (ASM+RF).
- Soft Thresholding (ST) [16]. This method, presented in 2010, is based on relating each pixel in the image to the different regions via a membership function, rather than through hard decisions, and such a membership function is derived from the image histogram. As a consequence, each pixel will belong to different regions with a different level of membership.

  In a first stage, the normalized histogram of the image under study is calculated (using the maximum grey level to bound the histogram to the interval [0, 1]). Then, a sum of weighted known distributions is fit to the histogram, and each probability distribution represents the probability for a pixel with a certain value to belong to the corresponding region.

  This segmentation technique has the following advantages: (1) it is totally automatic, and does not require human intervention, which makes it suitable for

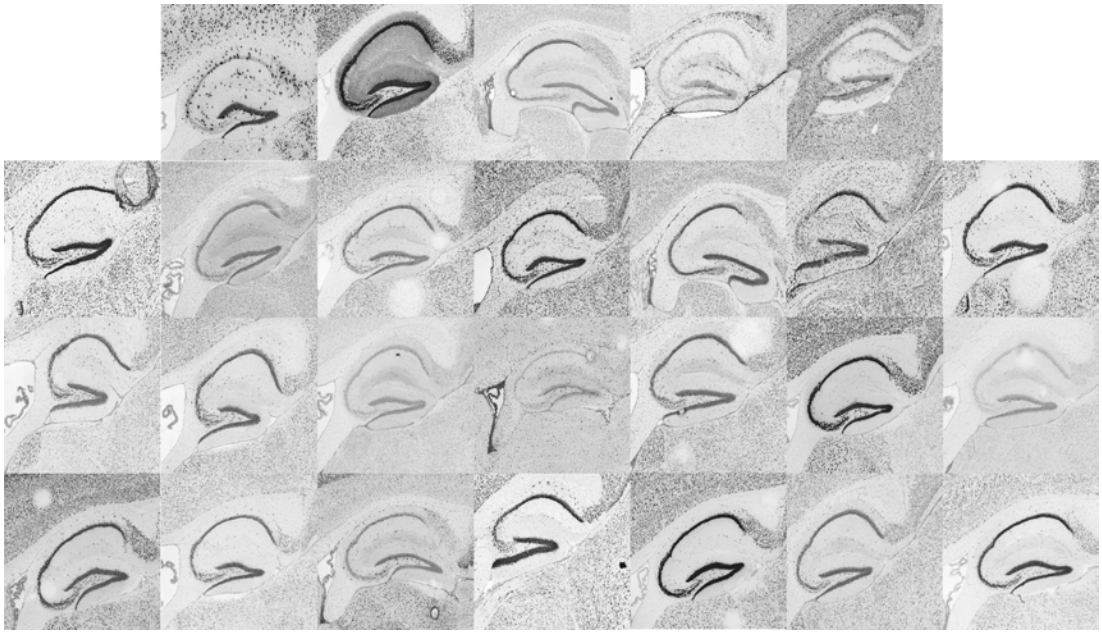


Figure 5.16: All images used in the experiments. From up to down, and from left to right: Gad1, Camk2a, 0610010D24Rik, Gfap, Mbp, B230215L15Rik, 3830406C13Rik, A030009H04Rik, Gapdh, 5430437P03Rik, Atp1b2, Tubb3, Azin1, 1300018I05Rik_94, Trem1, Cd9, Wars, Slc17a7, Zim1, Atrx, 1300018I05Rik_86, Cutl2, Camk2g, Camk2b, Wbp7, and Nmt1.

automatic processes; (2) the hard decision is postponed to the final stage. So, all the spatial operations are performed before taking into account the different memberships and obtaining the thresholded image; and (3) the spatial operations performed make the thresholding more robust to noise and artifacts.

Having been succesfully applied to CT, MRI and ultrasound, it seemed interesting to apply it also to histological imaging, to check its performance with this image modality.

- Otsu Thresholding Method (OTSU) [15]. This method, introduced in 1979, was proposed from the viewpoint of discriminant analysis, and it automatically selects an optimal threshold. Otsu's thresholding method involves iterating through all possible threshold values and calculating a measure of spread for the pixel levels on each side of the threshold. The aim is to find the threshold value that minimizes the sum of foreground and background spreads, i.e. it tries to minimize the within-class-variance by maximizing the between-class-variance (which is less computationally intensive to calculate).

  The algorithm assumes that the image to be thresholded contains two classes of pixels, or that its histogram is bi-modal (e.g. foreground and background), and then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal. Since ASM+RF uses Otsu in an iterative and local way, it also seemed interesting to test the proper Otsu method directly over the whole ROI, in order to check the improvement obtained by the ASM+RF method when including prior shape knowledge, as well as texture information.

- Geodesic Active Contours (GAC). See 2.2 for a description of the method.

  Two implementations of GAC have been tested. The first one uses as initial contour the whole image, while the second one, called GAC+ASM, employs the localization method of ASM+RF (i.e. an ASM using DE) to create the initial contour of the geometric DM. Except the initial contour, all the remaining parameters were exactly the same in both approaches.

- Chan&Vese Level Set Model (CV). See 2.2 for a description of the method.

  Also in this case, like in GAC, two implementations have been tested. The first one uses the whole image as initial contour, and the second one employs the localization results obtained by ASM to create the initial contour.

It is important to notice that all methods used in this experimentation, apart from ASM+RF, and GAC and CV when using ASM to obtain the initial contour, are deterministic techniques, i.e. given a particular input, they will always produce the same output. This is not the case of the other three approaches, since stochastic methods, like DE, are embedded in these algorithms, and can exhibit different behaviors on different runs. Due to this, it is essential to run such algorithms several times to estimate and compare their performances.

In order to evaluate the accuracy of all methods, tests on real images were run on an Intel® Core™ i5-2410M CPU @ 2.30GHz with 4.00 GB of RAM, using MATLAB as programming language. Table 5.5 reports the results obtained using 26 real images, employing 25 runs per image (for non-deterministic methods), for a total of 2054 experiments. The results are organized per blocks of five rows each. Every block corresponds to a different metric [201]: Dice Similarity Coefficient, Hausdorff Distance, Jaccard Index, True Positive Rate, and False Positive Rate (see Appendix II for more information). Inside every block, the average, median, standard deviation, best value, and worst values are shown. These calculations are computed over 26 or 650 values depending on the nature of the method under consideration (deterministic or non-deterministic, respectively).

A ground truth was created by manually segmenting the hippocampus in these 26 images. In order to avoid erroneous or incomplete manual segmentations, these were supervised by an expert in molecular biology. Every image was manually segmented 5 times and the intersection and union images were calculated. Nevertheless, the difference between the maximum (union) and minimum (intersection) area segmented in these (equally valid) segmentations showed an average difference of 28.44% with a standard deviation of 5.61%. Simultaneous truth and performance level estimation

(STAPLE) [202] was used to create a consensus ground truth, since it is considered one of the state-of-the-art methods to create a ground truth from several manual segmentations [203].

Table 5.4: Parameters used in testing different segmentation techniques.

ST	OTSU	ASM+RF	CV	GAC
L = 2 regions Relative max norm.	No params.	$Cr = 0.9$ $F = 0.7$ Uniform Crossover DE/target-to-best/1 Population Size = 80 Iterations = 250 Median Filter [25×25] RF with 500 trees	num. iter. = 500 μ = 0.1 (length term) ν = 0 (area term) $\lambda_1 = \lambda_2$	num. iter. = 500 β = -1 (expansion weight) α = 3 (contour weight)

A post-processing phase was applied to remove isolated pixels: a median filtering (15x15), only in the case of ST and Otsu, and the removal of connected components with less than 5000 pixels in all methods, except ASM+RF. The manually tuned parameters used on each method are shown in Table 5.4. For GAC, CV, GAC+ASM and CV+ASM, a median filtering (10x10) was applied during pre-processing in order to regularize the surface of the image.

The test set is composed of 26 representative genes downloaded from the ABA database (see Figure 5.16). In particular, the images downloaded, and used for the comparative study, were the ones associated with the following gene symbols: Wbp7, 1300018I05Rik_86, 0610010D24Rik, 1300018I05Rik_94, 5430437P03Rik, Atrx, B230215L15Rik, Nmt1, Wars, Tubb3, Trem1, 3830406C13Rik, A030009H04Rik, Atp1b2, Azin1, Camk2a, Camk2b, Camk2g, Cd9, Cutl2, Gad1, Gapdh, Gfap, Mbp, Slc17a7, and Zim1.

In order to check the statistical significance of the results obtained, a Kruskal-Wallis test was performed with a level of confidence of 0.01. Since the normality and homoscedasticity assumptions were not accomplished, as checked through the application of Kolmogorov-Smirnov and Bartlett's tests, non-parametric tests were used. The p-value is near zero, suggesting that at least one sample median is significantly different from the others.

Some conclusions can be extracted from the numerical data in Table 5.5 and from

Table 5.5: Segmentation Results using 5 different metrics: Dice Similarity Coefficient (DSC), Jaccard Index (JI), Hausdorff Distance (HD), True Positives (TP), and False Positives (FP). The best results for every metric are shown in bold letters.

		Methods						
		OTSU	ST	CV	CV+ASM	GAC	GAC+ASM	ASM+RF
DSC	Average	0.5755	0.6798	0.6164	0.6488	0.6373	0.7469	**0.8882**
	Median	0.5788	0.7505	0.6906	0.7577	0.6985	0.7971	**0.8975**
	Std	0.1820	0.2276	0.2432	0.2052	0.2723	0.1993	**0.0347**
	Best	0.8341	0.9084	0.9120	0.8909	0.9189	0.9198	**0.9350**
	Worst	0.2868	0.0687	0.2183	0.2358	0.0000	0.0214	**0.7422**
JI	Average	0.4263	0.5531	0.4869	0.5121	0.5205	0.6283	**0.8006**
	Median	0.4090	0.6007	0.5274	0.6099	0.5367	0.6626	**0.8141**
	Std	0.1823	0.2330	0.2465	0.2117	0.2782	0.2080	**0.0541**
	Best	0.7154	0.8321	0.8382	0.8033	0.8500	0.8514	**0.8779**
	Worst	0.1674	0.0356	0.1225	0.1336	0.0000	0.0108	**0.5901**
HD	Average	936.2499	660.4281	566.7963	397.8478	543.2649	247.6652	**110.8424**
	Median	988.7435	723.4520	602.9961	378.8250	565.7718	212.2499	**93.9415**
	Std	220.5734	354.8354	340.3507	253.4867	392.5249	174.1109	**47.6210**
	Best	506.7790	57.0088	33.5410	40	67.0820	**30**	44.7214
	Worst	1.3837e+003	1.3911e+003	1.1942e+003	1.0330e+003	1.3067e+003	721.9591	**296.0152**
TP	Average	89.4691	80.0974	95.6464	**97.2764**	85.1330	82.4358	87.3440
	Median	95.4543	90.2218	98.5369	**99.2808**	97.4342	95.2620	87.7858
	Std	15.5791	23.5970	9.0898	**6.8649**	25.9801	25.5602	4.6220
	Best	99.9584	99.9659	**100**	**100**	99.9364	99.9364	94.9942
	Worst	35.8155	5.7835	**64.1182**	**64.1182**	0.0000	1.3607	60.5404
FP	Average	54.7366	34.2408	50.3108	47.9808	41.3426	28.1579	**9.2846**
	Median	58.7041	26.9258	46.5393	38.9488	27.4716	22.1582	**7.8321**
	Std	19.3433	26.0209	25.0742	21.5470	30.7668	19.4360	**5.5229**
	Best	25.6864	**1.5257**	11.6259	16.0377	7.8152	3.6460	3.0877
	Worst	81.2674	91.5339	87.7290	86.5843	100	95.0183	**32.4049**

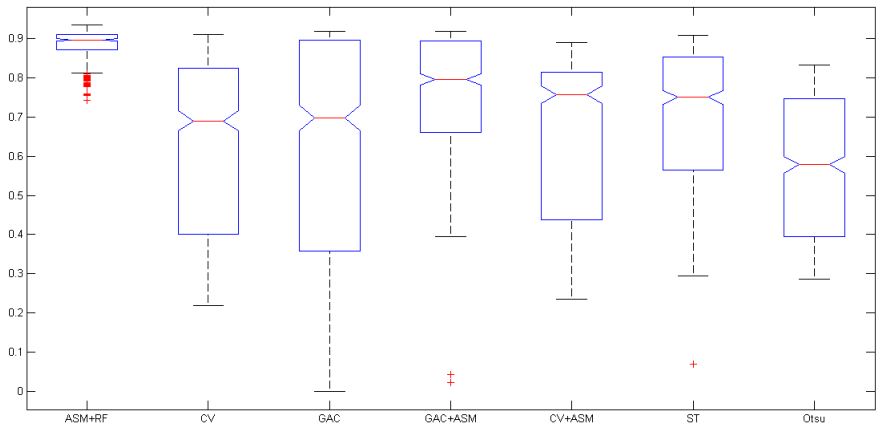


Figure 5.17: Box-plot representing the DSC for all methods. The best results are obtained by ASM+RF.

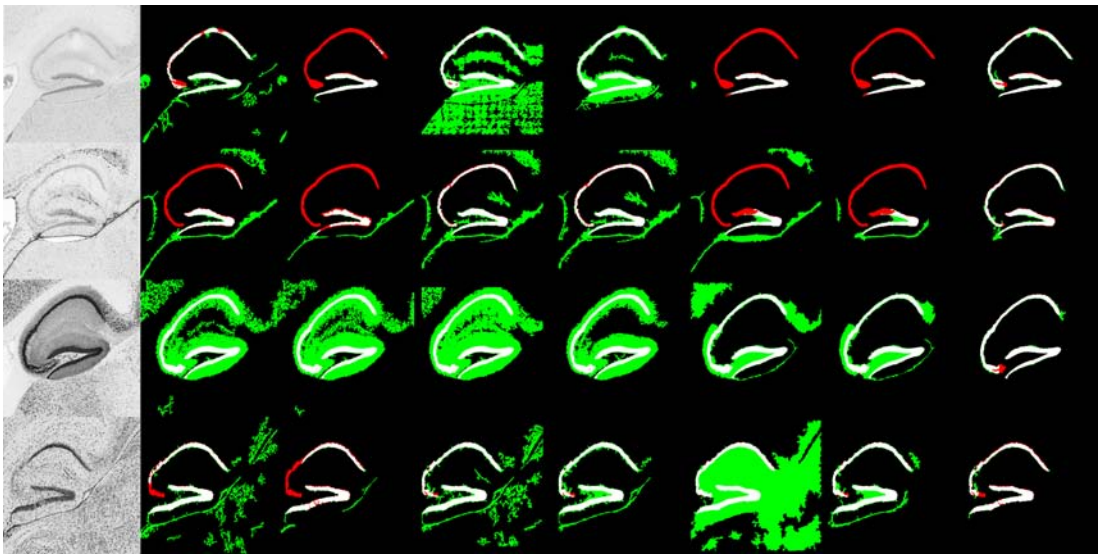


Figure 5.18: Results for some images (from up to down: Zim1, Gfap, Camk2a, and Atp1b2). The order of methods is (from left to right): OTSU, ST, CV, CV+ASM, GAC, GAC+ASM and ASM+RF. True Positives are represented in white, False Positives in green and False Negatives in red.

some of the representative results included in Figure 5.18. The first conclusion could be the difficulty of tackling satisfactorily these images, since established and well-known segmentation methods, like CV and GAC, did not obtain as good results as one could expect. In particular, GAC presents problems with images that have small variations of their gradient values, i.e. when the difference, in terms of intensity, between the hippocampus and the background is low.

According to Table 5.5, the best results were obtained by ASM+RF, being the best in all metrics but in TP rate, where CV, CV+ASM and Otsu achieved better results. This is explained by the tendency of these three methods to oversegment the images and, therefore, to segment the hippocampus and many other regions in the image. In general terms, the best methods to segment these images have been ASM+RF, CV+ASM, GAC+ASM, and ST.

From the values of HD interesting conclusions can also be drawn. This metric represents the largest of all the distances from a point in one set (the ground truth) to the closest point in the other set (the result of the automatic segmentation). It is interesting

to check how ST has a higher (worst) HD with respect to CV, even if it obtained better DSC results than the latter. This could be justified by the absence in the former of any kind of shape restrictions or regularization terms, that makes that the pixels are segmented only taken into account their grey level values.

On the other hand, Otsu and ST are the fastest methods since they work only, and directly, on the histogram. The difference in performance between Otsu and ASM+RF demonstrate the great improvement obtained with the introduction of prior knowledge.

Geometric DMs have shown a good ability of changing topology and evolving naturally the contour, but on the other hand, they are very sensitive to initialization, pre-processing stages[4], and parameter selection. GAC and CV suffer with bad initializations (the use of a better initial contour favors a fast convergence to the contour and better results), but this affects specially to GAC (as can be cheked in the improvement obtained when using the localization method based on ASM). If the initial contour is located partially inside and outside of the structure of interest (as sometimes happens due to a bad localization), only the boundaries will be segmented and not the internal part of the hippocampus, because the high gradient values are located on the boundary and not in the internal part of the structure.

Regarding both geometric approaches (CV and GAC, with and without localization stage), the results would improve in great manner if statistical shape priors for level sets were used (thickness, length, statistical information about the image to segment,...). The only piece of prior knowledge that GAC and CV are using now is the existence of high gradient values in the boundary of the objects to segment, and the prior knowledge included in the parameter selection (weighting the different components of the energy). Another aspect to take into account is that, sometimes, the inner part of sg is completely segmented without separating the two extremes of this region. This owes to the fact that the regularization factor does not allow the evolving contour to "pass" through small narrow regions and, therefore, it fills completely the empty space of sg. Furthermore,

[4] A brief example to show the influence of the type of pre-processing applied. If the size of the median filter were 20x20 instead of 10x10 the results would get worse for CV (DSC Mean: 0.586159 — DSC Median: 0.639210 — DSC Std: 0.256346) and they would improve for GAC (DSC Mean: 0.668008 — DSC Median: 0.841995 — DSC Std: 0.344408)

the problem with GAC and CV in noisy images is that, if the images are smoothed too much, it will be impossible for the method to stop at the gradients, but if they are not smoothed, the noise would provoke the segmentation of many noisy parts in the image.

As previously said, ASM+RF obtained the best results, but it is fair to underline its ad-hoc nature. It needs a training set of shapes to create the template and its possible deformations, and it also needs a training set of textural patterns for the expansion phase. Also it is not able to manage topological changes in a natural way, as geometric DMs can do. So, it is a very accurate method but it sacrifices its general applicability to segment other shapes.

5.4 Real-world application

The image analysis pipeline described in previous sections was tested in a real-world application for the identification of synapse-enriched RNAs from brain ISH data. Such application, developed in collaboration with the Molecular Biotechnology Center of Torino (`www.mbcunito.it`), after bioinformatics validation, showed to be very effective in the identification of RNAs that are known to be enriched in spines and to play a role in synaptic function and potentiation. In addition, experts from the same research center experimentally validated the approach by identifying two non-coding transcripts enriched in mouse synaptosomes.

The DM control points were used to locate different areas of interest within the hippocampus containing both cell bodies and neuropil. They were approximately centered on the region of maximum curvature of the CA1 region and of the CA3 region in the Stratum Pyramidale (sp) of the Ammon's horn (AH), and in the medial half of the Stratum Granulosum (sg) for what concerns the dentate gyrus (DG) (see Figure 5.19). Afterwards, such areas were segmented using the approach previously explained (see Section 5.3).

After segmenting the different regions, 220 textural features of first and second order (using GLCM) were extracted by each region. The size of the windows for each

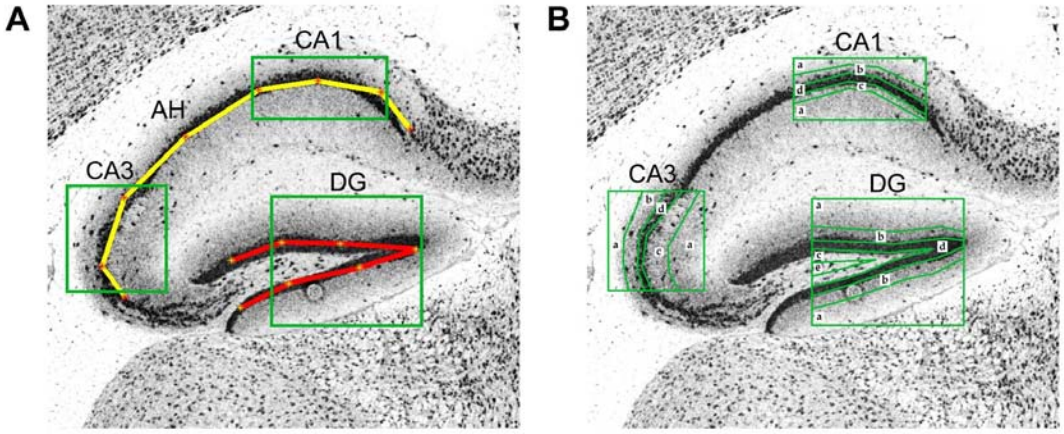


Figure 5.19: Graphical illustration of the localization and segmentation steps. (A) The deformable model of the Ammon's horn (AH) and of the dentate gyrus (DG) are represented by a yellow line connecting red stars and by a red line connecting yellow stars, respectively. Selection of the regions of interest is represented by green boxes. (B) The different regions of interest are explicitly indicated.

region, as well as the textural features used, are reported in Table 5.6. After that, a small dataset composed of 20 genes was used to train the system. The idea was to select a feature subset able both to represent well the dendrite-enriched mRNAs and to successfully distinguish them from negative examples. To do so, we used as positive examples (prototypes) three genes which are well known for the dendrite and spine enrichment of their mRNA, namely Camk2a (ID=12322), Map2 (ID=17756) and Arc (ID=11838). 17 genes were selected as negative examples with different characteristics:

- Three genes whose mRNA is specifically expressed in neurons but is not transported in dendrites: CaMKIIb (ID=12323), Tubb3 (ID=22152) and Grin1 (ID=14810);
- Three ubiquitous metabolic enzymes whose mRNA is strongly expressed in neurons but is not transported in dendrites: Gapdh (ID=14433), Pgk1 (ID=18655) and Pfkm (ID=18642);
- Two genes expressed in glial cells, particularly in astrocytes: Gfap (ID=14580) and Slc1A2 (ID=20511);

- Three genes genes expressed in oligodendrocytes: Mag (ID=17136), Mog (ID=17441) and Mbp (ID=17196);
- Two genes expressed in GABA-ergic interneurons: Gad1 (ID=14415) and Slc6A1 (ID=232333);
- Three genes expressed in glutamatergic neurons: Slc1A1 (ID=20510), Slc1A3 (ID=20512) and Slc17A7 (ID=72961);
- A gene producing a non-coding RNA localized in the nucleus: Sox2OT (320478).

The images chosen for the above genes were reviewed by an expert, that confirmed the correspondence of the expected expression pattern with the pattern revealed by the ABA in situ hybridizations. Features that, in this small dataset, showed a very high correlation between each other (> 0.99) were considered as being the same, and one of them was removed. After this, a binary GA (population size 50, 300 generations, crossover rate 0.8, mutation rate 0.06, tournament selection with size 4) was used. Every subset of features was encoded in an individual of the genetic algorithm, and the silhouette index [204] was used to evaluate its goodness. This index is computed using the selected subset of features and considering the positive examples in one group and the negative examples in another. The silhouette index is a measure of an object's similarity to the others of the same group and dissimilarity from the elements in the other groups.

Since a GA is a stochastic MH, which may produce different solutions each time it runs, this procedure was repeated 15 times and the features selected in at least 50% of the runs were chosen. This led us to select a subset of 52 features (see Table 5.7), which were used to process all genes in the ABA and compute the Pearson coefficient between their feature vector and the prototype one, generated by averaging the features of three prototype genes. The results using the 220 features were 0.5066, 0.4302, 0.8150, while using only the selected features were 0.8610, 0.8796, 0.9441; proving the ability of the proposed feature-selection method to facilitate the detection of correlations between genes. It is important to underscore that this pipeline has general applicability and

would be able to rank genes according to their similarity to any kind of features, only by providing it with a training set including positive and negative examples.

This pipeline was used to scan the sagittal images contained in the ABA and detect similarities with the texture vectors of the prototype neuropil-enriched genes that we used for training, i.e. Camk2a, Arc and Map2. To avoid scoring genes expressed at negligible levels in adult mouse hippocampus, we only considered image series in which the expression level or the expression density reported for the hippocampal region of for the hippocampal formation was above 20, a background level determined on the basis of cell cycle genes which are known to be silent in brain after the end of development. For all the remaining genes (more than 9000), we selected para-sagittal sections corresponding to levels from 117 to 175 of the ABA reference atlas. In particular, we selected as default the level 145, which we consider as 'center' of the hemisphere, and moved towards the two boundaries if we were not able to get results with the selected slice. For each section we identified the different hippocampal regions and determined the values for the corresponding texture parameters, obtaining vectors of texture features. We then calculated the Pearson correlation coefficients of these vectors with a prototype, or reference vector, obtained by averaging the vectors of the prototype genes. Importantly, vectors obtained from sections corresponding to slightly different levels were very similar. In particular genes showing a high Pearson correlation coefficient, that provides a measure of the strength of linear association between two variables, with the prototype vector displayed a very stable behavior (some examples are shown in Figure 5.20).

Finally, coding, non-coding and ambiguous transcripts, as defined on the basis of the ABA annotation, were ranked by decreasing Pearson correlation coefficient. To evaluate the effectiveness of the ranking procedure, we first analyzed the 20 top scoring probes and, as expected, a Gene Ontology analysis performed with the DAVID software [205] revealed that the most significant common keyword associated to the corresponding genes is "dendrite" (p-value = 0.004). In particular, a manual inspection revealed that this list contains some of the best known examples of transcripts localized to dendrites and/or associated with dendritic functions: Dendrin (Dnd), Psd (also known

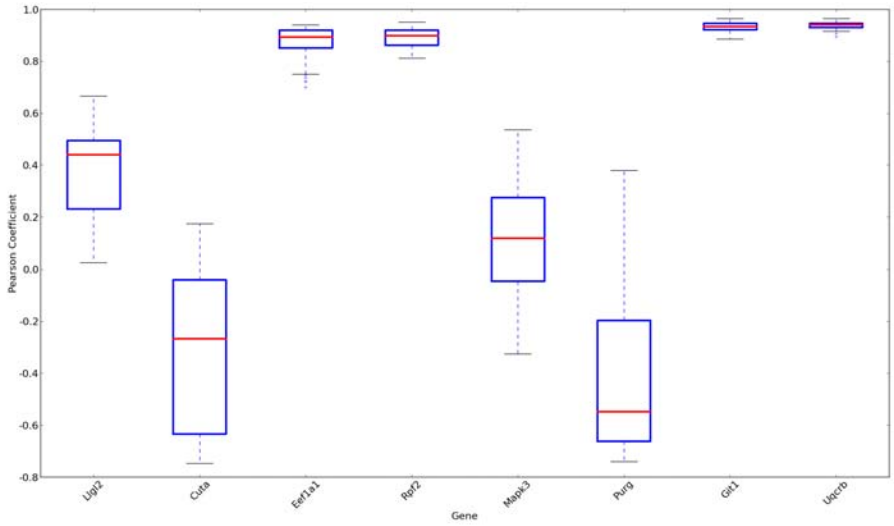


Figure 5.20: Boxplot of the correlation values of the feature vectors obtained from 6 slices of 8 randomly selected genes, as compared to the prototype vector. Note the very stable behavior of genes characterized by a high Pearson coefficient.

as Efa6a), microtubule-associated protein 2 (Mtpa2), Git1 and Spinophilin (Ppp1r9b). Accordingly, visual inspection of the corresponding ABA images confirmed a significant signal enrichment in neuropil for many of the probes, as is the case for the Rnf10 gene (Figure 5.21).

Table 5.6: Complete set of features used for the texture analysis. “Energy2” stands for second order feature Energy.

CA1a	CA1b	CA1c	CA1d	CA3a	CA3b	CA3c	CA3d	DGa	DGb	DGc	DGd	DGe
18,36	24	24	14,28	18,36	24	24	14,28	18,36	24	24	14,28	18,36
First Order	Mean, Standard Deviation, Coefficient of Variation, Skewness, Kurtosis, Energy, Entropy											
Second Order	Contrast, Correlation, Energy2, Homogeneity											

5.5 Conclusions

In this chapter, we have presented a three-step algorithm, or four-step if segmentation and refinement are considered separately, aimed at automatically segmenting the hippocampus in histological images. The first phase, described in section 5.2, roughly

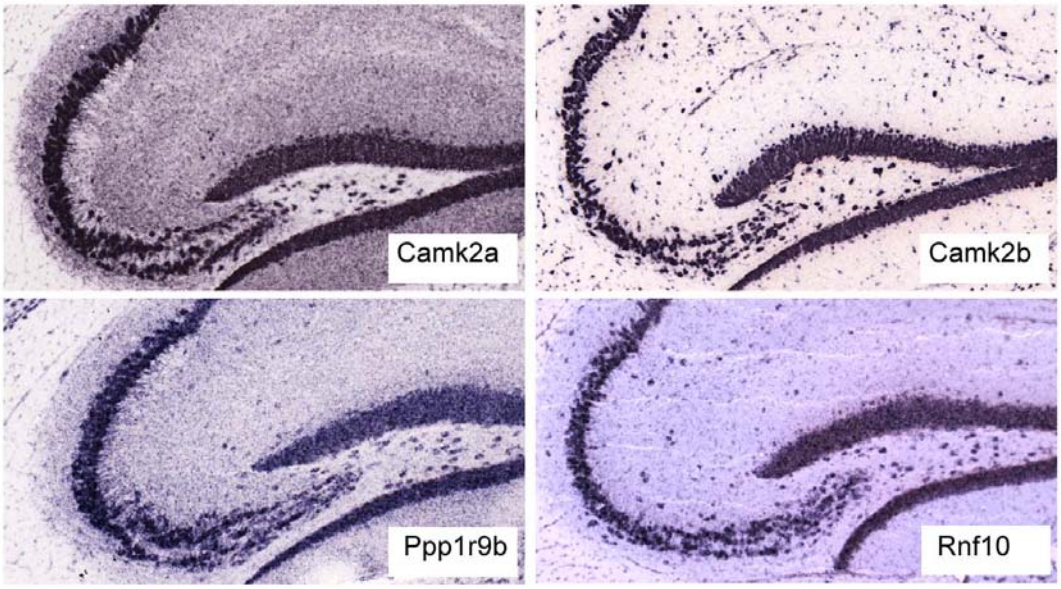


Figure 5.21: In situ hybridization pattern of the indicated protein-coding genes, obtained from the ABA.

Table 5.7: Features used to perform the ranking procedure. The number in brackets represents the size of the window on which the feature was calculated, when more than one window size was used for the region under consideration.

CA1 C Mean	CA1 C StdDev	CA1 C Skewness	CA1 C Kurtosis
CA1 C Energy	CA1 C Contrast	CA1 C Correlation	CA1 C Energy2
CA1 C Homogeneity	CA1 B Mean	CA1 B Skewness	CA1 B Kurtosis
CA1 B Energy	CA1 B Contrast	CA1 B Correlation	CA1 B Energy2
CA1 B Homogeneity	CA1 D Mean	CA1 D StdDev	CA1 D Entropy (14)
CA1 D Contrast (14)	CA1 D Energy2 (14)	CA1 D Homogeneity (14)	CA1 D Energy (28)
CA1 D Contrast (28)	CA1 D Homogeneity (28)	CA1 A Mean (18)	CA1 A Energy (18)
CA1 A Entropy (18)	CA1 A Correlation (18)	CA1 A Energy2 (18)	CA1 A Homogeneity (18)
CA1 A StdDev (36)	CA1 A Energy (36)	CA1 A Entropy (36)	CA1 A Correlation (36)
CA1 A Energy2 (36)	CA3 B StdDev	CA3 B Contrast	CA3 B Homogeneity
CA3 D Energy2 (28)	CA3 A Correlation (18)	DG C Kurtosis	DG B StdDev
DG B cv	DG B Entropy	DG B Contrast	DG B Correlation
DG E Mean (18)	DG E Mean (36)	DG A Correlation (18)	DG A Correlation (36)

locates the anatomical structure by comparing the image under consideration with images taken from the ABA reference atlas. In the second phase, the structure is more precisely localized using a DE-based parametric deformable model that adapts its shape to match the anatomical structure of interest. The method is able to deal with imprecise and incomplete images, and, in our tests on actual images from the ABA, has been successful in 90.9% of cases. Finally, in the third stage, the hippocampus is segmented from the located points using Otsu's thresholding method and the segmentation is refined by means of RFs.

Since classic gradient-based local search methods (like LM) are not able to solve this problem satisfactorily, due to its multimodality and high dimensionality, we have applied and studied different metaheuristics: DE, SA, GA, PSO, and SS. DE achieved the best results, as concerns both average and standard deviation. This result can be explained by a better balance between exploration/diversification and exploitation/intensification and upholds the results obtained in other cases by DE on benchmarks containing multimodal non-separable functions [98]. Moreover, DE has shown greater robustness: it is able to reproduce good results more consistently over many trials, whereas the performance of other algorithms, like GA or SS, is more dependent on the stochastic initialization of individuals and parameters. DE is also, together with GA, the metaheuristic that converges faster to good solutions.

The segmenting method, together with the refinement approach, was applied to real and synthetic images, obtaining an average accuracy of 92.25% in the first case, and 92.11% in the second one. For achieving this goal, the method only needs an anatomical atlas and a parametric model, associated to the atlas, representing the structure of interest. It automatically selects the most suitable reference slice, localizes the hippocampus using a parametric deformable model, and segments this anatomical structure using an iterative version of Otsu's thresholding method. For the expansion of the segmentation, when the localization was not perfect, an ensemble classifier (Random Forest) has been used. The use of the expansion increases the TP rate, keeping almost constant the FP percentage and the standard deviation. Therefore, the segmentation with expansion is a better method for tackling these images.

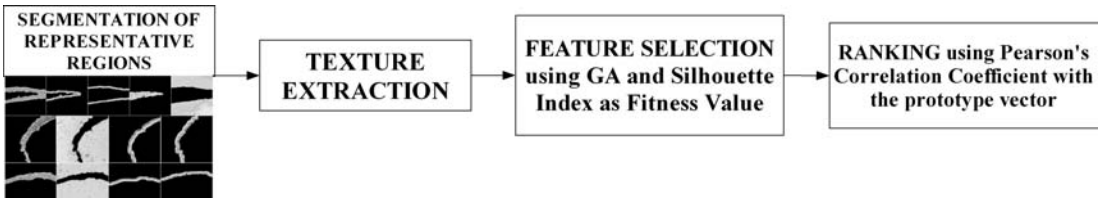


Figure 5.22: General overview of the gene-ranking method developed for the visual search of neuropil-enriched RNAs from brain ISH data.

Also, seven approaches, from deterministic to non-deterministic ones, and from recent trends to classic computer vision techniques, have been compared using different standard metrics (Dice Similariry Coefficient, Jaccard Index, Hausdorff Distance, True Positive Rate and False Positive Rate). Proper statistical tests have been performed to draw accurate conclusions about the results, and the best performance was obtained by our proposal (yielding an average Dice Similarity Coefficient of 0.89 with a standard deviation of 0.03).

Finally, a bioinformatics problem, whose aim was to find genes having features similar to those of a paradigmatic subset of genes, was effectively solved using this segmentation pipeline, feature selection using GA, and textural features ranking from ISH images using Pearson coefficient. Figures 5.23 and 5.22 give a general overview of the segmentation and clustering pipelines presented in this chapter, respectively.

In conclusion, the localization/segmentation method presented in this chapter is divided into four different stages (DM initialization, localization using DMs and meta-heuristics, segmentation, and expansion of the segmentation using ensemble classifiers), and can be seen as an intelligent manner of localizing promising areas where a fast and well-established segmentation technique is going to be applied. The main advantages of this approach are its execution time and accuracy, while its main disadvantages are its ad-hoc nature (it needs a training set of shapes/textures, as well as suitable parametric models of the object to locate), the introduction of prior shape knowledge only in the localization step (and not in the segmentation stage, allowing the existence of problems with very ambiguous regions), the impossibility of managing easily topological changes, and the manual and tedious creation of the templates of the object to find.

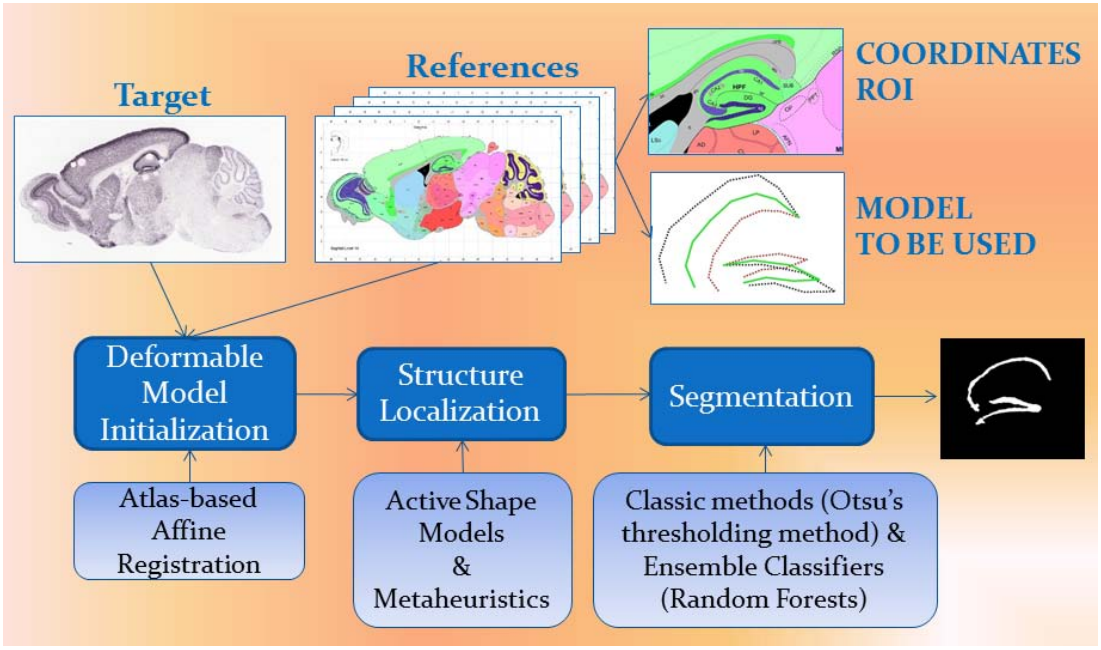


Figure 5.23: Complete pipeline of the segmentation approach including localization using ASMs and metaheuristics, segmentation using Otsu's thresholding method, and refinement by means of Ensemble Classifiers.

Chapter 6

Hippocampus Segmentation in Histological Images using a Metaheuristic-based Level Set Approach

6.1 Previous Approach

The procedure based on Active Shape Models and Random Forests (ASM+RF) [123, 152], as introduced in the previous chapter to accurately segment ABA images, obtained very promising results in comparison with many other segmentation techniques [180] but, despite its good performance, some drawbacks were also highlighted:

- impossibility to deal with topological changes in a natural manner;
- ad hoc nature, since the procedure needs a training set of shapes to manually create the parametric template and its possible deformations, as well as a training set of textural patterns for the ensemble classifier used to refine the results of the segmentation; and

- long and complex pipeline of four independent stages (initialization of the deformable model, localization of the anatomical structure of interest, segmentation, and refinement of the segmentation using classifiers). Therefore, a more integrated and compact approach would be desiderable.

The main idea now was to overcome some of these problems using another kind of deformable model (the LS method) that can: a) easily handle topological changes of the contours, b) be adapted to solve large dimensional problems without great effort, and c) easily determine the areas inside and outside an active contour.

The work developed is based on the one by Ghosh et al in [4, 135] which, in turn, was inspired by [206]. It is important to underline that this research represents one of the very few cases in which metaheuristics have been used to optimize a geometric deformable model, let alone their application to histological images, and that the current work also presents significant differences with respect to the papers mentioned above, such as:

- an intensity-based term from Chan and Vese's approach [45] has been included in the fitness function, trying to combine a region-based approach with prior knowledge about texture and shape;
- in order to adapt this approach to the particular nature of histological images, the textural part has also been critically modified: we do not use Laws' textural measures or Gabor wavelet transform-based features, like in the original paper, but textural features extracted from the Gray Level Co-occurrence Matrix, as explained in chapter 5.3;
- in relation to the genetic operators used, the single-point crossover used in [4] has been replaced by a real-coded one like the BLX-α, due to the nature of the GA chromosomes; and
- the comparison has been extended to Particle Swarm Optimization (PSO) and Differential Evolution (DE), which makes this approach one of the few examples of application of these two stochastic techniques to geometric deformable models,

and the approach has been tested on histological images (instead of usual MRI and CT).

6.2 Proposed Method

As explained before, the work developed in this chapter essentially adapts the ideas introduced by Ghosh and Mitchell in [4] to the segmentation of the hippocampus in histological images. Such an adaptation solves some of the problems of the original method and takes into account the specificities of the case under study.

On the one hand, we introduce a real-valued crossover operator conforming to the nature of our chromosomes (see Section 2.5). On the other hand, we use textural features that differ from the ones use in [4], since we have shown that the Gray-level Co-occurrence Matrix provide very good results with the same image modality [152]. Finally, the structure orientation (pose) is not taken into account during optimization because it increases the temporal and computational cost of optimization, adding a burden that is not justified by the results obtained.

The method consists of two phases: training and proper segmentation (see Figure 6.1). In the first one, from a set of manually segmented training images we compute the average shape, the main modes of variation, and the median texture, trying to characterize an "ideal" hippocampus. An implicit representation of the segmenting curve is created by applying the principal component analysis to a collection of signed distance representations of the training data. During segmentation, the MH searches the best weights to linearly combine the mean shape and shape variabilities, and the texture enclosed by the evolving contour is compared with the "representative" texture.

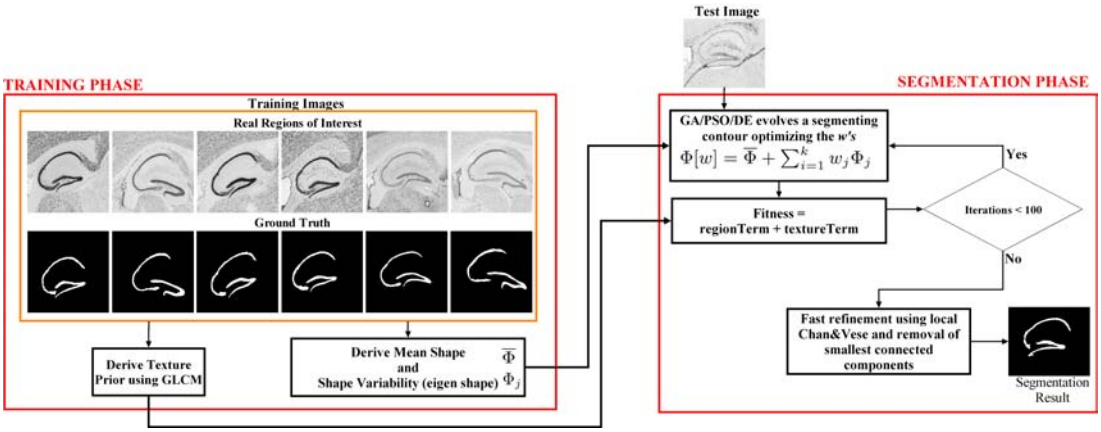


Figure 6.1: General Overview of the Hippocampus Segmentation System

Training Phase

Shape

First of all, the shape priors are derived from the training set. To do so, each contour from the training data is represented as the zero LS of the signed distance function

$$\psi_i(x,y) = \begin{cases} -d(x,y) & \text{if } (x,y) \in insideEvolvingContour \\ 0 & \text{if } (x,y) \in EvolvingContour \\ d(x,y) & \text{if } (x,y) \in outsideEvolvingContour \end{cases}$$

where $i = 1 \ldots n$, (x,y) are the pixel coordinates, n is the number of training contours used to assess shape variability, and $d(x,y)$ is the eucliden distance from the (x,y) point to the closest point in the evolving contour. The signed distance function is the shape representation: the boundaries of each shape are embedded as the zero LS of a signed distance function with negative distances assigned to the inside and positive distances assigned to the outside of the object.

The mean LS function is defined as:

$$\overline{\Phi}(x,y) = \frac{1}{n}\sum_{i=1}^{n}\psi_i(x,y)$$

Mean offset functions are then derived by subtracting the mean from the signed

distance representations of the training contours:

$$\widetilde{\psi}_i = \psi_i - \overline{\Phi}$$

Assume that the images are of size $N = N_1 \times N_2$. Given the huge size of the images under consideration, all images have been resized to $N_1 \times N_2 = 500 \times 500$ pixels. The columns of the mean offset functions ($|N| = N_1 \times N_2$) are then serially stacked to form one column vector β_i of size $1 \times N$. The shape variability matrix S (of size $N \times n$) is obtained from these n column vectors:

$$S = [\beta_1, \beta_2, \beta_3, \ldots, \beta_n]$$

The shape variance is then computed by an eigenvalue decomposition on this shape variability matrix:

$$\tfrac{1}{n} S S^T = U \Sigma U^T$$

where U is an $N \times n$ matrix whose columns represent n orthogonal modes of shape variation, Σ is an $n \times n$ diagonal matrix of eigenvalues, and the columns of $U = [u_i]$ are the corresponding eigenvectors. In this paper, instead of computing the eigenvectors of such a large matrix, we have considered the smaller one, $\frac{1}{n} S^T S$, because it is more computationally efficient [207] to obtain the n different eigenshapes $\{\Phi_1, \Phi_2, \Phi_3, \ldots, \Phi_n\}$.

After this procedure, introduced in [206], the mean shape and shape variability derived as described are used to define a LS function that implicitly represents the segmenting curve:

$$\Phi[w] = \overline{\Phi} + \textstyle\sum_{i=1}^{k} w_j \Phi_j$$

Thus, the task of the metaheuristic will be to find the values of w that minimize a fitness function to be defined in the test phase.

Texture

The textural priors used here are the same described in chapter 5.3, which yielded very good performance. The training patterns, as well as the test values calculated during

the segmentation, were encoded as a vector of 11 textural features:

- first order measures: mean, standard deviation, skewness, kurtosis, entropy, coefficient of variation and energy;
- second order measures: contrast, correlation, energy and homogeneity from the GLCM, using (1,1) as spatial relationship (i.e. $\theta = 315^{\circ}$ and d=1 pixel), employing a window of size 30×30 pixels.

The main idea is to capture the textural essence of the hippocampus and compare, during the segmentation phase, the texture enclosed by the evolving surface of our DM with the "ideal" texture of the training set. To create this median texture, p points per image in the training set were randomly selected inside the hippocampus, and t textural patterns were extracted using those points as pixels of interest. This procedure creates a $(t \cdot p) \times n$ matrix, n being the number of training images, and $t \cdot p$ the number of textural features multiplied by the number of selected points. In this case, $t = 11$ and $p = 100$. From this matrix, the median is calculated to obtain a "general representation" of the texture in the hippocampus: all points which are closer to these values are assumed to belong to the hippocampus.

Test Phase

The test phase corresponds with the actual segmentation of the object of interest. The implicit representation of the contour is deformed guided by a metaheuristic which tries to fit the boundaries of the hippocampus. To do so, such a metaheuristic stochastically generates weights to combine the mean and the variability to create new shapes.

The fitness function combines region- and texture-based terms. The former represents the CV model [45], while the latter is the euclidean distance between the texture enclosed by our contour and the median texture found in our training set.

The energy functional to minimize is the following:

$$F(i_1, i_2, C) = \alpha \cdot (\mu \cdot Length(C) + \nu \cdot Area(inside(C))$$

$$+\lambda_1 \int_{inside(C)} |u_0(x,y) - i_1|^2 dxdy$$
$$+\lambda_2 \int_{outside(C)} |u_0(x,y) - i_2|^2 dxdy)$$
$$+\beta \cdot (d(t(C), \overline{T}))$$

where u_0 is the image to segment, formed by two regions (internal and external), C is the evolving curve, i_1 and i_2 are constants depending on C (pixel intensity averages of u_0 inside and outside C, respectively), and $d(t(C), \overline{T})$ the euclidean distance between the median texture extracted from the training set ($\overline{T}$) and the actual texture enclosed by the evolving contour ($t(C)$). Using this functional, our model simultaneously takes into account intensity and texture criteria. In this case, both terms have been weighted equally ($\alpha = \beta = 1$).

Finally, a fast refinement step is applied. This refinement sequentially applies 50 iterations of a local implementation of the CV algorithm [208], that takes into account only a neighbourhood of the boundary, and the removal of connected components whose area is smaller than a threshold (in this case 500 pixels).

6.3 Experimental Results

To check the performance of the new method, 25 and 10 manually segmented images were used as training and test set, respectively. The 10 images selected as test are representative of the problem and include different scenarios (see Figure 6.2). Every segmentation method was run 15 times per image (excluding CV and GAC because they are deterministic methods and always produce the same result). The performance of GA (LS-GA), PSO (LS-PSO) and DE (LS-DE) was compared with the one obtained by two classic geometric DMs: a region-based method (CV) and an edge-based one (GAC). With respect to the population and the number of iterations used in the metaheuristic, the same configuration as the original paper was used (50 individuals, 100 iterations), and the images were resized to 500×500 pixels.

In particular, we have used the PSO version described in chapter 2.5, where the inertia factor w adapts its value to the fitness function of each particle and a particle is

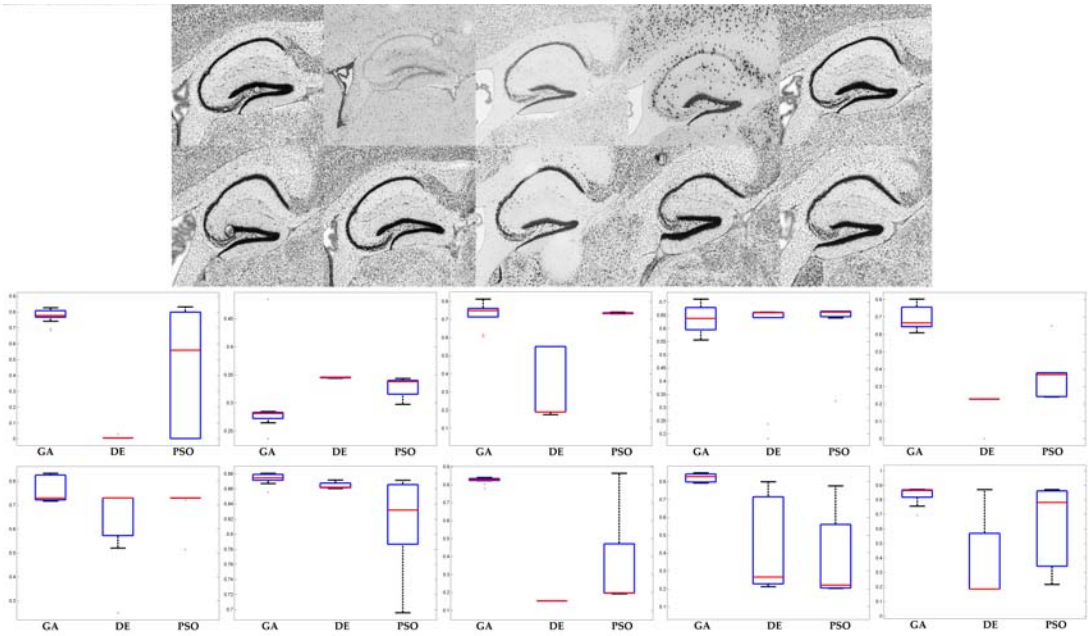


Figure 6.2: Test set used in experiments and boxplots of the Dice Similarity Coefficient results obtained per image by the stochastic methods under study. Each plot refers to the image having the corresponding position in the upper row.

re-initialized in case of stagnation.

Standard segmentation metrics, like the Jaccard similarity index (JI), the Dice similarity coefficient (DSC), and the Hausdorff distance (HD), were used to evaluate performance, and the results were analyzed using proper statistical tests. See Appendices for more information.

In order to perform a fair comparison with the deterministic/classic methods, the latter have been intensively tuned using an exhaustive search among the most commonly used parameter values over the complete dataset (training and test sets). Also, for these two methods, the input images were pre-processed using a median filter (5×5 in CV and 10×10 in GAC) while, as a post-processing stage, the removal of connected components smaller than 500 pixels was also applied.

Since the normality and homoscedasticity assumptions were not satisfied, as checked through the application of Kolmogorov-Smirnov and Bartlett's tests, non-parametric tests were used. To check the statistical significance of the results obtained, a Friedman test was performed with a level of confidence of 0.01 for the null hypothesis that all samples are drawn from distributions with the same median. Since the p-value was near zero, the statistical test suggested that at least one sample median is significantly different from the others. Pairwise statistical differences were studied using Friedman test with the Tukey-Kramer correction, and real-coded GA was found to be the best method among the ones in the comparison.

Some conclusions can be derived from the numerical data in Table 6.2 and the visual information contained in Figure 6.3. The first conclusion could be the difficulty of tackling these images satisfactorily, since established and well-known segmentation methods, like CV and GAC, did not obtain as good results as one could expect. In particular, GAC presents problems when the difference, in terms of gradient values, between the hippocampus and the background is small. In fact, between the two classic methods, the region-based approach (CV) obtained better results than the one relying on edge information. This can be justified by the fuzzy boundaries of the hippocampus that may have hampered the performance of GAC. Also, the dissatisfactory results obtained by CV can be explained by the complex background of these images, since it is well-

known that the performance of CV is better when dealing with images where foreground and background determine two regions with clearly different average intensity levels.

Considering the stochastic approaches, the best results were obtained by LS-GA, which appeared to be the best in all metrics. RCGA obtained the best results in all images but in one, while the worst results were obtained by DE. This might be quite surprising if one considers the results reported in the previous chapters. Nevertheless, this is only a preliminary study, so the convergence problems of DE and PSO with some images should be investigated more deeply. For example, the PSO topology used here has been the global best; tests with other topologies should be performed to check if there is some improvement in the results. Also, in the case of DE, tests with other crossover operators could be run to study the influence of this factor in the general performance of the segmentation algorithm. Finally, experiments with a larger population size or a larger number of iterations should be tested, considering the huge search space these techniques have to explore.

From the values of HD, interesting conclusions can also be drawn. It is interesting to check how our shape-based approach has a smaller (best) average and median HD with respect to CV and GAC. This could be justified by the absence in the latter of any kind of shape restriction, that impose that the pixels are segmented only taking into account intensity and boundary information, respectively.

Table 6.1: Parameters used in testing the different algorithms. The values for LS-GA, LS-DE and LS-PSO were based on the literature and on a brief empirical study about the suitability of different combinations of parameters.

LS-DE	LS-PSO	LS-GA	CV	GAC
$Cr = 0.9$	$w_{min} = 0.2$	$Cr = 0.9$	number of iterations = 500	number of iterations = 500
$F = 0.7$	$w_{max} = 1.0$	$BLX - 0.3$	μ = 0.01 (length term)	β = -1 (expansion weight)
Uniform Crossover	$c_1 = 2.05$	$Mut = 0.09$	ν = 0 (area term)	α = 3 (contour weight)
DE/target-to-best/1	$c_2 = 1.75$	$Tournament(4)$	$\lambda_1 = \lambda_2 = 1$	

Table 6.2: Segmentation results using three different metrics: Dice Similarity Coefficient (DSC), Jaccard Index (JI), and Hausdorff Distance (HD).

		Methods				
		LS-GA	LS-DE	LS-PSO	CV	GAC
DSC	Average	0.7204	0.4145	0.5383	0.5044	0.3966
	Median	0.7695	0.3446	0.6622	0.5294	0.2758
	Std	0.1672	0.2842	0.2614	0.1546	0.2151
	Best	0.8806	0.8716	0.8711	0.6648	0.7950
	Worst	0.2363	0.0059	0.0015	0.1107	0.2404
JI	Average	0.5871	0.3058	0.4120	0.3485	0.2710
	Median	0.6271	0.2081	0.4950	0.3600	0.1601
	Std	0.1745	0.2515	0.2378	0.1212	0.1962
	Best	0.7867	0.7725	0.7717	0.4979	0.6597
	Worst	0.1340	0.0029	0.0007	0.0586	0.1366
HD	Average	420.8674	617.6873	549.0358	762.4849	622.4544
	Median	176.0064	372.8288	338.5550	553.3118	507.1295
	Std	518.0279	549.3174	541.2218	354.1919	242.5857
	Best	28.3456	45.0627	45.0627	403.6087	395.1266
	Worst	1.7878e+003	2.1191e+003	1.7196e+003	1.4290e+003	1.0341e+003

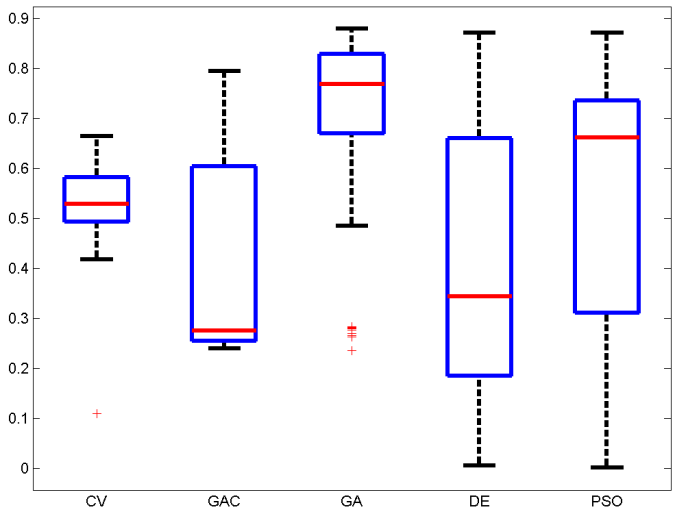


Figure 6.3: Box-plots representing the global DSC results obtained by the five methods compared over the whole dataset.

6.4 Conclusions

This chapter has investigated hybridizations between metaheuristics and the level set method, and represents a first approach to the segmentation of histological images using eigenshapes, whose linear combination is optimized using PSO, DE and real-coded GA. On the one hand, metaheuristics can optimize the energy function of the deformable model or find the most suitable parameters for such a model. On the other hand, the level set method represents an elegant solution to the main drawbacks of parametric deformable models, like the possibility of managing topological changes in a natural manner. With these concepts in mind, we have studied ways of using a training set of shapes and textures to solve a difficult problem, like segmenting the hippocampus in histological images, and compare different global search optimization techniques. Finally, this work represents one of the very few cases in which PSO and DE have been used to optimize the level set method, and also one of the few examples of application of geometric DMs to the segmentation of histological images.

It is important to remark that the work presented here is an initial approach that should be refined (using a broader training set, improving the extraction of textural features, or testing a different fitness function) and optimized (maybe using GPGPU programming). First, it is slow compared to classic approaches based on only one feature (intensity, edge), mainly due to the calculation of the textural features per individual and iteration[1]. Obviously, if someone is looking for a fast method this would not be the best option: what this method can offer is accuracy. For instance, segmentations obtained by LS-GA have a median Dice coefficient which is three times better than GAC. Also, a more sophisticated use of texture could be taken into account (for example, differentiating the two structures of the hippocampus), or including more prior information and restrictions (for instance, the relative position of the hippocampus with respect to other organs). In any case, this initial approximation has shown good performance using three different standard metrics. The results obtained have been

[1]In order to give some reference in this regard, the average time to perform one run with our metaheuristic-based approach on a MacBook Pro i7 dual core @ 2,7GHz with 8GB of RAM, and using MATLAB as programming environment, it is around 10 times slower than CV and GAC, that take approximately 5 minutes.

better than the ones yielded by very well-established segmentation algorithms, but worse than the previous approach (see chapter 5.3) and with a larger execution time. Thus, it can be seen as an intermediate stage that will lead us to the development of the next segmentation method.

Chapter 7

Biomedical Image Segmentation using Geometric Deformable Models and Metaheuristics

As explained in the Theoretical Background (section 2), a single source of prior knowledge is usually not enough to satisfactorily tackle medical image segmentation problems. Therefore, the development of hybrid approaches combining different sources of information has been a major focus in the field of image segmentation [209–211]. In this chapter, the search/learning abilities of metaheuristics and the capability of geometric deformable models to handle topological changes are combined. Three sources of information (a region term, a shape prior, and an edge term) are used to accurately segment the organs of interest in different medical image modalities: microscopy, X-ray computed tomography, and magnetic resonance imaging. In our proposal, metaheuristics [212] have capital importance in two stages. First, during the training process of the new model, the tuning of the parameters is carried out by a Genetic Algorithm [88]. Second, in the proper segmentation stage, the shape prior is obtained by a deformable registration process guided by Scatter Search [102].

Every image modality has its own peculiarities, thus the training phase allows our model to learn the most suitable parameters for a specific modality/anatomical district

using few images as paradigmatic examples. In turn, the segmentation phase uses only one manually segmented reference image to generate the prior shape knowledge that will guide, together with the region- and edge-based terms, the evolution of the moving contour.

To assess the quality of the new approach, we developed an experimental comparison between seven state-of-the-art segmentation methods. The study was carried out on four different datasets, for a total of 22 microscopy, 11 MR, and 5 CT images.

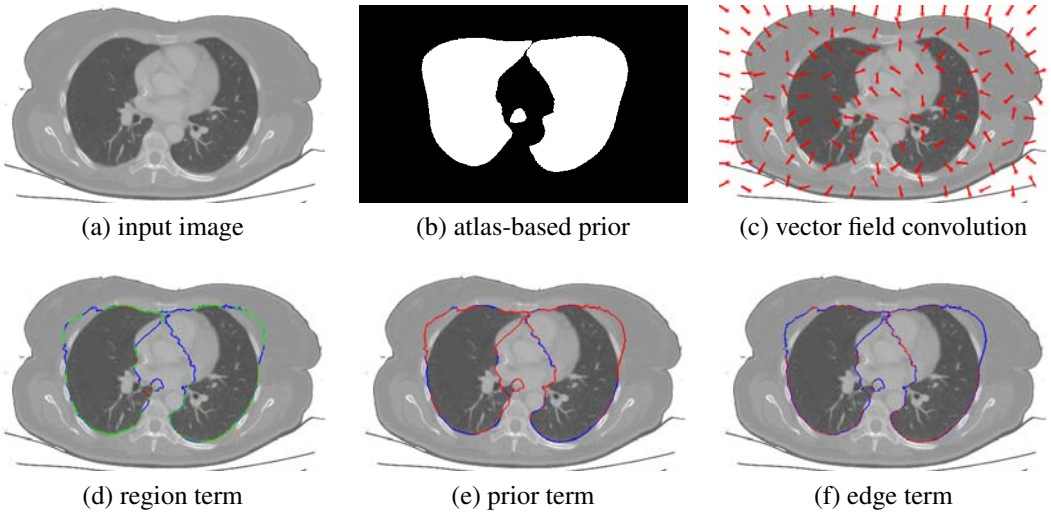
This chapter is structured as follows: in section 7.1, a general overview of the method is presented, providing details about the different terms used in our deformable model. Finally, section 7.2 presents the results and the statistical analysis of the experimental comparison, followed, in section 7.3, by some final remarks and a discussion about possible future developments.

7.1 Proposed Method

In this section, we present a novel segmentation approach based on the LS method, called HybridLS, that combines edge, region and prior shape knowledge of the target object to guide the LS evolution. Moreover, we take advantage of the characteristics of metaheuristics to automatically learn the inherent parameters of a specific type of object using training data (a set of already segmented images).

In its first stage, using an atlas of the target object, HybridLS performs an atlas-based segmentation of the image under consideration, as in section 2.3. This requires the availability of a single image of a similar target object, along with its segmentation. The initial registration-based step provides a prior segmentation that will allow the LS to start its evolution near the area to be segmented. This benefits both the speed and the accuracy of the segmentation since, with a default initialization over the whole image, features located far from the target area are more likely to negatively influence the evolution of the LS.

The LS moves under the influence of three force terms, each providing information



(a) input image (b) atlas-based prior (c) vector field convolution

(d) region term (e) prior term (f) edge term

Figure 7.1: A visualization of the different force terms. In the bottom row, the input image is overlapped with the current contour, which is colored according to the force. Green means the force is close to zero, while blue and red colors mark inward and outward forces, respectively. In this example, the region term (d) is correctly attracting the contour towards the lungs boundaries, as they define two very homogeneous areas. The prior term (e) is just pulling towards the prior segmentation (b). Finally, the edge term (f) is moving the level set towards the closest edges, whether these belongs to the lungs boundaries or not.

about a different characteristic of the current contour. There are a region, an edge and a prior term. The region term minimizes the inhomogeneity of the intensity values inside and outside the surface enclosed by the evolving contour, while the edge term attracts the curve towards natural boundaries and other image edges. Finally, the prior term attracts the LS towards the prior segmentation obtained by the registration, incorporating the information gathered in the first stage of the method in the subsequent segmentation process. Note that this is rather different than just using the prior as initial contour for the LS. Indeed, the prior term, rather than its initial location, influences the evolution of the contour, and can balance the other forces when they are small or inconsistent, leading to a more "conservative" segmentation with respect to using the initial contour. In practice, we have a function combining three components (region, edge and shape information) whose result is a value, not a vector, calculated for each

point in the LS contour, and the sign of such a value gives us the orientation of the movement (towards the inside or the outside of the evolving contour) since the direction is always perpendicular.

Figure 7.2 provides an overview of HybridLS, while Figure 7.1 shows a visual example of the effect of three force terms on the segmentation of a lung CT. The total force acting on the LS is a linear combination of the force terms

$$F_{\text{tot}} = w_{\text{r}} F_{\text{r}}(C) + w_{\text{e}} F_{\text{e}}(C) + w_{\text{p}} F_{\text{p}}(C, P) \tag{7.1}$$

where C is the current contour and P is the prior segmentation. Along with the specific parameters for each term, the use of weights provides flexibility to our approach, allowing it to be adapted to the features and particularities of the objects to be segmented. In HybridLS, a GA is in charge of tuning the weights and the parameters of each term based on training data.

In what follows, we describe the components of HybridLS, starting with the computation of the registration-based prior. Then, we define the three force terms and show how to compute them. Finally, we provide details about the GA and the parameter learning phase.

Registration-based prior

For registration we used a recent algorithm [213] called SS$^+$. The optimization procedure, at the core of the registration process, is based on the Scatter Search metaheuristic (section 2.5), which has been successfully used in a number of works in image registration [214]. In its original study on brain MRI, SS$^+$ provided better results compared to other well-established techniques. Moreover, SS$^+$ delivered the best performance in a preliminary study on the registration of histological images.

In this work, the applications of SS$^+$ are extended to histological and CT images. The registration is performed in two steps, beginning with affine registration. Being a composition of translation, rotation, scaling and shearing operations, an affine transform

can remove large misalignments between the images. Then, a deformable B-Spline-based registration takes care of adjusting the overlap locally, to match the finer details.

To compute a prior, one of the training images plays the role of the atlas. In cases where the target object has a large anatomical variability, a single atlas cannot express the whole distribution of possible shapes the object can assume, leading to poor registration results. This can be improved by using multiple atlases and selecting the most similar atlas for the registration to the target.

In HybridLS, the prior is obtained considering multiple images to be used as atlas. To select the actual atlas, all candidate images are registered to the target image using affine registration. Then, the candidate atlas having the highest similarity metric value is selected for the further B-Spline registration step.

Force terms

Region term

Our region term is borrowed from the classic "Active Contours Without Edges" [45] method by Chan and Vese (see Section 2.2).

In HybridLS, we are interested in a pure region-based term without area or length restrictions, therefore we just use the two homogeneity terms. Therefore, in terms of force acting on the LS, we get

$$F_r(\overline{I}_C, \overline{I}_{\Omega\setminus C}, C) = \begin{cases} \lambda_1 \, |I(x,y) - \overline{I}_C|^2 & (x,y) \in C \\ \lambda_2 \, |I(x,y) - \overline{I}_{\Omega\setminus C}|^2 & (x,y) \notin C \end{cases} \tag{7.2}$$

where Ω is the image domain, C the boundary of an open subset w of Ω, I is the pixel intensity value of the image to be segmented, and $\overline{I}_C$ and $\overline{I}_{\Omega\setminus C}$ are the average values inside and outside C, respectively.

Edge term

The edge term incorporates the information about the boundaries in the image. Basically, the edge term pulls each point of the contour towards the closest edge. Our edge term is based on Vector Field Convolution (VFC) [215]. Compared to other edge-based forces such as Gradient Vector Flow [216], VFC has a lower computational cost and shows greater robustness to noise and initialization. In addition, it showed good performance as external force for DMs [134, 215].

The VFC is static, in the sense that it does not depend on the current LS but only on the target image, therefore the field is calculated only once. The computation of the force occurs in two independent steps. First, an edge map of the target image is obtained applying Gaussian smoothing followed by the Sobel edge detector [217]. Then, the edge map is convolved with a vector field kernel K in which each vector points to the origin, as in Figure 7.3. The magnitude of the vectors decreases with the distance d, in such a way that distant edges produce a smaller force than close edges (the actual value is $1/d^{\gamma+1}$ with $\gamma > 0$). For a point c of contour C, the edge term is simply the normal component of the VFC with respect to C.

Prior term

The aim of the prior term is to move the LS towards the prior segmentation. Also, we want the module of every force vector to be proportional to the overlap between the current evolving curve and the prior segmentation. The idea about how to compute the actual force comes from the region term. If one considers the prior segmentation as a binary image, having an intensity value inside the object and another one outside, this image has two regions that are perfectly homogeneous. This is exactly the kind of result our region-based term was designed to deliver. Therefore, to compute the prior term we simply calculate the region term on the *prior* image, rather than on the target image.

$$F_p(\overline{P}_C, \overline{P}_{\Omega\setminus C}, C) = \begin{cases} |P(x,y) - \overline{P}_C|^2 & (x,y) \in C \\ |P(x,y) - \overline{P}_{\Omega\setminus C}|^2 & (x,y) \notin C \end{cases} \tag{7.3}$$

In this case, λ_1 and λ_2 have been set to 1, since the images used to calculate the prior term force are binary, which means they present perfectly homogeneous foreground and background, and it is not necessary to weigh one region more heavily.

Implementation

In HybridLS, the contour C evolves according to

$$\frac{dC}{dt} = F_{\text{tot}} \cdot \vec{N} \tag{7.4}$$

were F_{tot} is the weighted sum of the three force terms (Equation 7.1) and $\vec{N}$ is the normal direction of C. We used Shi and Karl's Fast-Two-Cycle (FTC) algorithm [218], a fast LS implementation which does not need to solve partial differential equations (PDE). It is a narrow band technique that restricts the calculations of the LS to a much smaller region than the whole grid, and significantly speed up the curve evolution process. This method also separates the evolution process into two different cycles: one cycle for the data-dependent term and a second cycle to regularize smoothness.

Parameter learning using metaheuristics

HybridLS has the ability to learn optimal parameter settings for every specific dataset. Provided a training set of already segmented images of the same class, the parameters are learned using a classic machine learning approach: configurations of parameters are tested on the training data, and the results are compared with the ground truth to assess their quality. In the most basic approach, all combinations of parameters need to be tested, but this exhaustive search is very time consuming, if not even impossible when a large number of parameters are involved. Fortunately, we can overcome this

problem by using metaheuristics, since a properly designed metaheuristic has the ability of learning optimal parameter values faster than an exhaustive search.

In this work, we developed a GA to learn the weights of the force terms (w_r, w_e, w_p) and their corresponding parameters (λ_1, λ_2 for the region term and γ for the edge term). A solution of the problem, or an individual in GA terms, is a string of real values encoding the parameters values. The quality of a solution s (its *fitness*) is defined as the average quality of the segmentations obtained using the parameters values in s. In this case, we measured the average Dice coefficient obtained segmenting the images in the training set.

The GA starts by creating a set of random solutions (a *population*) of fixed size. Then, individuals are selected and variation operators are applied to create a new generation of solutions. The current population is then entirely replaced by a new one except for the best individual, that is never discarded (elitism). The individuals are selected using a *tournament*: k individuals are drawn at random, and the best individual of the group (the winner of the tournament) is selected. The variation operators, responsible for combining and altering solutions, are blend crossover (BLX-α) [219] and random mutation [220]. The random mutation operator picks randomly one of the individual parameters and replaces it with a random value in the parameter's range, both times using uniform probability. The blend crossover operator is more complex: given two individuals x and y, called "parents", for each position i of the parents' coding, the algorithm computes the value $d = |x_i - y_i|$ and then randomly generates two values a, b in the interval $[\min(x_i, y_i) - \alpha d, \max(x_i, y_i) + \alpha d]$ with uniform probability. The values a and b are assigned to the i-th positions of the two offspring, and α is a positive value controlling the width of the ranges in which the new parameters' values are drawn.

It is important to notice that, when testing combinations of parameter values, not all segmentation steps need all parameters. For instance, the VFC of an image depends only on γ. Having this in mind, and in order to speed up the learning process, we saved in a cache all the information that are shared between different configurations. This is especially important for the prior, which is the most computationally demanding step in the segmentation process by far. The prior does not use any of the parameters in

the learning process, therefore only one per image is needed but, since the registration algorithm is non-deterministic, we represented its variability by creating a pool of 30 priors for each image. The priors used in tuning the segmentation parameters were drawn at random from the pool. This approach led to an impressive speedup of the training process. Once the priors and the VFC of each training image have been computed, a single parameter configuration could be tested in less than a second.

7.2 Experimental Setup

One of the main aims of this research is to develop a method that, combining the advantages of geometric DMs, metaheuristics and prior shape knowledge, can achieve remarkable results with different medical image modalities and anatomical structures of interest. To accomplish this purpose, three image modalities with completely different characteristics and different structures have been tested. In this section, these datasets will be described, as well as the anatomical structures to be segmented. Then, we will present the different methods included in the comparison, and devote two separate sections to the atlas registration and the tuning of the parameters (given their critical importance in our pipeline). Finally the final results of segmentation will be presented and analyzed.

Datasets

Three kinds of biomedical image modalities were used to verify the global performance of the different methods over different datasets. We focused our interest on microscopy histological images derived using In Situ Hybridization, X-Ray computed tomography images, and magnetic resonance images.

- In Situ Hybridization-derived images (ISH). 26 microscopy histological images were downloaded from the Allen Brain Atlas (ABA) [184]. The anatomical structure to segment was the hippocampus, and the ground truth was created man-

ually by an expert in molecular biology: every image was manually segmented 5 times and, for each group of 5 manual segmentations, the consensus image was calculated and used as ground truth. The typical resolution of ABA images is about 15,000 $\times$ 7,000 pixels, and the ROIs taken into consideration about 2,500 $\times$ 2,000 pixels.

- Magnetic Resonance Imaging (MRI). A set of 17 T_1-weighted brain MRI were retrieved from a NMR database with their associated manual segmentations [188]. The deep brain structures to segment were: caudate, putamen, globus pallidus, and thalamus. All MR images used in training and test having a resolution of 256 $\times$ 256 pixels.

- X-Ray Computed Tomography (CT). A set of 10 CT images were used in the experiments [134]. Four of them correspond to a human knee and the other six to human lungs. The bone and the lungs are among the darkest objects in the image. Knee images have an average size of 410 $\times$ 435 pixels, while Lung images have a size of 510 $\times$ 350 pixels.

All four datasets, considering lungs and knee as different image sets, were divided in training and test data. The training images were used by HybridLS for the learning of the parameters, while the test images were the ones used in the final experiments to check the performance of the methods.

In ISH, 22 images were used for testing and 4 as a training set. As reference atlas for the registration, the actual references in the ABA were employed to obtain the shape prior. With respect to MRI, 3 images were used as training set, 3 other were used as atlas, and the remaining 11 as test set. Finally, in relation to CT, one image of every organ was used as training and as atlas for the registration, leaving 3 lung and 2 knee images for testing the system.

Methods included in the comparison

In our comparisons we have included both deterministic and non-deterministic methods, as well as classic and very recent proposals. The stress has been focused on DMs and their hybridization with metaheuristics, but other kinds of approaches have also been taken into account.

- Active Shape Models (and Iterative Otsu Thresholding Method) refined using Random Forests (ASM + RF) [123, 152]. This method, explained in chapter 5 and published in 2012, uses a medial-based shape representation in polar coordinates, with the objective of creating simple models that can be managed in an easy and fast manner. Such a parametric model is moved and deformed by a metaheuristic (Differential Evolution (DE) [99]) according to an intensity-based similarity function between the model and the object itself. After that, Otsu's thresholding method [15] is iteratively applied on every region identified by the located control points. Finally, Random Forests [121] are applied to expand the segmented area to the regions that were not properly localized. This segmentation algorithm has shown very good performance in histological images, but needs a training set of shapes to manually create the parametric template and its possible deformations, as well as a training set of textural patterns for the expansion phase. Due to these restrictions it was only applied to ISH images.

- Soft Thresholding (ST) [16]. This deterministic method, presented in 2010, is based on relating each pixel in the image to the different regions via a membership function, rather than through hard decisions, and such a membership function is derived from the image histogram. In a first stage, the normalized histogram of the image is calculated and a sum of weighted known distributions is fit to it. Each probability distribution represents the probability for a pixel with a certain value to belong to the corresponding region. This segmentation technique is totally automatic, and the spatial operations make thresholding more robust to noise and artifacts. Having been successfully applied to CT, MRI and ultrasound images, it seemed interesting to apply it also to microscopy histological images

and compare its performance with other state-of-the-art methods.

- Atlas-based deformable segmentation (DS) [213]. This method refers to the atlas-based segmentation procedure used in HybridLS to compute the prior (section 7.1). This is actually a stand-alone segmentation method, therefore it is included in the experimental study as a representative of registration-based segmentation algorithms. Moreover, comparing DS's and HybridLS's results will assess the influence of the prior term on the performance of the second method. During the whole study, the setup and the atlas selection mechanism of DS (section 7.1) are always the same both when the method is used as a stand-alone tool and when it is embedded in another segmentation technique.

- Geodesic Active Contours (GAC) [47]. This technique, introduced in 1997, connects 'snakes' based on energy minimization and geometric active contours based on the theory of curve evolution (see section 2.2). It is based on active contours that evolve in time according to intrinsic geometric measures of the image: the evolving contours naturally split and merge, allowing the simultaneous detection of several objects and both interior and exterior boundaries.

  In this chapter, two implementations of GAC have been tested. The first one uses as initial contour the whole image, while the second one, called DSGAC, employs the segmentation obtained using DS to create the initial contour of the geometric DM.

- Chan&Vese Level Set Model (CV) [45]. This implicit DM, presented in 2001, was also included in the comparison to check its performance in comparison with the other approaches (see section 2.2). Also in this case, like in GAC, two implementations have been tested. The first one uses the whole image as initial contour, and the second one employs the segmentation results obtained by DS as the LS initial contour.

Parameter settings

As HybridLS has an automatic parameter learning phase, it would be unfair to compare it against other methods that do not include some kind of parameter tuning. Manual parameter tuning is time consuming and error-prone, while using the GA to tune all methods could introduce a bias, as the behavior of the GA could vary with each method. In general, we want the competitors to deliver their best performance, regardless of their parameter sensitivity or their ability to be tuned. Therefore, we decided to tune the competitors with an exhaustive search using the *test data*, rather than the training set. This means that the results reported for all methods but HybridLS are actually the best average results they can obtain on these datasets. This gives them a clear advantage over HybridLS, as for the latter the parameters are learned using only the training data.

For CV, GAC, DSGAC and DSCV, all the possible combinations of the values in Table 7.1 were tested. Also, a pre-processing and a post-processing stages were included to improve the results obtained. The post-processing stage refines the results removing the connected components that are smaller than a certain number of pixels, while pre-processing is performed using a median filter to remove the salt-and-pepper-like noise present in some of the images. Moreover, for DSGAC and DSCV, 10 different initial masks were created using DS and the best one was used in the tuning. The number of iterations for GAC and CV was set to 500 to ensure the process reached convergence. In a few cases, on the ISH dataset, CV failed to converge within the limit due to poor parameter values. This occurred only while producing very low quality, degenerate segmentations, therefore the early stopping did not affect the tuning process.

After tuning these methods, the minimum allowed size in pixels of the connected components was set to 75, 200 and 25000 for MRI, CT and ISH, respectively. For ASM+RF, the parameters used (Table 7.3) were those suggested in our previous experiments with the ISH dataset (see section 5).

For HybridLS, the parameters settings were learned by the GA using the training data. The size of the population was set to 50 individuals, and the evolution lasted 50 generations. The probability of crossover and mutation was set to 0.7 and 0.1,

respectively, and the size of the tournament was 3. The range of λ_1, λ_2 was restricted to $\{1, 2, 5\}$ to match the settings used with the other methods.

The final parameters configurations are reported in Table 7.2. It is interesting to remark how the GA detected a different level of importance for each term in different datasets. For instance, in MRI the edge term is not used ($w_e = 0$) since our machine learning system determines that, for a good segmentation, the region term and prior shape knowledge are enough. When segmenting CT-imaged lungs the only term used is the region-based one. In this case, λ_1 and λ_2 were set to 5 and 2, respectively. This means that our final segmentation will have a more uniform foreground region (since the energy contributed by the "variance" in the foreground region has a larger weight), at the expense of allowing more variation in the background.

Table 7.1: Combination of parameters tested for CV, GAC, DSCV and DSGAC.

Parameter	Values
α contour_weight	{1, 2, 3}
β expansion_weight	{-1, -0.5}
μ weightLengthTerm	{0.01, 0.1, 0.25, 0.5, 0.75}
λ_1	{ 1, 2, 5}
λ_2	{ 1, 2, 5}
size median filter	{1, 3, 5, 10}
minimum size allowed	{1, 50, 75, 100, 200, 5000, 25000}

Experimental results

To evaluate the performance of the segmentation methods, we employed three standard segmentation metrics: the Dice coefficient (DSC), the Jaccard similarity index (JI) and the Hausdorff distance (HD). See Appendix II for more information in this regard.

It is important to remark that ASM+RF, DS, DSCV, DSGAC and HybridLS are non-deterministic, since stochastic methods, like Differential Evolution or Scatter Search, are embedded in these algorithms. It is essential to execute such algorithms several times to estimate and compare their performances. In this case, 20 repetitions per image were run and the mean, median and standard deviation values were calculated over the whole set of results (see Table 7.5). For instance, in ISH the mean Dice value of DS

Table 7.2: Parameters obtained after tuning ST, GAC, CV, DS+GAC, DS+CV, and training HybridLS.

CV	GAC	CV+DS	GAC+DS	HybridLS
		Magnetic Resonance Imaging		
500 iterations ν = 0 μ = 0.01 $\lambda_1 = \lambda_2 = 1$ medFiltSize = 1	500 iterations β = -1 α = 3 medFiltSize = 3	500 iterations ν = 0 μ = 0.01 λ_1 = 1 λ_2 =1 medFiltSize = 5	500 iterations β = -0.5 α = 3 medFiltSize = 1	λ_1 = 5 λ_2 = 1 w_r = 5.1 w_p = 1.1 w_e = 0 γ = 1.5
		Computerized Tomography - Knee		
500 iterations ν = 0 μ = 0.01 λ_1 = 5 λ_2 = 2 medFiltSize = 3	500 iterations α = 1 β = -0.5 medFiltSize = 1	500 iterations ν = 0 μ = 0.01 λ_1 = 1 λ_2 = 1 medFiltSize = 1	500 iterations α = 3 β = -0.5 medFiltSize = 1	λ_1 = 2 λ_2 = 5 w_r = 4.8 w_p = 0.9 w_e = 2 γ = 1.5
		Computerized Tomography - Lungs		
500 iterations ν = 0 μ = 0.01 λ_1 = 5 λ_2 = 2 medFiltSize = 3	500 iterations β = -1 α = 2 medFiltSize = 3	500 iterations ν = 0 μ = 0.01 λ_1 =1 λ_2 = 5 medFiltSize = 3	500 iterations β = -1 α = 3 medFiltSize = 3	λ_1 = 5 λ_2 = 2 w_r = 1.5 w_p = 0 w_e = 0 γ = 1.5
		In Situ Hybridization-derived images		
500 iterations ν = 0 μ = 0.01 $\lambda_1 = \lambda_2 = 1$ medFiltSize = 5	500 iterations β = -1 α = 3 medFiltSize = 10	500 iterations ν = 0 μ = 0.01 λ_1 = 1 λ_2 = 1 medFiltSize = 5	500 iterations β = -1 α = 3 medFiltSize = 10	λ_1 = 1 λ_2 = 1 w_r = 1.9 w_p = 2.2 w_e = 1 γ = 2

Table 7.3: Parameters used in ST, DS and ASM+RF. All parameters were taken from the original proposals.

ST	ASM+RF	DS
L = 2 regions Relative max normalization	$Cr = 0.9$ $F = 0.7$ Uniform Crossover DE/target-to-best/1 Population Size = 80 Iterations = 250 Median Filter [25×25] RF with 500 trees	Metric = AdvancedNormalizedCorrelation Optimizer = ScatterSearch SS_b = 12 PSize = 32 BLX-α = 0.3 LS-iterations = 25 NumberOfIterations = 15 NumberOfResolutions = 3 NumberOfSpatialSamples = 2000 5000 10000 Restarts = 3

is 0.876, and represents the average of 440 experiments performed (20 repetitions per image and 22 images).

We also performed a statistical analysis of the results. When comparing two methods, we used Wilcoxon rank-sum test [221], a non-parametric statistical test that checks whether one of two independent samples tends to have larger values than the other. When multiple comparisons were performed, Holm correction [222] was applied to the p-values to control the family-wise error rate. Note that, in the Lungs and Knee datasets, the number of images is not large enough for comparing the deterministic methods (ST, CV and GAC), therefore these methods have been excluded from the test.

In Table 7.4, some concise information about the running time of each algorithm is provided with an illustrative purpose. The fastest method is ST with MRI, which takes only 1 second per image, while the slowest are the applications of DS to ISH, employing up to 10 minutes to process an image. Nevertheless, several factors affect the accuracy of a comparison in terms of execution time. First, some of the methods have been developed in MATLAB and others in C++. Moreover, the size of the images differs from one image modality to another, as well as some of the pre- and post-processing stages we used. Finally, the nature of the algorithms is completely different, making them hard to compare[1].

Table 7.4: Average execution time per method and kind of image. All values are in seconds, and were obtained running the experiments in an Intel Core i5-2410M @ 2.3GHz with 4.00 GB of RAM. Also the programming environment has been included between brackets.

	ASM+RF (MATLAB, C++)	ST (MATLAB)	CV (MATLAB)	GAC (MATLAB, C++)	DSCV (MATLAB, C++)	DSGAC (MATLAB, C++)	DS (C++)	HybridLS (C++)
ISH	35	39	87	32	582	493	471	545
Lungs	-	1.7	36	15	384	342	326	331
Knee	-	2.5	67	16	310	265	245	252
MRI	-	1	11	1.5	429	407	404	405

[1]For instance, ST is only based on the image histogram, while others have a registration step that imposes a notable computational overhead.

Analysis

The experimental results are reported in Table 7.5. Visual examples of two segmentations obtained by the methods on each dataset are provided in Figure 7.5. For simplicity, our discussion focuses on the results in terms of mean DSC, but note that this choice does not really affect the outcome of the comparison, as there is an almost perfect agreement with the other validation measures we considered.

The performance of the segmentation methods varies greatly across the four datasets. The easiest problem to be solved has been the segmentation of Lungs in CT images, with all methods but GAC and DS scoring higher than 0.95. The most complex task has shown to be the segmentation of deep anatomical structures in brain MRI, where four of the compared methods have obtained an average DSC of 0.2 or less).

The per-dataset results are shown in Figure 7.4 using boxplots and in Table 7.6 through the average rankings. Obviously, the performance of every method depends on the nature of the image to be segmented. For instance, techniques based on the intensity level (such as CV and ST) yielded worse results in image modalities, like MRI, having less contrast and small differences in terms of pixel intensity.

HybridLS has obtained the best results in all biomedical image datasets. It achieved the best values for the mean DSC and JI metrics, and it was ranked as the best method in every image modality. The Wilcoxon test (Table 7.6) showed, with really high confidence, that the difference between HybridLS and the other methods is statistically significant in all but one case (DS on MRI). This behavior is also robust, as shown by the low standard deviation values. We can then conclude that our proposal is the best segmentation method in the comparison.

The DS method has been one of the best-performing algorithms, ranking second or third over three datasets. More in general, all methods using the registration-based initialization scored better than those using a standard one. This applies also to CV and GAC: in all but one case, both DSCV and DSGAC ranked better than their counterpart, with a statistically significant difference (Table 7.7).

Overall, DSGAC delivered an acceptable performance, ranking above average in

three datasets out of four. This is remarkable, as the regular GAC ranked constantly in the last three positions, and it can be explained by the high sensitivity of GAC to its initialization.

DSCV reached an average ranking in all datasets, performing slightly worse, although more consistently, than DSGAC. The plain CV method achieved a bad performance, ranking last or second to last in three datasets. Only on the Lungs dataset, where the gray value is enough to segment the target quite accurately, CV delivered good results.

ST results showed a similar pattern to CV. It performed better than CV, but since ST is based on the image histogram it showed limited ability to cope with complex scenarios. On the other hand, ST is the fastest method among the ones taken into consideration and it has virtually no parameters to be set.

ASM+RF obtained some of the best results with ISH images, being also one of the fastest techniques, but it is fair to underline its ad-hoc nature. It needs a training set of shapes to create the template and its possible deformations, and it also needs a training set of textural patterns for the expansion phase. Also, it is not able to manage topological changes in a natural way, as geometric DMs can do.

Finally, regarding the values of HD, it is interesting to notice how methods without shape deformation restrictions, like ST, CV and GAC, have a higher (worse) HD with respect to others introducing prior shape knowledge, like ASM+RF, DS and HybridLS.

7.3 Conclusions

It is crucial to highlight the main features of HybridLS:

- it is an accurate and also general segmentation method (it obtains very good results with all the medical image modalities tested, even overcoming well-consolidated techniques);
- its overall standard deviation is the lowest among the different methods we

Table 7.5: Segmentation Results using 3 different metrics: Dice Similarity Coefficient (DSC), Jaccard Index (JI), and Hausdorff Distance (HD). Values are sorted in descending order using their average DSC value as criterion. The best results for every metric are shown in bold.

Dataset	Method	Dice Coefficient			Jaccard Index			Hausdorff Distance		
		mean	median	stdev	mean	median	stdev	mean	median	stdev
ISH	HybridLS	**0.888**	**0.918**	0.079	**0.806**	**0.849**	0.109	103.614	**70.109**	100.115
	ASM+RF	0.885	0.896	**0.040**	0.797	0.812	**0.061**	114.906	94.736	**52.785**
	DS	0.876	0.907	0.078	0.787	0.829	0.108	**101.437**	75.166	79.779
	DSGAC	0.791	0.830	0.143	0.674	0.709	0.172	215.237	150.727	196.908
	ST	0.728	0.775	0.175	0.597	0.632	0.192	578.876	665.301	303.314
	DSCV	0.673	0.764	0.203	0.538	0.618	0.203	263.345	176.706	234.192
	GAC	0.670	0.722	0.181	0.528	0.564	0.192	693.257	707.560	265.589
	CV	0.589	0.723	0.257	0.460	0.567	0.242	839.800	844.773	331.804
Knee	HybridLS	**0.868**	**0.872**	0.087	**0.777**	**0.782**	0.136	**13.690**	**13.618**	**1.735**
	DSGAC	0.810	0.811	0.142	0.705	0.705	0.204	84.611	84.611	4.040
	DS	0.687	0.685	0.227	0.569	0.563	0.271	45.015	45.308	25.303
	DSCV	0.528	0.527	0.079	0.363	0.361	0.073	69.642	76.261	16.468
	GAC	0.486	0.486	0.310	0.349	0.349	0.276	187.087	187.087	29.528
	ST	0.398	0.398	0.088	0.250	0.250	0.069	144.864	144.864	36.792
	CV	0.230	0.230	**0.072**	0.131	0.131	**0.046**	179.680	179.680	30.499
Lungs	HybridLS	**0.996**	**0.997**	**0.001**	**0.992**	**0.994**	**0.003**	**2.949**	**3.606**	**1.415**
	ST	0.979	0.990	0.023	0.960	0.981	0.044	51.550	60.208	42.882
	CV	0.973	0.992	0.034	0.949	0.983	0.063	52.903	60.208	38.592
	DSCV	0.966	0.985	0.034	0.935	0.971	0.062	26.512	25.495	9.321
	DSGAC	0.950	0.952	0.027	0.906	0.908	0.049	37.828	29.155	27.197
	DS	0.896	0.897	0.062	0.818	0.814	0.102	39.378	51.478	18.023
	GAC	0.670	0.627	0.251	0.542	0.457	0.309	143.938	137.568	34.430
MRI	HybridLS	**0.758**	0.780	0.048	**0.612**	0.639	0.062	**7.819**	**6.708**	2.874
	DS	0.752	**0.783**	0.056	0.606	**0.643**	0.071	10.838	10.000	3.192
	DSGAC	0.585	0.613	0.087	0.418	0.442	0.084	25.113	25.495	**2.414**
	DSCV	0.204	0.213	0.054	0.115	0.119	0.033	76.792	76.322	5.955
	CV	0.155	0.171	**0.042**	0.084	0.093	0.024	93.727	93.408	4.670
	ST	0.175	0.181	0.053	0.097	0.100	0.032	84.936	85.586	6.289
	GAC	0.124	0.139	0.035	0.066	0.074	**0.020**	90.856	90.139	5.459

Table 7.6: Average rank achieved per image modality and adjusted p-value of Wilcoxon test comparing each algorithm against HybridLS.

Dataset	Method	Mean Rank	p-value
ISH	HybridLS	1.82	
	DS	2.50	0.000
	ASM+RF	2.64	0.000
	DSGAC	4.14	0.000
	ST	5.68	0.000
	DSCV	6.23	0.000
	GAC	6.36	0.000
	CV	6.64	0.000
Knee	HybridLS	1.50	
	DSGAC	2.00	0.000
	DS	3.00	0.000
	DSCV	4.50	0.000
	GAC	5.00	-
	ST	5.50	-
	CV	6.50	-
Lungs	HybridLS	1.00	
	ST	2.33	-
	CV	3.00	-
	DSCV	3.67	0.000
	DSGAC	5.33	0.000
	DS	5.67	0.000
	GAC	7.00	-
MRI	HybridLS	1.27	
	DS	1.73	0.46
	DSGAC	3.00	0.000
	DSCV	4.18	0.000
	ST	5.09	0.000
	CV	5.73	0.000
	GAC	7.00	0.000

Table 7.7: Pairwise comparison of all the methods but HybridLS. Each cell of the table reports the p-value of Wilcoxon test comparing the method on the corresponding row with the method associated with the column.

ISH

	ASM+RF	CV	DS	DSCV	DSGAC	GAC
CV	0.00					
DS	0.01	0.00				
DSCV	0.00	0.38	0.00			
DSGAC	0.00	0.00	0.00	0.00		
GAC	0.00	0.92	0.00	0.92	0.00	
ST	0.00	0.32	0.00	0.61	0.26	0.61

Knee

	DS	DSCV
DSCV	0.00	
DSGAC	0.00	0.00

Lungs

	DS	DSCV
DSCV	0.00	
DSGAC	0.00	0.00

MRI

	CV	DS	DSCV	DSGAC	GAC
DS	0.00				
DSCV	0.02	0.00			
DSGAC	0.00	0.00	0.00		
GAC	0.10	0.00	0.00	0.00	
ST	0.79	0.00	0.26	0.00	0.08

compared, therefore we can affirm that the developed approach is consistent and stable in terms of performance;

- it does not need a training set of textures or shapes to segment the object of interest (it needs only one reference image to obtain the shape prior);
- it performs self-adaptation of its own parameters depending on the medical image modality to segment;
- it uses metaheuristics to generate the shape prior and to perform the previously mentioned learning of parameters.

Thanks to the automatic learning of the model parameters, our hybrid proposal is able to perform an effective segmentation with very different medical image modalities, adapting the importance of every term to each image modality and anatomical structure.

The main drawback of HybridLS is that it is not as fast as ST or even ASM+RF. This is obvious since it can be as fast as its components and, obviously, DS is a deformable registration process that can take several minutes on a general purpose computer. More sophisticated implementations, like GPGPU programming, can be tested to speed-up the computations. Finally, the introduction of a textural term could be taken into consideration if the benefits obtained with its use justify it.

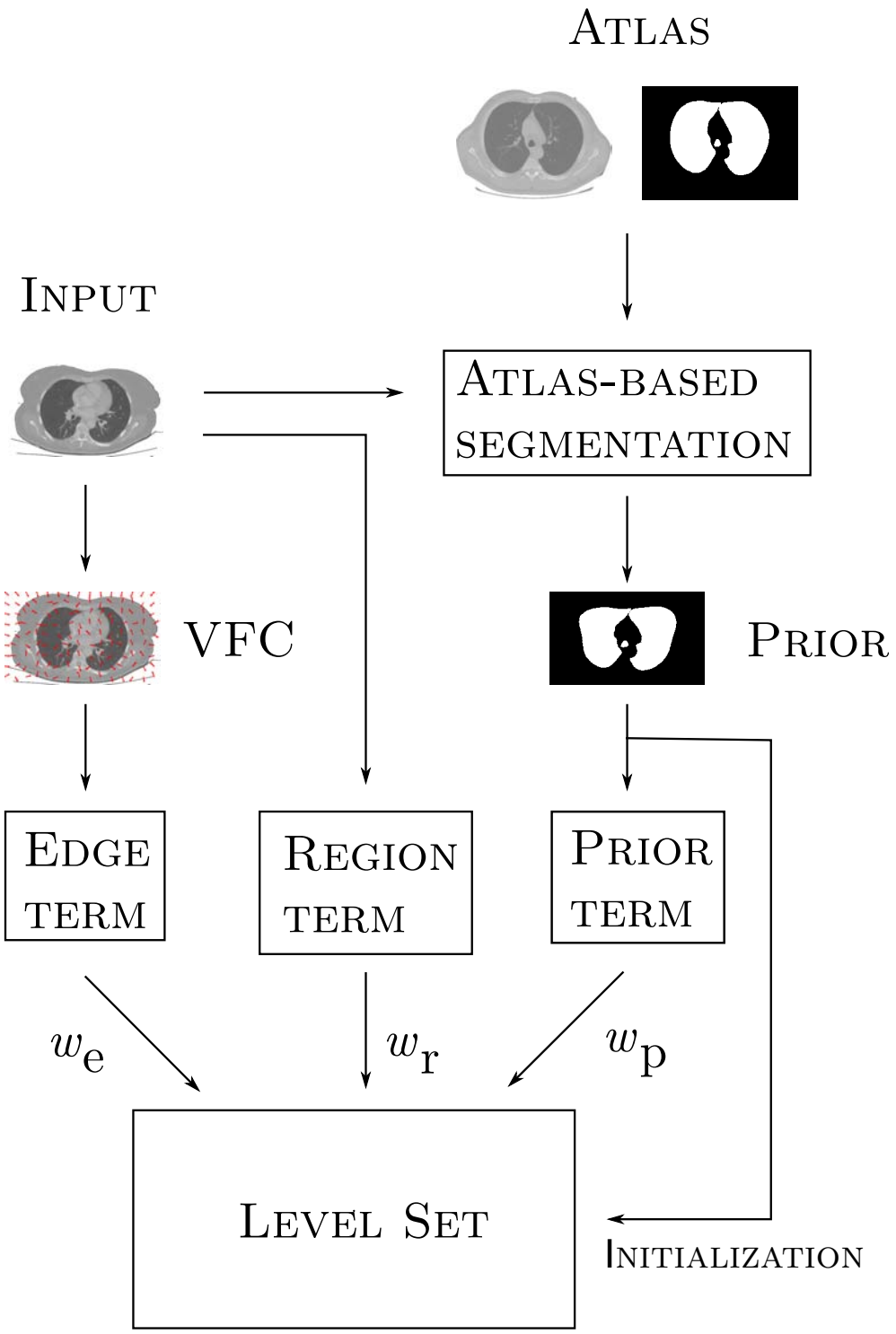


Figure 7.2: The schematic view of the interaction among the components of HybridLS.

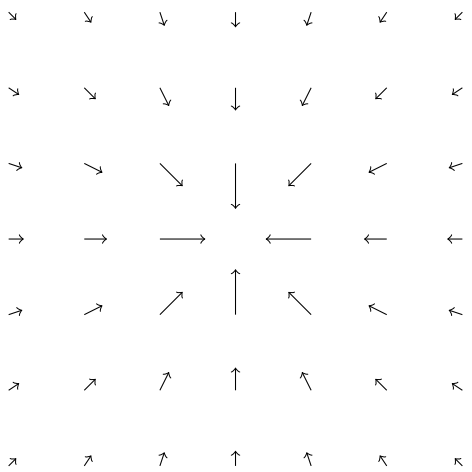


Figure 7.3: Vector field kernel used to compute the VFC term.

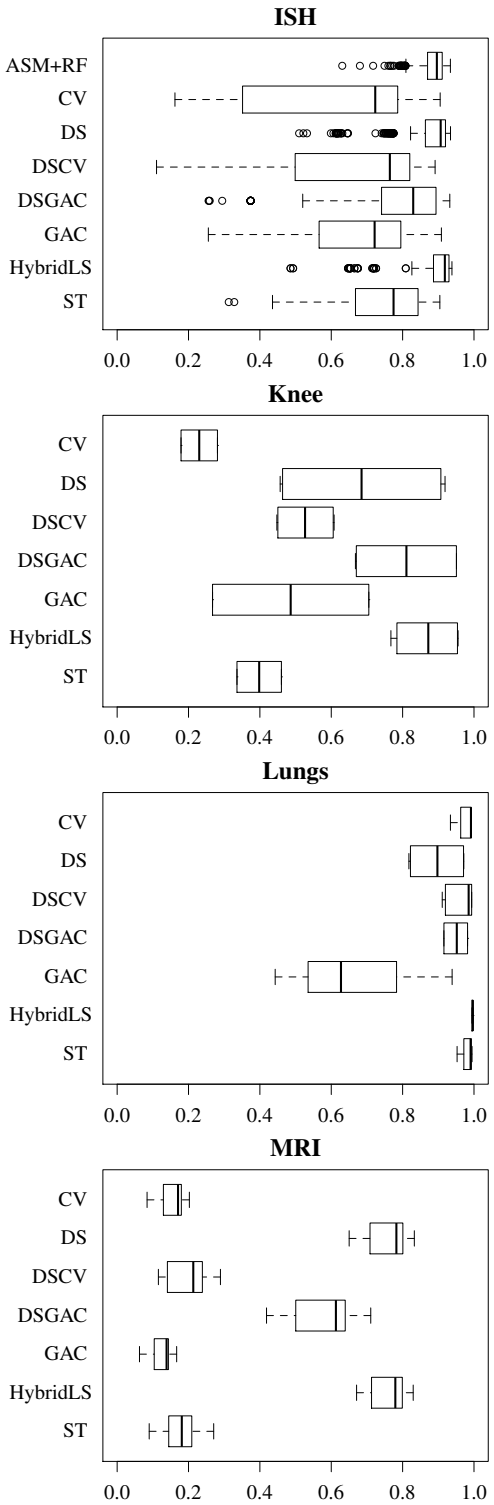


Figure 7.4: Box-plot representing the DSC for all methods.

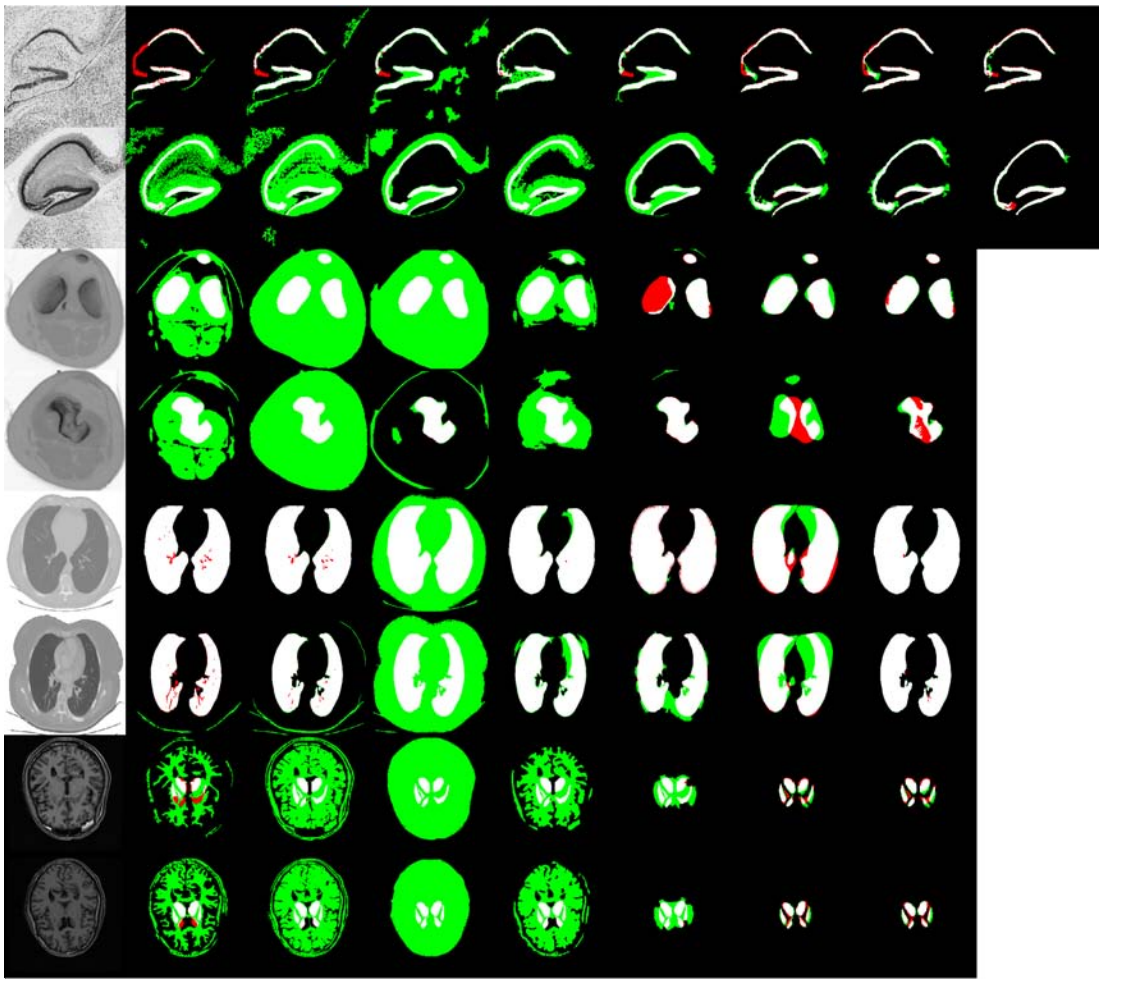


Figure 7.5: Some visual examples of the results obtained. Two images per image modality and per structure to segment have been selected: the first two rows correspond to ISH, the next two rows to CT-Knee, and the last four to CT-Lungs and MRI. White represents true positives, red false negatives, and green false positives.

Part III: Final Remarks

The last part of this PhD dissertation is focused on the summary and conclusions of the activity performed, and on the introduction of potential future research work which may extend the results described in this dissertation.

Chapter 8

Further Work

There are several ways in which the research work described in this dissertation can be further developed. The first of these could be a 3D extension of the 2D segmentation methods that have been developed so far, in order to be also used to segment volumes. Another interesting future development could be the optimization of computation efficiency of the algorithms by means of GPGPU (General Purpose Graphics Processing Unit) programming. In particular, CUDA™ (Compute Unified Distributed Architecture) is a parallel computing environment by nVIDIA™ which exploits the massively parallel computation capabilities of its GPUs (containing up to several hundreds of execution cores that can execute the same code on different data). A possible further study could be focused on the parallelization possibilities and consequent increase in execution speed that these techniques can offer.

In medical IS, sometimes it is possible to define multiple criteria that need to be optimized simultaneously. Hence, another major issue in this regard is the application of multiobjective optimization (MOO) techniques [223, 224] that can be effectively utilized to yield a set of Pareto optimal solutions that the domain expert can then analyze. Moreover, hybridizations of MOO techniques with other computational intelligence techniques, like fuzzy sets or neural networks, are interesting directions of future research in this field. In fact, all challenging and realistic problems can be considered fuzzy and multiobjective, in the sense that they present different conflicting objectives,

for which it is necessary to find a trade-off, and, on the other hand, they present some degree of uncertainty that could be processed using fuzzy logic/systems.

Chapter 9

Summary and Conclusions

¿Ves aquel señor graduado,
roja borla, blanco guante,
que némine discrepante
fue en Salamanca aprobado?
Pues con su borla, su grado,
cátedra, renta y dinero,
es un grande majadero.
José Iglesias de la Casa

With all things being equal, the simplest explanation tends to be the right one.
The explanation requiring the fewest assumptions is most likely to be correct.
William of Ockham - Ockham's razor (in Latin lex parsimoniae)

Imagination is more important than knowledge.
Knowledge is limited. Imagination encircles the world.
Albert Einstein, quoted in interview by G.S. Viereck, October 26,1929.
Reprinted in "Glimpses of the Great"(1930)

This research has been focused on studying how hybridizations between soft/bio-

inspired computing and computer vision techniques could help in biomedical image processing tasks. Within the complete pipeline of medical imaging (acquisition, restoration, registration, reconstruction, tracking,...), the topic selected has been the automatic segmentation of anatomical structures. Image segmentation is the partition of an image into meaningful and non-overlapping regions following some criterion, and it represents an essential step in many important biomedical activities (detection of lessions, quantitative tissue analysis, computer-integrated surgery,...).

Three different methods have been presented, analyzing their corresponding advantages and disadvantages, in which soft computing approaches (metaheuristics and ensemble classifiers) are combined with deformable models (both parametric and geometric) to segment anatomical structures. Chapter 5 describes a parametric approach, based on active shape models, metaheuristics and ensemble classifiers, for segmenting the hippocampus in histological mouse brain images, comparing different optimization techniques and segmentation methods on this task. It also introduces a real-world application in which this parametric segmentation algorithm was successfully applied to the identification of RNAs in gene-expression images. In chapter 6, a geometric method using eigenshapes, which combines texture and intensity information, is used to segment the same anatomical structure. Finally, in chapter 7, a level set approach including three different terms (intensity, boundaries and prior shape knowledge) is applied to segment knee and lungs in CT images, the hippocampus in histological mouse brain images, and caudate, putamen, globus pallidus, and thalamus in T_1-weighted brain MRI.

Throughout this work, metaheuristics have been used to tune an algorithm (i.e., to learn its working parameters), to perform feature selection, and to optimize a target function that determines the proper segmentation (or registration) of the structure under consideration. Finally, an ensemble classifier has been used to refine the segmentation performed in histological brain images.

There are some lessons learned that can be used to provide recommendations for future researchers in this field:

- There are few publications (around ten in total) about automatic tuning of the deformable model parameters. Metaheuristics, or automatic tuners like *irace*, should be used to automatically configure computer vision algorithms in order to overcome the problems related to manual tuning (tedious, error-prone, time-consuming).

- There are also few approaches dealing with local deformations. Usually the deformable model is adapted globally using metaheuristics, when maybe the subdivision of the DM in sections to be optimized locally could be an interesting approach to take into account in the future.

- The use of the level set method and prior shape knowledge using metaheuristics can be one of the most interesting future trends, due to its potentiality and to the current lack of publications on the topic. In fact, a very good starting point when segmenting anatomical structures is to include as much prior information as possible (shape, texture, intensity, boundaries), since this can speed the execution time and increase the accuracy of the results obtained. In this PhD thesis, sections 6 and 7 provide an example of how this can be accomplished.

- It is important to consider the trade-off between generality and accuracy. This doctoral thesis has confirmed that a general method tends to be less accurate than an ad-hoc approach to a certain problem. In fact, the generalization ability of a method can be increased by automatically tuning its parameters, since such a tuner can perform the parameter configuration to effectively work with a particular type of structure to segment and/or a particular medical imaging modality.

- The application of standard metrics and statistical tests (absent in many scientific reports) should be considered from now on as an obligation for a rigurous investigation. Also, the size of the test sets should be increased to allow a better interpretation of the results, avoiding methods and scenarios that lead to very specific solutions. In direct relation to this aspect, the datasets used should be made public and shared by the scientific community, in order to facilitate the comparison of results between researchers, as well as the development and testing

of new algorithms thanks to the existence of ground truth for every dataset.

- The use of novel metaheuristics like Differential Evolution, Particle Swarm Optimization or Scatter Search is strongly recommended, since Simulated Annealing and similar approaches are obsolete and unable to efficiently solve real-world problems. Obviously, the usually multimodal, non-convex and high-dimensional nature of segmentation problems makes it useless to apply gradient-based/classical approaches (Gauss-Newton, Levenberg-Marquardt,...). A possible alternative would be finding a suitable transformation (convexization) of the objective function in order to use such classical approaches, but the use of metaheuristics appears to be faster and even easier.

- With reference to the previous point, when using an evolutionary algorithm, it is fundamental to apply operators which fit the nature of the problem and the characteristics of the chromosomes under consideration. For instance, if one is using real-coded genomes, the best option is to use SBX, PBX or BLX-α to perform recombination. Also, since fitness computation is the most time-consuming component of an evolutionary algorithm, its effective design is crucial for a successful application of metaheuristics.

- Finally, deformable models still seem to be the most natural manner of including prior shape knowledge into a computer vision technique, and are extensively used in the medical field, to name just a single example. So, the use of these methods is a safe bet and a good starting point when designing a new medical image segmentation algorithm.

Appendix I: Statistical tests for analyzing Soft Computing techniques behaviour

He uses statistics as a drunken man uses lamp-posts...
for support rather than illumination.
Andrew Lang

The use of statistical tests is the only way to rigorously compare the performance of computational intelligence methods [225–227]. Usually, they are employed based on the experimental results they have obtained to decide if one algorithm should be considered better than another. This task is compulsory to confirm whether a newly proposed method offers a significant improvement, or not, over the existing methods for a given problem.

Statistical tests measure the probability of obtaining the experimental results D if the hypothesis H is correct, computing $P(D|H)$, and can be categorized into two classes: parametric and non-parametric procedures. Parametric tests have been commonly used in the analysis of experiments in computational intelligence. Unfortunately, they are usually based on assumptions (independence, normality, and homoscedasticity) which are most probably violated when analyzing the performance of stochastic algorithms [228].

To introduce these methods, it is necessary to define essential concepts in statistical

inference as null hypothesis, p-value or type I/II Errors. The **null hypothesis** H_0 refers to a general or default statement of a scientific experiment. It is presumed to be true until statistical evidence nullifies it for an alternative hypothesis, and it assumes that any kind of difference or significance one sees in a set of data is due to chance. For example, in a clinical trial of a new drug, the null hypothesis might be that the new drug is no better, on average, than the current drug. A **type I error** is the incorrect rejection of a true null hypothesis and represents a false positive, while a **type II error** is the failure to reject a false null hypothesis, thus representing a false negative (see Table 9.1).

Another fundamental concept to be defined is the **p-value**, i.e., the estimated probability of rejecting H_0 when it is true. In other words, the p-value is the probability of obtaining a test statistic, at least, as extreme as the one that was actually observed, assuming that the null hypothesis is true. If the p-value is less than the chosen level of significance α then the null hypothesis is rejected, thus α tells us how extreme observed results must be in order to reject the null hypothesis (giving us the probability of a type I error)[1]. Finally, the **statistical power** of a test is the probability of rejecting H_0 when it is false (i.e., the probability of not committing a type II error). As the power increases, the chances of a type II error occurrence decrease. Note that the statistical power determines the probability of finding a difference that does exist, as opposed to the likelihood of detecting a difference that does not exist (which is known as a type I error, or "false positive").

Table 9.1: Type I and type II errors.

	Null hypothesis (H_0) is true	Null hypothesis (H_0) is false
Reject null hypothesis	Type I error / False positive	True positive
Fail to reject null hypothesis	True negative	Type II error / False negative

Furthermore, in order to perfectly understand the following explanations, it is fundamental to know that an independent variable (IV), sometimes called an experimental or predictor variable, is a variable that is being manipulated in an experiment in order to observe the effect on a dependent variable (DV), sometimes called outcome variable. So,

[1]For instance, if the p-value is less than the predetermined significance level (usually 0.05 or 0.01), it indicates that the observed result would be highly unlikely under H_0 or, in other words, the observed result is highly unlikely to be the result of random chance).

if the IVs change, the DV will change. In other words, in an experiment, the factor (also called IV) is an explanatory variable manipulated by the experimenter. Each factor has two or more levels, i.e., different values of the factor. Combinations of factor levels are called treatments. For instance, a doctor can treat a patient with two drugs (factors) to see which is more effective, and every drug can be provided in three different amounts (5mg, 10 mg, 15 mg). Therefore, for this particular example, the experiment has six treatments.

The next sections explain some of the tests useful for the analysis of the algorithms behaviour in the experimental parts of this PhD dissertation (sections 5.2, 5.3, 6.3 and 7.2).

Student's t-test

A t-test is a parametric statistical test that checks the null hypothesis that the means of two normally distributed populations are equal, assuming the variances of the two populations to be equal. Welch's t test is an adaptation of Student's t-test intended for use with two samples having possibly unequal variances.

In computational intelligence, a common way to test whether the difference between two classifiers' results over various data sets is non-random has been to compute a paired t-test, which checks whether the average difference in their performance over the data sets is significantly different from zero. Nevertheless, this approach suffers from three weaknesses [225]: commensurability (it only makes sense when the differences over the data sets are commensurate), sample size (unless the sample size is large enough, the paired t-test requires that the differences between the two random variables being compared are distributed normally[2]), and outliers (it is affected by outliers which skew the test statistics and decrease the test's power by increasing the estimated standard error).

[2]Also, usually, the statistical tests to check the normality of a distribution have little power on small samples, that is, they are unlikely to detect abnormalities and warn against using the t-test.

Mann-Whitney U test

The Mann-Whitney U test (also called Wilcoxon rank-sum test) [221] is a non-parametric test for checking the equality of population medians of two independent samples. Although a Mann-Whitney's U test can be considered as a non-parametric version of the t-test, it compares the medians of the two groups, not the means, and it treats the data as ordinal data, calculating the rank for each value instead of using the values themselves.

Wilcoxon signed-rank test

The Wilcoxon signed-rank test [198] is a non-parametric statistical hypothesis test used when comparing two related samples to assess whether their population mean ranks differ (i.e., it is a paired difference test). It can be used as an alternative to the paired Student's t-test, t-test for matched pairs[3], when the population cannot be assumed to be normally distributed. Therefore, the case where one wants to use a Wilcoxon Signed-rank test is the same as a Mann-Whitney's U test but having the paired data.

Analysis of Variance

Analysis of variance (ANOVA) is a collection of statistical models used to analyze the differences between group means and their associated procedures (such as "variation" among and between groups), in which the observed variance of a particular variable is partitioned into components attributable to different sources of variation. In its simplest form, ANOVA provides a statistical test of whether or not the means of several groups are all equal, and therefore generalizes the t-test to more than two groups. Doing multiple two-sample t-tests would result in an increased chance of committing a type I error. For this reason, ANOVAs are useful in comparing three or more means (groups or variables) for statistical significance. If we compare three or more unmatched groups, we will use the one-way ANOVA test but, while the one-way ANOVA measures the significant effect of one IV, the two-way ANOVA is used when there are more than one

[3]Two data samples are matched/paired if they come from repeated observations of the same subject.

IVs and multiple observations for each IV.

Friedman test

The Friedman test [229] is a non-parametric statistical test developed by the economist Milton Friedman. Similar to the parametric repeated measures ANOVA, it is used to detect differences in treatments across multiple test attempts. The procedure involves ranking each row (or block) together, then considering the values of ranks by columns.

Kruskal-Wallis test

The Kruskal-Wallis test [197] is a non-parametric version of the classic one-way ANOVA, and an extension of the Wilcoxon rank-sum test to more than two groups. It is used for comparing more than two samples that are independent, and it is used to test the null hypothesis assuming that the samples are from identical populations or, in other words, samples from possibly different populations actually originate from similar populations, at least as far as their central tendencies, or medians, are concerned.

Post-hoc procedures

Statistical tests can perform two kinds of analysis: pairwise comparisons and multiple comparisons. Pairwise statistical procedures perform individual comparisons between two algorithms, obtaining in each application a p-value independent from another one. Therefore, in order to carry out a comparison which involves more than two algorithms, multiple comparisons tests should be used. In $1\times$N comparisons, a control method is highlighted (the best performing algorithm) through the application of the test. Then, all hypotheses of equality between the control method and the rest can be tested by the application of a set of post-hoc procedures. N$\times$N comparisons, considering the hypotheses of equality between all existing pairs of algorithms, are also possible, with the inclusion of specific post-hoc procedures for this task.

In pairwise analysis, if we try to draw a conclusion involving more than one pairwise comparison, we will obtain an accumulated error coming from its combination. In statistical terms, we are losing control on the Family-Wise Error Rate (FWER), defined as the probability of making one or more false discoveries (type I errors) among all the hypotheses when performing multiple pairwise tests. Examples of post-hoc procedures, used to control the FWER, are Bonferroni-Dunn [230], Holm [222], Hochberg [231], Hommel [232], Holland [233], Rom [234], or Nemenyi [235].

According to [228], Holm's procedure can always be considered better than Bonferroni-Dunn's procedure, because it appropriately controls the FWER and it is more powerful than Bonferroni-Dunn's procedure. In relation to the post-hoc procedures, the differences of power between many methods are rather small, with some exceptions: Bonferroni-Dunn test should not be used in spite of its simplicity, because it is a very conservative test and many differences may not be detected, but procedures like Holm, Hochberg, Hommel, Holland, or Rom have a similar power (although the Hommel and Rom procedures are the most difficult to apply and understand). In any case, the use of the most powerful statistical procedures does not imply that the results obtained will be better: the choice of a statistical technique is ruled by a trade-off between its power and its complexity, taking into account the fulfillment of the assumptions to use every test (normality, homoscedasticity, statistically independent) as well as the nature of the data under study (nominal, ordinal, quantitative) and the kind of question we want to answer (with respect to means, medians, variances, proportions).

Implementation

In particular, many of these statistical tests were included in a tool implemented in collaboration with Prof. Ottmar Beucher of the University of Applied Sciences of Karlsruhe, taking advantage of the features offered by the MATLAB Statistic Toolbox, and a software developed in JAVA by the "Soft Computing and Intelligent Information Systems" research group at the University of Granada (`http://sci2s.ugr.es/sicidm/`). In Figure 9.1, the main pipeline of the statistical framework used to check the results obtained in the experiments is shown. First, after checking the normality and

homoscedasticity assumptions, we decide between using parametric or non-parametric approaches. After that, depending on the number of methods under study (two or more than two), the particular kind of statistical test is chosen. It is important to note that, usually, we do not use two-way ANOVA because, for such a test, the data in different columns represent changes in one factor and the data in different rows represent changes in the other factor. But, in our data, not all rows indicate a possible influence factor in the results, i.e. a group of rows (a function/dataset/problem) can be a factor, but not every single row (we are not using averages, we are usually using all the results). In this scenario, what we are testing with ANOVA is the existence of general differences between the different methods omitting any reference to every single function or dataset.

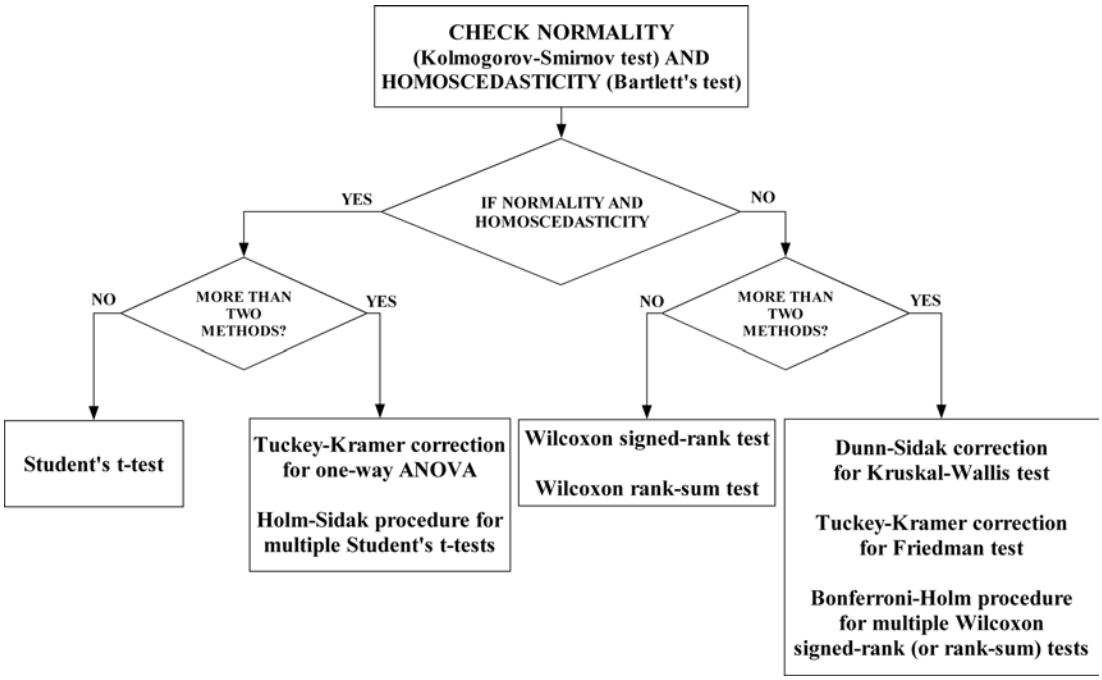


Figure 9.1: General overview of the statistical procedures used.

Appendix II: Standard Segmentation Metrics

Throughout this PhD dissertation, standard segmentation metrics were used to evaluate the performance of the different segmentation methods compared. The five metrics most commonly used in this PhD are introduced and described in this appendix.

The Jaccard Index (JI) and the Dice Similarity Coefficient (DSC) measure set agreement: a value of 0 indicates no overlap with the ground truth, and a value of 1 indicates perfect agreement. Both can be calculated applying the following formulas:

$$DSC = \frac{2 * |GT \cap segmentation|}{|GT \cap segmentation| + |GT \cup segmentation|} \quad (9.1)$$

$$= \frac{2 * |GT \cap segmentation|}{|GT| + |segmentation|} \quad (9.2)$$

$$= \frac{2 * TP}{TP + FP + TP + FN} \quad (9.3)$$

$$JI = \frac{|GT \cap segmentation|}{|GT \cup segmentation|} \quad (9.4)$$

$$= \frac{DSC}{2 - DSC} \quad (9.5)$$

where TP, FN and FP are, respectively, the number of true positive, false negative, and false positive pixels with respect to the whole image; GT is the ground truth used to calculate the metric, and $segmentation$ is the actual result obtained using the automatic segmentation method under consideration. This formula represents the size of the union of 2 sets divided by the average size of the two sets.

In turn, the Hausdorff Distance (HD) represents a measure of the spatial distance between two sets of points: it is the largest of all distances from any point in the resulting segmentation to the closest point in the ground truth. HD gives an interesting measure of the mutual proximity of two images, by indicating the maximal distance between any point of one image to the other image. HD can be calculated as $HD(GT, segmentation) = max(h(GT, segmentation), h(segmentation, GT))$ where $h(GT, segmentation) = max_{gt \in GT}\{min_{s \in segmentation}\{d(gt, s)\}\}$, and $d(gt, s)$ is any metric between these points (gt and s) [4].

Finally, the True Positive (TP) and False Positive (FP) rates [5], used in Tables 5.3 and 5.5, refer to the percentage of pixels belonging to the object of interest correctly segmented, and the percentage of pixels erroneously segmented (i.e., pixels that should have not been segmented), respectively.

[4]The following web page is a good reference with respect to the HD: `http://cgm.cs.mcgill.ca/~godfried/teaching/cg-projects/98/normand/main.html`

[5]False Negatives (FN) are not taken into consideration since they can be easily calculated from TP as FN = 100 - TP.

Appendix III: Publications July 2010-July 2013

This chapter contains the scientific publications, related to the fields of medical image processing and/or soft computing research, produced during my Marie Curie ESR contract, within which the work described in this dissertation has been produced.

- JCR-SCI Indexed Journal Papers (5)
  - Ugolotti, R., *Mesejo, P.*, Zongaro, S., Bardoni, B., Berto, G., Bianchi, F., Molineris, I., Giacobini, M., Cagnoni, S., and Di Cunto, F., **"Visual search of neuropil-enriched RNAs from brain in situ hybridization data through the image analysis pipeline Hippo-ATESC"**, PLoS ONE 8 (9): e74481. doi:10.1371/journal.pone.0074481, September 2013 (IF2011: 4.092, 12th of 85 journals indexed in subject category BIOLOGY, 1st quartile)
  - *Mesejo, P.*, Valsecchi, A., Marrakchi-Kacem, L., Cagnoni, S., and Damas, S., **"Biomedical Image Segmentation using Geometric Deformable Models and Metaheuristics"**, Computerized Medical Imaging and Graphics, 2013. In Press. (IF2011: 1.467, 43rd of 72 journals indexed in subject category BIOMEDICAL ENGINEERING and 72nd of 116 journals indexed in subject category RADIOLOGY, NUCLEAR MEDICINE & MEDICAL IMAGING, 3rd quartile)
  - *Mesejo, P.*, Ugolotti, R., Di Cunto, F., Giacobini, M., Cagnoni, S., **"Automatic Hippocampus Localization in Histological Images using Differ-**

ential Evolution-Based Deformable Models", Pattern Recognition Letters 34 (3), 299-307, February 2013 (IF2011: 1.034, 63rd of 111 journals indexed in subject category COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE, 3rd quartile)

- Ugolotti, R., Nashed, Y., *Mesejo, P.*, Ivekovič, S., Mussi, L., Cagnoni, S., **"Particle Swarm Optimization and Differential Evolution for Model-based Object Detection"**, Applied Soft Computing 13 (6), 3092-3105, June 2013 (IF2011: 2.612, 13th of 111 journals indexed in subject category COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE, 1st quartile)
- Porto-Pazos, A., Veiguela, N., *Mesejo, P.*, Navarrete, M., Alvarellos, A., Ibáñez, O., Munteanu, C., Pazos, A., Araque, A., **"Artificial Astrocytes improve neural network performance"**, PLoS ONE 6 (4): e19109, doi: 10.1371/journal.pone.0019109, April 2011 (IF2010: 4.411, 12th of 85 journals indexed in subject category BIOLOGY, 1st quartile)

- Conference Papers (12)

  - Ugolotti, R., *Mesejo, P.*, Nashed, Y., and Cagnoni, S., **"Automatic Configuration of a Hippocampus Localization System using GPU-based Differential Evolution"**, Proceedings of the 16th Portuguese Conference on Artificial Intelligence (EPIA'13), Lecture Notes in Computer Science Volume 8154, 114-125, Azores, September - 2013.
  - Ugolotti, R., Nashed, Y.S.G., *Mesejo, P.*, and Cagnoni, S., **"Algorithm Configuration using GPU-based Metaheuristics"**, Proceedings of the 15th annual conference companion on Genetic and Evolutionary Computation (GECCO'13), 221-222, Amsterdam, July - 2013.
  - Fernandez-Lozano, C., Seoane, J.A., *Mesejo, P.*, Nashed, Y., Cagnoni, S., and Dorado, J., **"Texture Classification of Proteins Using Support Vector Machines and Bio-Inspired Metaheuristics"**, Lecture Notes in Computer Science, series "Communications in Computer and Information Science", 2013. In Press.

- *Mesejo, P.*, Cagnoni, S., Costalunga, A., and Valeriani, D., **"Segmentation of Histological Images using a Metaheuristic-based Level Set Approach"**, 9th GECCO Workshop on Medical Applications of Genetic and Evolutionary Computation (MedGec), Proceedings of the 15th annual conference companion on Genetic and Evolutionary Computation (GECCO'13), 1455-1462, Amsterdam, July - 2013
- *Mesejo, P.* and Cagnoni, S., **"An experimental study on the automatic segmentation of in situ hybridization-derived images"**, Proceedings of the 1st International Conference on Medical Imaging using Bio-Inspired and Soft Computing (MIBISOC'13), 153-160, Brussels, May - 2013
- Fernandez-Lozano, C., Seoane, J.A., *Mesejo, P.*, Nashed, Y., Cagnoni, S., and Dorado, J., **"2D-PAGE Texture Classification Using Support Vector Machines and Genetic Algorithms"**, Proceedings of the 6th International Joint Conference on Biomedical Engineering Systems and Technologies: International Conference on Bioinformatics Models, Methods and Algorithms (BIOSTEC'13), 5-14, Barcelona, February - 2013.
- Nashed, Y., *Mesejo, P.*, Ugolotti, R., Dubois-Lacoste, J., Cagnoni, S., **"A comparative study of three GPU-based metaheuristics"**, Proceedings of the 12th International Conference on Parallel Problem Solving from Nature (PPSN'12), Volume Part II, 398-407, Taormina, September - 2012
- Nashed, Y., Ugolotti, R., *Mesejo, P.*, Cagnoni, S., **"libCudaOptimize: an Open Source Library of GPU-based Metaheuristics"**, 1st GECCO Workshop on Evolutionary Computation Software Systems (EvoSoft), Proceedings of the 14th annual conference companion on Genetic and Evolutionary Computation (GECCO'12), 117-124, Philadelphia, July - 2012
- Cagnoni, S., Cordón, O., *Mesejo, P.*, Nashed, Y., Ugolotti, R., **"First Results and Future Developments of the MIBISOC Project in the IBISlab of the University of Parma"**, 8th GECCO Workshop on Medical Applications of Genetic and Evolutionary Computation (MedGec), Proceedings of the 14th annual conference companion on Genetic and Evolutionary

Computation (GECCO'12), 509-516, Philadelphia, July - 2012

– *Mesejo, P.*, Ugolotti, R., Di Cunto, F., Cagnoni, S., Giacobini, M., **"Automatic Segmentation of Hippocampus in Histological Images of Mouse Brains using Deformable Models and Random Forest"**, Proceedings of the 25th IEEE International Symposium on Computer-Based Medical Systems (CBMS'12), 1-4, Rome, June - 2012

– Romero-Porta, F., *Mesejo P.*, Ibáñez O. y Porto-Pazos, A. B., **"Optimización mediante Computación Evolutiva de la interacción Neurona-Astrocito en Redes Neurogliales Artificiales"**, Proceedings of the 8th Spanish Conference on Metaheuristics, Evolutionary and Bioinspired Algorithms (MAEB'12), Albacete, February - 2012

– Ugolotti, R., *Mesejo, P.*, Cagnoni, S., Giacobini, M., Di Cunto, F., **"Automatic Hippocampus Localization in Histological Images using PSO-based Deformable Models"**, 7th GECCO Workshop on Medical Applications of Genetic and Evolutionary Computation (MedGec), Proceedings of the 13th annual conference companion on Genetic and Evolutionary Computation (GECCO'11), 487-494, Dublin, July - 2011

Appendix IV: List of Abbreviations

AAM Active Appearance Model

ABA Allen Brain Atlas

ACM Active Contour Model

ACO Ant Colony Optimization

AI Artificial Intelligence

AL Artificial Life

ANN Artificial Neural Network

ASM Active Shape Model

ASM+RF Active Shape Model (and Iterative Otsu's Thresholding Method) refined using Random Forests

CA Cornu Ammonis (Ammon's Horn)

CT Computed Tomography

CV Chan and Vese segmentation method

DCO Discrete Crossover Operators

DE Differential Evolution

DG Dentate Gyrus

DM Deformable Model

DSC Dice Similarity Coefficient

DT Deformable Template

DTr Decision Tree

EA Evolutionary Algorithm

EC Evolutionary Computation

ECL Ensemble Classifier

EP Evolutionary Programming

ES Evolution Strategies

ESR Early Stage Researcher

FP False Positives

FS Fuzzy Systems

GA Genetic Algorithm

GAC Geodesic Active Contours

GDT Geometrically Deformable Template

GLCM Gray Level Co-Occurrence Matrix

GP Genetic Programming

GPGPU General Purpose Graphics Processing Unit

HD Hausdorff Distance

HPF Hippocampal Formation

ICM Iterated Conditional Modes

IR Image Registration

IS Image Segmentation

ISH In Situ Hybridization

JI Jaccard Index

LM Levenberg-Marquardt

LS Level Set

MA Memetic Algorithm

MH Metaheuristic

MI Medical Imaging

MIBISOC Medical Imaging using Bio-Inspired and Soft Computing

ML Machine Learning

PBR Probabilistic Reasoning

PCA Principal Component Analysis

PDM Point Distribution Model

PET Positron-Emission Tomography

PR Pattern Recognition

PSO Particle Swarm Optimization

RCGA Real-coded Genetic Algorithm

RF Random Forests

ROI Region of Interest

SA Simulated Annealing

SC Soft Computing

sg Stratum Granulosum (granule cell layer)

SI Swarm Intelligence

sp Stratum Pyramidale (pyramidal cell layer)

SPECT Single-Photon Emission Computed Tomography

SS Scatter Search

SSM Statistical Shape Model

ST Soft Thresholding

SUB Subiculum

SVM Support Vector Machines

TAN Topological Active Net

TP True Positives

US Ultrasound

VFC Vector Field Convolution

Bibliography

[1] European Centre for Soft Computing (ECSC). Medical imaging using bio-inspired and soft computing. `http://www.mibisoc-itn.eu`, 2010-13.

[2] Pablo Mesejo, Andrea Valsecchi, Linda Marrakchi-Kacem, Stefano Cagnoni, and Sergio Damas. Biomedical Image Segmentation using Geometric Deformable Models and Metaheuristics. *Computerized Medical Imaging and Graphics*, 2013. In Press.

[3] Marie-Pierre Dubuisson Jolly, Sridhar Lakshmanan, and Anil K. Jain. Vehicle segmentation and classification using deformable templates. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(3):293–308, 1996.

[4] Payel Ghosh and Melanie Mitchell. Segmentation of medical images using a genetic algorithm. In *Procs. of the Genetic and Evolutionary Computation Conference (GECCO '06)*, pages 1171–1178, 2006.

[5] Mohammadreza Heydarian, Michael D. Noseworthy, Markad V. Kamath, Colm Boylan, and W. F. S. Poehlman. Optimizing the Level Set Algorithm for Detecting Object Edges in MR and CT Images. *IEEE Trans. on Nuclear Science*, 56(1):156 –166, 2009.

[6] Chunming Li, Chenyang Xu, Changfeng Gui, and Martin D. Fox. Level set evolution without re-initialization: A new variational formulation. In *Procs. of the Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pages 430–436, 2005.

[7] Clifford Ross Slaughterbeck, Suvro Datta, Simon Smith, Dan Ritchie, and Paul Wohnoutka. High-Throughput Automated Microscopy Platform for the Allen Brain Atlas. *Journal of the Association for Laboratory Automation*, 12(6):377–383, 2007.

[8] F.J. Cabrero Fraile. *Imagen Radiológica: Principios Físicos e Instrumentación*. Masson, 2004.

[9] Stuart J. Russell and Peter Norvig. *Artificial Intelligence - A Modern Approach*. Pearson Education, 3rd edition, 2009.

[10] Umberto Eco. *Come si fa una tesi di laurea*. Bompiani, 18th edition, 1977.

[11] Dzung L. Pham, Chenyang Xu, and Jerry L. Prince. Current Methods in Medical Image Segmentation. *Annual Review of Biomedical Engineering*, 2:315–337, 2000.

[12] K.S. Fu and J.K. Mui. A survey on image segmentation. *Pattern Recognition*, 13(1):3 – 16, 1981.

[13] Robert M. Haralick and Linda G. Shapiro. Image segmentation techniques. *Computer Vision, Graphics, and Image Processing*, 29(1):100 – 132, 1985.

[14] Nikhil R. Pal and Sankar K. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26(9):1277 – 1294, 1993.

[15] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Trans. on Systems, Man and Cybernetics*, 9(1):62 –66, 1979.

[16] S. Aja-Fernandez, G. Vegas-Sanchez-Ferrero, and M.A. Martin Fernandez. Soft thresholding for medical image segmentation. In *Procs. of the International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '10)*, pages 4752 –4755, 2010.

[17] John Canny. A Computational Approach to Edge Detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[18] D. Ziou and S. Tabbone. Edge Detection Techniques - An Overview. *International Journal of Pattern Recognition and Image Analysis*, 8:537–559, 1998.

[19] H. P. Ng, S. H. Ong, K. W. C. Foong, P. S. Goh, and W. L. Nowinski. Medical image segmentation using k-means clustering and improved watershed algorithm. In *Procs. of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 61–65, 2006.

[20] J.P. Gambotto. A hierarchical segmentation algorithm. In *Procs. of the IEEE Conference on Pattern Recognition*, pages 951–953, 1986.

[21] Yian-Leng Chang and Xiaobo Li. Adaptive image region-growing. *IEEE Trans. on Image Processing*, 3(6):868–872, 1994.

[22] D. Terzopoulos and K. Fleischer. Deformable models. *The Visual Computer*, 4:306–331, 1988.

[23] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on deformable models: Recovering 3D shape and nonrigid motion. *Artificial Intelligence*, 36:91–123, 1988.

[24] Chenyang Xu, Dzung L. Pham, and Jerry L. Prince. Image segmentation using deformable models. In *Handbook of Medical Imaging. Vol.2 Medical Image Processing and Analysis*, pages 175–272. Springer Verlag, 2000.

[25] Anil K. Jain, Yu Zhong, and Sridhar Lakshmanan. Object matching using deformable templates. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18:267–278, 1996.

[26] Yu Zhong and Anil K. Jain. Object tracking using deformable templates. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22:544–549, 2000.

[27] T.F. Cootes, A. Hill, C.J. Taylor, and J. Haslam. The Use of Active Shape Models For Locating Structures in Medical Images. In *Procs. of the International Conference on Information Processing in Medical Imaging (IPMI '93)*, pages 33–47, 1994.

[28] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1:321–331, 1988.

[29] Isaac Cohen, Laurent D. Cohen, and Nicholas Ayache. Using deformable surfaces to segment 3-d images and infer differential structures. *CVGIP: Image Understanding*, 56(2):242–263, 1992.

[30] Tim Mcinerney and Demetri Terzopoulos. Topology adaptive deformable surfaces for medical image volume segmentation. *IEEE Trans. on Medical Imaging*, 18:840–850, 1999.

[31] Johan Montagnat, Hervé Delingette, and Nicholas Ayache. A review of deformable surfaces: topology, geometry and deformation. *Image and Vision Computing*, 9(14):1023–1040, 2001.

[32] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active Appearance Models. In *Procs. of the European Conference on Computer Vision*, volume 2, pages 484–498, 1998.

[33] Tobias Heimann and Hans-Peter Meinzer. Statistical shape models for 3D medical image segmentation: a review. *Medical Image Analysis*, 13:543–563, 2009.

[34] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models - their training and application. *Computer Vision and Image Understanding*, 61:38–59, 1995.

[35] T. F. Cootes, G. Edwards, and C.J. Taylor. Comparing active shape models with active appearance models. In *Procs. of the British Machine Vision Conference (BMVC '99)*, pages 173–182, 1999.

[36] M. Bro-Nielsen. Active nets and cubes. *IMM Technical Report*, 1994.

[37] V. Caselles. Geometric models for active contours. In *Procs. of International Conference on Image Processing*, volume 3, 1995.

[38] Ravikanth Malladi, James A. Sethian, and Baba C. Vemuri. Shape modeling with front propagation: a level set approach. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(2):158–175, 1995.

[39] Benjamin B. Kimia, Allen R. Tannenbaum, and Steven W. Zucker. Shapes, Shocks, and Deformations I: The Components of Two-Dimensional Shape and the Reaction-Diffusion Space. *International Journal of Computer Vision*, 15:189–224, 1994.

[40] Ron Kimmel, Arnon Amir, and Alfred M. Bruckstein. Finding shortest paths on surfaces using level sets propagation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(6):635–640, 1995.

[41] Guillermo Sapiro and Allen Tannenbaum. Affine invariant scale-space. *International Journal of Computer Vision*, 11(1):25–44, 1993.

[42] S. Osher and J.A. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

[43] J.A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 1999.

[44] Stanley J. Osher and Ronald P. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2002.

[45] T.F. Chan and L.A. Vese. Active contours without edges. *IEEE Trans. on Image Processing*, 10(2):266 –277, 2001.

[46] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.

[47] Vicent Caselles, Ron Kimmel, and Guillermo Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22:61–79, 1997.

[48] Barbara Zitová and Jan Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21:977–1000, 2003.

[49] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Mutual-information-based registration of medical images: a survey. *IEEE Trans. on Medical Imaging*, 22(8):986–1004, 2003.

[50] F. Maes, D. Vandermeulen, and P. Suetens. Comparative evaluation of multiresolution optimization strategies for image registration by maximization of mutual information. *Medical Image Analysis*, 3(4):373–386, 1999.

[51] S. Klein, M. Staring, and J. P W Pluim. Evaluation of optimization methods for nonrigid medical image registration using mutual information and b-splines. *IEEE Trans. on Image Processing*, 16(12):2879–2890, 2007.

[52] Sergio Damas, Oscar Cordón, and José Santamaría. Medical image registration using evolutionary computation: An experimental survey. *IEEE Computational Intelligence Magazine*, 6(4):26 –42, nov. 2011.

[53] Mariano Cabezas, Arnau Oliver, Xavier Lladó, Jordi Freixenet, and Meritxell Bach Cuadra. A review of atlas-based segmentation for magnetic resonance brain images. *Computer Methods and Programs in Biomedicine*, 104(3):e158 – e177, 2011.

[54] Robert M. Haralick, K. Shanmugam, and Its'Hak Dinstein. Textural Features for Image Classification. *IEEE Trans. on Systems, Man and Cybernetics*, SMC-3(6):610–621, 1973.

[55] Andries P. Engelbrecht. *Computational Intelligence: An Introduction*. Wiley Publishing, 2nd edition, 2007.

[56] Agoston E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing*. Springer Verlag, 2003.

[57] M. Dorigo and M. Birattari. Swarm Intelligence. `http://www.scholarpedia.org/article/Swarm_intelligence`, 2007.

[58] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, 1999.

[59] Christopher M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, 1995.

[60] Simon Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1999.

[61] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[62] Jyh shing Roger Jang. Anfis: Adaptive-network-based fuzzy inference system. *IEEE Trans. on Systems, Man, and Cybernetics*, 23:665–685, 1993.

[63] L.-X. Wang and J.M. Mendel. Generating fuzzy rules by learning from examples. *IEEE Trans. on Systems, Man and Cybernetics*, 22(6):1414 –1427, 1992.

[64] Nils J. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28(1):71–87, 1986.

[65] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[66] José Santos Reyes. *Vida Artificial. Realizaciones Computacionales*. Servizo de Publicacións da Universidade da Coruña, 2007.

[67] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, 2006.

[68] Ethem Alpaydin. *Introduction to Machine Learning*. MIT Press, 2nd edition, 2010.

[69] John Haugeland. *Artificial intelligence: the very idea*. MIT press, 1985.

[70] Richard Ernest Bellman. *An introduction to artificial intelligence : can computers think?* Boyd & Fraser, 1978.

[71] Eugene Charniak and Drew McDermott. *Introduction to artificial intelligence*. Addison-Wesley Longman, 1985.

[72] Patrick Henry Winston. *Artificial intelligence*. Addison-Wesley, 3rd edition, 1992.

[73] Raymond Kurzweil. *Age of intelligent machines*. MIT Press, 1990.

[74] Elaine Rich and Kevin Knight. *Artificial intelligence*. McGraw-Hill, 1991.

[75] Robert J. Schalkoff. *Artificial Intelligence: An Engineering Approach*. Mcgraw-Hill, 1990.

[76] George F. Luger. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. Addison Wesley, 1993.

[77] U. Maulik. Medical image segmentation using genetic algorithms. *IEEE Trans. on Information Technology in Biomedicine*, 13:166–173, 2009.

[78] J. Grefenstette. Optimization of control parameters for genetic algorithms. *IEEE Trans. on System, Man and Cybernetics*, 16(1):122–128, 1986.

[79] Vasileios Zografos. Comparison of optimisation algorithms for deformable template matching. In *Procs. of the International Symposium on Advances in Visual Computing (ISVC '09)*, pages 1097–1108, 2009.

[80] Sean Luke. *Essentials of Metaheuristics*. Lulu, 2009. Available for free at http://cs.gmu.edu/∼sean/book/metaheuristics/.

[81] Fred Glover and Manuel Laguna. *Tabu Search*. Kluwer Academic Publishers, 1997.

[82] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by Simulated Annealing. *Science*, 220:671–680, 1983.

[83] Matthijs den Besten, Thomas Stützle, and Marco Dorigo. Design of iterated local search algorithms. In *Procs. of the Applications of Evolutionary Computing (EvoWorkshops '01)*, pages 441–451, 2001.

[84] Pierre Hansen, Nenad Mladenovic, and José A. Moreno-Pérez. Variable neighbourhood search: methods and applications. *Annals of Operations Research*, 175(1):367–407, 2010.

[85] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, 1999.

[86] Pablo Moscato. A gentle introduction to memetic algorithms. In *Handbook of Metaheuristics*, pages 105–144. Kluwer Academic Publishers, 2003.

[87] J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

[88] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman, 1989.

[89] Darrell Whitley. A genetic algorithm tutorial. *Statistics and Computing*, 4:65–85, 1994.

[90] Larry J. Eshelman. The CHC Adaptive Search Algorithm : How to Have Safe Search When Engaging in Nontraditional Genetic Recombination. *Foundations of Genetic Algorithms*, pages 265–283, 1991.

[91] F. Herrera, M. Lozano, and A. M. Sánchez. A taxonomy for the crossover operator for real-coded Genetic Algorithms: An experimental study. *International Journal of Intelligent Systems*, 18:309–338, 2003.

[92] Francisco Herrera, Manuel Lozano, and Jose L. Verdegay. Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis. *Artificial Intelligence Review*, 12:265–319, 1998.

[93] J. Kennedy and R. Eberhart. Particle Swarm Optimization. In *Procs. of IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948, 1995.

[94] Riccardo Poli. Analysis of the publications on the applications of Particle Swarm Optimisation. *J. Artificial Evolution and Applications*, pages 1–10, 2008.

[95] Alec Banks, Jonathan Vincent, and Chukwudi Anyakoha. A review of Particle Swarm Optimization. Part I: background and development. *Natural Computing*, 6:467–484, 2007.

[96] Bo Liu, Ling Wang, Yi-Hui Jin, Fang Tang, and De-Xian Huang. Improved Particle Swarm Optimization combined with chaos. *Chaos, Solitons & Fractals*, 25:1261–1271, 2005.

[97] R. Storn and K. Price. Differential Evolution- A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. Technical report, International Computer Science Institute, 1995.

[98] J. Vesterstrom and R. Thomsen. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In *Procs. of the Congress on Evolutionary Computation (CEC '04)*, pages 1980–1987, 2004.

[99] S. Das and P.N. Suganthan. Differential Evolution: A Survey of the State-of-the-Art. *IEEE Trans. on Evolutionary Computation*, 15:4–31, 2011.

[100] N. Krasnogor and J. Smith. A tutorial for competent memetic algorithms: model, taxonomy, and design issues. *IEEE Trans. on Evolutionary Computation*, 9(5):474–488, 2005.

[101] Richard Dawkins. *The Selfish Gene*. Oxford University Press, 1976.

[102] M. Laguna and R. Martí. *Scatter search: methodology and implementations in C*. Kluwer Academic Publishers, 2003.

[103] Francisco J. Solis and Roger J. B. Wets. Minimization by Random Search Techniques. *Mathematics of Operations Research*, 6(1):19–30, 1981.

[104] J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.

[105] Manuel Lozano, Francisco Herrera, Natalio Krasnogor, and Daniel Molina. Real-coded memetic algorithms with crossover hill-climbing. *Evolutionary Computation*, 12(3):273–302, 2004.

[106] F. Herrera, M. Lozano, and D. Molina. Continuous scatter search: An analysis of the integration of some combination methods and improvement strategies. *European Journal of Operational Research*, 169(2):450–476, 2006.

[107] O. Cordón, S. Damas, and J. Santamaría. A fast and accurate approach for 3D image registration using the scatter search evolutionary algorithm. *Pattern Recognition Letters*, 27:1191–1200, 2006.

[108] Daniel Molina, Manuel Lozano, Ana M. Sánchez, and Francisco Herrera. Memetic algorithms based on local search chains for large scale continuous optimisation problems: MA-SSW-Chains. *Soft Computing*, 15(11):2201–2220, 2011.

[109] Ingo Rechenberg. *Evolutionstrategie: Optimierung Technisher Systeme nach Prinzipien des Biologischen Evolution*. Fromman-Hozlboog Verlag, 1973.

[110] Hans-Paul Schwefel. *Numerical Optimization of Computer Models*. John Wiley & Sons, Inc., 1981.

[111] N. Hansen. The CMA evolution strategy: a comparing review. In *Towards a new evolutionary computation. Advances on estimation of distribution algorithms*, pages 75–102. Springer, 2006.

[112] Thomas Bäck and Hans-Paul Schwefel. An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation*, 1(1):1–23, 1993.

[113] L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence through Simulated Evolution*. John Wiley, 1966.

[114] D. B. Fogel. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. IEEE Press Series on Computational Intelligence. 3rd edition, 2005.

[115] A. E. Eiben and Marc Schoenauer. Evolutionary computing. *Information Processing Letters*, 82(1):1–6, 2002.

[116] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection (Complex Adaptive Systems)*. A Bradford Book, 1 edition, 1992.

[117] Wolfgang Banzhaf, Frank D. Francone, Robert E. Keller, and Peter Nordin. *Genetic programming: an introduction: on the automatic evolution of computer programs and its applications*. Morgan Kaufmann Publishers Inc., 1998.

[118] Lior Rokach. Ensemble-based classifiers. *Artificial Intelligence Review*, 33:1–39, 2010.

[119] David Opitz and Richard Maclin. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198, 1999.

[120] T. G. Dietterich. Ensemble learning. In M.A. Arbib, editor, *Handbook of Brain Theory and Neural Networks*. MIT Press, 2002.

[121] Leo Breiman. Random forests. *Maching Learning*, 45:5–32, 2001.

[122] Leo Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, 1984.

[123] Pablo Mesejo, Roberto Ugolotti, Ferdinando Di Cunto, Mario Giacobini, and Stefano Cagnoni. Automatic hippocampus localization in histological images using Differential Evolution-based deformable models. *Pattern Recognition Letters*, 34(3):299 – 307, 2013.

[124] Lucia Ballerini. Genetic snakes for medical images segmentation. In *Evolutionary Image Analysis, Signal Processing and Telecommunications*, volume 1596, pages 59–73. 1999.

[125] Ding-Horng Chen and Yung-Nien Sun. A self-learning segmentation framework—the taguchi approach. *Computerized Medical Imaging and Graphics*, 24(5):283 – 296, 2000.

[126] Chris McIntosh and Ghassan Hamarneh. Medial-based deformable models in non-convex shape-spaces for medical image segmentation using genetic algorithms. *IEEE Trans. on Medical Imaging*, 31(1):33–50, 2012.

[127] Chih-Yu Hsu, Chun-You Liu, and Chung-Ming Chen. Automatic segmentation of liver pet images. *Computerized Medical Imaging and Graphics*, 32(7):601 – 610, 2008.

[128] S. Cagnoni, A. B. Dobrzeniecki, R. Poli, and J. C. Yanch. Genetic algorithm-based interactive segmentation of 3D medical images. *Image and Vision Computing*, 17:881–895, 1999.

[129] M.A. Asl and S.A. Seyedin. Active contour optimization using particle swarm optimizer. In *Procs. of the International Conference on Information and Communication Technologies (ICTTA '06)*, volume 1, pages 1522–1523, 2006.

[130] L.A. MacEachern and T. Manku. Genetic algorithms for active contour optimization. In *Procs. of the IEEE International Symposium on Circuits and Systems (ISCAS '98)*, volume 4, pages 229–232, 1998.

[131] Chun-Chieh Tseng, Jer-Guang Hsieh, and Jyh-Horng Jeng. Active contour model via multi-population particle swarm optimization. *Expert Systems with Applications*, 36:5348–5352, 2009.

[132] O. Ibáñez, N. Barreira, J. Santos, and M. G. Penedo. Genetic approaches for topological active nets optimization. *Pattern Recognition*, 42(5):907–917, 2009.

[133] J. Novo, J. Santos, and M. G. Penedo. Topological active models optimization with differential evolution. *Expert Systems with Applications*, 39(15):12165–12176, 2012.

[134] Nicola Bova, Óscar Ibáñez, and Oscar Cordón. Image segmentation using extended topological active nets optimized by scatter search. *IEEE Computational Intelligence Magazine*, 8(1):16–32, 2013.

[135] Payel Ghosh, Melanie Mitchell, James A. Tanyi, and Arthur Hung. A Genetic Algorithm-Based Level Set Curve Evolution for Prostate Segmentation on Pelvic CT and MRI Images. In *Biomedical Image Analysis and Machine Learning Technologies: Applications and Techniques*, pages 127–149. IGI Global, 2010.

[136] David Feltell and Li Bai. 3d level set image segmentation refined by intelligent agent swarm. In *Procs. of the IEEE Congress on Evolutionary Computation (CEC '10)*, pages 1–8, 2010.

[137] Stefano Cagnoni. Evolutionary computer vision: A taxonomic tutorial. In *8th International Conference on Hybrid Intelligent Systems (HIS '08)*, pages 1–6, 2008.

[138] Andrew Hill, Christopher J. Taylor, and Timothy F. Cootes. Object Recognition by Flexible Template Matching using Genetic Algorithms. In *Procs. of the European Conference on Computer Vision (ECCV '92)*, volume 588, pages 852–856, 1992.

[139] A. Hill and C. J. Taylor. Model-based image interpretation using genetic algorithms. *Image and Vision Computing*, 10(5):295–300, 1992.

[140] M Mignotte, J Meunier, and JC Tardif. Endocardial boundary estimation and tracking in echocardiographic images Ăusing deformable templates and Markov Random Fields. *Pattern Analysis and Applications*, 4(4):256–271, 2001.

[141] S Karungaru, M Fukumi, and N Akamatsu. Feature extraction for face detection and recognition. In *Procs. of the International Workshop on Robot and Human Interactive Communication (RO-MAN '04)*, pages 235–239, 2004.

[142] Vavilin Andrey and Kang-Hyun Jo. Road guidance sign recognition in urban areas by structure. In *Procs. of the International Forum on Strategic Technology (IFOST '06)*, pages 293–296, 2006.

[143] Abdul Sattar, Yasser Aidarous, and Renaud Seguier. GAGM-AAM: A genetic optimization with gaussian mixtures for active appearence models. In *Procs. of the International Conference on Image Processing (ICIP '08)*, pages 3220–3223, 2008.

[144] Abdul Sattar and Renaud Seguier. GGA-AAM: Novel Heuristic Method of Gradient Driven Genetic Algorithm for Active Appearance Models. In *Procs. of the International Conference on Digital Information Management (ICDIM '08)*, pages 47–52, 2008.

[145] Abdul Sattar and Renaud Seguier. MVAAM (Multi-View Active Appearance Model) Optimized by Multi-Objective Genetic Algorithm. In *Procs. of the International Conference on Automatic Face and Gesture Recognition (FG '08)*, pages 96–103, 2008.

[146] M. Mata, J. M. Armingol, J. Fernanez, and A. de la Escalera. Object learning and detection using evolutionary deformable models for mobile robot navigation. *Robotica*, 26(Part 1):99–107, 2008.

[147] H. C. Chen, C. H. Wu, C. J. Lin, Y. H. Liu, and Y. N. Sun. Automated Segmentation for Patella from Lateral Knee X-ray Images. In *Procs. of the International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '09)*, pages 3553–3556, 2009.

[148] Sandor A. Szilagyi, Laszlo Szilagyi, David Iclanzan, and Zoltan Benyo. A Weighted Patient Specific Electromechanical Model of the Heart. In *Procs. of the International Symposium on Applied Computational Intelligence and Informatics (SACI '09)*, pages 101–106, 2009.

[149] Hiroki Takahashi, Masafumi Komatsu, Hyoungseop Kim, Joo Kooi Tan, Seiji Ishikawa, and Akiyoshi Yamamoto. Segmentation Method for Cardiac Region

in CT Images Based on Active Shape Model. In *Procs. of the International Conference on Control, Automation and Systems (ICCAS '10)*, pages 2074–2077, 2010.

[150] Chris McIntosh and Ghassan Hamarneh. Evolutionary deformable models for medical image segmentation: A genetic algorithm approach to optimizing learned, intuitive, and localized medial-based shape deformation. In *Genetic and Evolutionary Computation: Medical Applications*, pages 46–67. John Wiley & Sons Ltd, 2010.

[151] Zhi Liu, Hongjun Wang, and Wei Jiang. Automated tongue area detection for computer-aided diagnosis based on ASM and GA. *Sensor Review*, 32(1):39–46, 2012.

[152] P. Mesejo, R. Ugolotti, S. Cagnoni, F. Di Cunto, and M. Giacobini. Automatic Segmentation of Hippocampus in Histological Images of Mouse Brains using Deformable Models and Random Forest. In *Procs. of the International Symposium on Computer-Based Medical Systems (CBMS '12)*, pages 1–4, 2012.

[153] Roberto Ugolotti, Pablo Mesejo, Stefano Cagnoni, Mario Giacobini, and Ferdinando Di Cunto. Automatic Hippocampus Localization in Histological Images using PSO-Based Deformable Models. In *Procs. of the Genetic and Evolutionary Computation Conference companion (GECCO '11)*, pages 487–494, 2011.

[154] KL Sum, WH Lau, SH Leung, AWC Liew, and KW Tse. A new optimization procedure for extracting the point-based lip contour using Active Shape Model. In *Procs. of the International Conference on Acoustics Speech and Signal Processing (ICASSP '01)*, pages 1485–1488, 2001.

[155] Luca Mussi, Špela Ivekovič, and Stefano Cagnoni. Markerless Articulated Human Body Tracking from Multi-View Video with GPU-PSO. In *Procs. of the International Conference on Evolvable Systems: From Biology to Hardware (ICES '10)*, number 6274, pages 195–207, 2010.

[156] Luca Mussi, Špela Ivekovič, Youssef S. G. Nashed, and Stefano Cagnoni. Multi-view human body pose estimation with cuda-pso. *International Journal of Adaptive, Resilient and Autonomic Systems*, 3(4):51–65, 2012.

[157] Roberto Ugolotti, Youssef S.G. Nashed, Pablo Mesejo, Špela Ivekovič, Luca Mussi, and Stefano Cagnoni. Particle swarm optimization and differential evolution for model-based object detection. *Applied Soft Computing*, 13(6):3092 – 3105, 2013.

[158] Zhaozheng Yin and Robert T. Collins. Shape Constrained Figure-Ground Segmentation and Tracking. In *Procs. of the Conference on Computer Vision and Pattern Recognition (CVPR '09)*, pages 731–738, 2009.

[159] Luca Mussi, Stefano Cagnoni, Elena Cardarelli, Fabio Daolio, Paolo Medici, and PierPaolo Porta. Gpu implementation of a road sign detector based on particle swarm optimization. *Evolutionary Intelligence*, 3(3-4):155–169, 2010.

[160] Roberto Ugolotti, YoussefS.G. Nashed, and Stefano Cagnoni. Real-time gpu based road sign detection and classification. In *Procs. of Parallel Problem Solving from Nature (PPSN '12)*, volume 7491, pages 153–162. 2012.

[161] D. Rueckert and P. Burger. Geometrically deformable templates for shape-based segmentation and tracking in cardiac MR images. In *Procs. of International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 1223, pages 83–98, 1997.

[162] D. Rueckert and P. Burger. Shape-based segmentation and tracking in 4D cardiac MR images. In Troccaz, J and Grimson, E and Mosges, R, editor, *Procs. of the Joint Conference - Computer Vision, Virtual Reality and Robotics in Medicine and Medical Robotics and Computer-Assited Surgery (CVRMED-MRCAS '97)*, volume 1205, pages 43–52, 1997.

[163] A. Pitiot, A.W. Toga, and P.M. Thompson. Adaptive elastic segmentation of brain MRI via shape-model-guided evolutionary programming. *IEEE Trans. on Medical Imaging*, 21(8):910 –923, 2002.

[164] M Lalonde and L Gagnon. Variable Neighborhood Search for geometrically deformable templates. In Kasturi, R and Laurendeau, D and Suen, C, editor, *Procs. of the International Conference on Pattern Recognition*, pages 689–692, 2002.

[165] N Betrouni, MF Vermandel, J Rousseau, and S Maouche. Automatic segmentation of prostate boundaries from abdominal ultrasound images using priori knowledge. In *Procs. of the International Symposium on Biomedical Imaging (ISBI '04)*, pages 496–499, 2004.

[166] Tobias Heimann, Sascha Münzing, Hans-Peter Meinzer, and Ivo Wolf. A shape-guided deformable model with evolutionary algorithm initialization for 3d soft tissue segmentation. In *Procs. of the International Conference on Information Processing in Medical Imaging (IPMI '07)*, pages 1–12, 2007.

[167] Sebastian Colutto, Florian Fruehauf, Matthias Fuchs, and Otmar Scherzer. The CMA-ES on Riemannian Manifolds to Reconstruct Shapes in 3-D Voxel Images. *IEEE Trans. on Evolutionary Computation*, 14(2):227–245, 2010.

[168] Sarra Ben Fredj, Tristan Glatard, Christopher Casta, and Patrick Clarysse. An exploration framework for segmentation parameters spaces. In *Procs. of the International Conference on Image Processing (ICIP '11)*, 2011.

[169] D. Rueckert and P. Burger. A multiscale approach to contour fitting for MR images. In Loew, MH and Hanson, KM, editor, *Procs. of the Conference on Image Processing - Medical Imaging*, volume 2710, pages 289–300, 1996.

[170] D. Rueckert. *Segmentation and Tracking in cardiovascular MR Images using Geometrically Deformable Models and Templates*. PhD thesis, University of London, 1997.

[171] P.P.C. Yip and Yoh-Han Pao. Combinatorial optimization with use of guided evolutionary simulated annealing. *IEEE Trans. on Neural Networks*, 6(2):290–295, 1995.

[172] Ahmed Afifi, Toshiya Nakaguchi, Norimichi Tsumura, and Yoichi Miyake. Particle Swarm Optimization Based Medical Image Segmentation Technique. http://www.mi.tj.chiba-u.jp/~tsumura/Tsumura/papers/JAMITI2010afifi.pdf, 2010.

[173] Pablo Mesejo, Stefano Cagnoni, Alessandro Costalunga, and Davide Valeriani. Segmentation of Histological Images using a Metaheuristicbased Level Set Approach. In *Procs. of the Genetic and Evolutionary Computation Conference companion (GECCO '13)*, 2013.

[174] Jiangming Kan, Hongjun Li, and Wenbin Li. Level Set method in standing tree image segmentation based on particle swarm optimization. In *Procs. of the Automatic Target Recognition and Image Analysis Conference (MIPPR '07)*, volume 6786, page V7864, 2007.

[175] Min Xiao, Shunren Xia, and Shiwei Wang. Geometric active contour model with color and intensity priors for medical image segmentation. In *Procs. of the International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6496–6499, 2005.

[176] X Wang, L He, Chia Y. Han, and W Wee. Deformable contour method: A constrained optimization approach. In *Procs. of the British Machine Vision Conference (BMVC '02)*, pages 183–192, 2002.

[177] Dimitris K. Iakovidis, Michalis A. Savelonas, Stavros A. Karkanis, and Dimitris E. Maroulis. A genetically optimized level set approach to segmentation of thyroid ultrasound images. *Applied Intelligence*, 27(3):193–203, 2007.

[178] D.A. Oliveira, R.Q. Feitosa, and M.M. Correia. Genetic adaptation of level sets parameters for medical imaging segmentation. In *Biomedical Image Analysis and Machine Learning Technologies: Applications and Techniques*, pages 150–165. IGI Global, 2010.

[179] D.A. Oliveira, R.Q. Feitosa, and M.M. Correia. Segmentation of liver, its vessels

and lesions from CT images for surgical planning. *BioMedical Engineering OnLine*, 10, 2011.

[180] P. Mesejo and S. Cagnoni. An experimental study on the automatic segmentation of In Situ Hybridization-derived images. In *Procs. of the International Conference on Medical Imaging using Bio-Inspired and Soft Computing (MIBISOC'13)*, pages 153–160, 2013.

[181] Yan Nei Law, Hwee Kuan Lee, and Andy M. Yip. A Multiresolution Stochastic Level Set Method for Mumford-Shah Image Segmentation. *IEEE Trans. on Image Processing*, 17(12):2289–2300, DEC 2008.

[182] X Wang, L He, and W Wee. Deformable contour method: A constrained optimization approach. *International Journal of Computer Vision*, 59(1):87–108, 2004.

[183] Vincenzo Della Mea. What is e-health (2): The death of telemedicine? *Journal of Medical Internet Research*, 3(2):e22, 2001.

[184] Allen Institute for Brain Science. Allen Reference Atlases. `http://mouse.brain-map.org`, 2004-2006.

[185] Michael J. Hawrylycz et al. An anatomically comprehensive atlas of the adult human brain transcriptome. *Nature*, 489(7416):391–399, 2012.

[186] Long Jin and Ricardo V. Lloyd. In situ hybridization: Methods and applications. *Journal of Clinical Laboratory Analysis*, 11:2–9, 1997.

[187] Guido Valli and Giuseppe Coppini. *Bioimmagini*. Pàtron, 2002.

[188] C. Poupon, F. Poupon, L. Allirol, and J.-F. Mangin. A database dedicated to anatomo-functional study of human brain connectivity. In *Procs. of the Annual Meeting of the Organization for Human Brain Mapping*, number 646, 2006.

[189] Kenneth A. Norman. How hippocampus and cortex contribute to recognition memory: revisiting the complementary learning systems model. *Hippocampus*, 20:1217–1227, 2010.

[190] K. Levenberg. A method for the solution of certain nonlinear problems in least squares. *Quarterly of Applied Mathematics*, 2(2):164–168, 1944.

[191] Anton Osokin, Dmitry Vetrov, and Dmitry Kropotov. 3-D Mouse Brain Model Reconstruction from a Sequence of 2-D Slices in Application to Allen Brain Atlas. In *Computational Intelligence Methods for Bioinformatics and Biostatistics*, volume 6160, pages 291–303. 2010.

[192] Paul Yushkevich, Brian Avants, Lydia Ng, Michael Hawrylycz, Pablo Burstein, Hui Zhang, and James Gee. 3D mouse brain reconstruction from histology using a coarse-to-fine approach. In *Biomedical Image Registration*, volume 4057, pages 230–237. 2006.

[193] S. Gefen, N. Kiryati, and J. Nissanov. Atlas-Based indexing of brain sections via 2-D to 3-D image registration. *IEEE Trans. on Biomedical Engineering*, 55:147–156, 2008.

[194] T. Riklin-Raviv, N. Sochen, N. Kiryati, N. Ben-Zadok, S. Gefen, L. Bertand, and J. Nissanov. Propagating distributions for segmentation of brain atlas. In *Procs. of the International Symposium on Biomedical Imaging: From Nano to Macro (ISBI '07)*, pages 1304–1307, 2007.

[195] O. V. Senyukova, A. S. Lukin, and D. P. Vetrov. Automated atlas-based segmentation of Nissl-stained mouse brain sections using supervised learning. *Programming and Computing Software*, 37:245–251, 2011.

[196] F. Herrera, M. Lozano, and D. Molina. Continuous scatter search: An analysis of the integration of some combination methods and improvement strategies. *European Journal of Operational Research*, 169:450–476, 2006.

[197] William H. Kruskal and W. Allen Wallis. Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association*, 47:583–621, 1952.

[198] Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, pages 80–83, 1945.

[199] Stefano Cagnoni, Oscar Cordón, Pablo Mesejo, Youssef S.G. Nashed, and Roberto Ugolotti. First results and future developments of the MIBISOC Project in the IBISlab of the University of Parma. In *Procs. of the Genetic and Evolutionary Computation Conference companion (GECCO '12)*, pages 509–516, 2012.

[200] Simon Perreault and Patrick Hébert. Median filtering in constant time. *IEEE Trans. on Image Processing*, 16(9):2389–2394, 2007.

[201] Kolawole O. Babalola, Brian Patenaude, Paul Aljabar, Julia Schnabel, David Kennedy, William Crum, Stephen Smith, Tim F. Cootes, Mark Jenkinson, and Daniel Rueckert. Comparison and Evaluation of Segmentation Techniques for Subcortical Structures in Brain MRI. In *Procs. of the Medical Image Computing and Computer Assisted Intervention Conference (MICCAI '08)*, pages 409–416, 2008.

[202] Simon K. Warfield, Kelly H. Zou, and William M. Wells. Simultaneous truth and performance level estimation (STAPLE): An algorithm for the validation of image segmentation. *IEEE Trans. on Medical Imaging*, pages 903–921, 2004.

[203] Alberto M. Biancardi, Artit C. Jirapatnakul, and Anthony P. Reeves. A comparison of ground truth estimation methods. *International Journal of Computer Assisted Radiology and Surgery*, 5(3):295–305, 2010.

[204] Peter Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20(1):53–65, 1987.

[205] Glynn Dennis, Brad T. Sherman, Douglas A. Hosack, Jun Yang, Wei Gao, H. Clifford Lane, and Richard A. Lempicki. DAVID: Database for Annotation, Visualization, and Integrated Discovery. *Genome biology*, 4(5), 2003.

[206] Andy Tsai, Anthony J. Yezzi, William M. Wells III, Clare M. Tempany, Dewey Tucker, Ayres C. Fan, W. Eric L. Grimson, and Alan S. Willsky. A Shape-Based

Approach to the Segmentation of Medical Imagery Using Level Sets. *IEEE Trans. on Medical Imaging*, 22(2):137–154, 2003.

[207] Michael Leventon. *Statistical Models for Medical Image Analysis*. PhD thesis, MIT, 2000.

[208] Xiao-Feng Wang, De-Shuang Huang, and Huan Xu. An efficient local Chan-Vese model for image segmentation. *Pattern Recognition*, 43(3):603–618, 2010.

[209] Jitendra Malik, Serge Belongie, Thomas Leung, and Jianbo Shi. Contour and texture analysis for image segmentation. *International Journal of Computer Vision*, 43(1):7–27, June 2001.

[210] Yan Zhang, Bogdan J. Matuszewski, Lik-Kwan Shark, and Christopher J. Moore. Medical Image Segmentation Using New Hybrid Level-Set Method. In *Procs. of the International Conference BioMedical Visualization: Information Visualization in Medical and Biomedical Informatics*, pages 71–76, 2008.

[211] Marie Chupin, Alexander Hammers, Rebecca S. N. Liu, Olivier Colliot, J. Burdett, Eric Bardinet, John S. Duncan, Line Garnero, and Louis Lemieux. Automatic segmentation of the hippocampus and the amygdala driven by hybrid constraints: Method and validation. *NeuroImage*, 46(3):749–761, 2009.

[212] Michel Gendreau and Jean-Yves Potvin. *Handbook of Metaheuristics*. Springer, 2nd edition, 2010.

[213] Andrea Valsecchi, Sergio Damas, José Santamaría, and Linda Marrakchi-Kacem. Intensity-based image registration using scatter search. Technical Report AFE 2012-14, 2012. In Press.

[214] Oscar Cordón, Sergio Damas, Jose Santamaría, and Rafael Martí. Scatter search for the point-matching problem in 3D image registration. *INFORMS Journal on Computing*, 20(1):55–68, 2008.

[215] Bing Li, Student Member, Scott T. Acton, and Senior Member. Active contour external force using vector field convolution for image segmentation. *IEEE Trans. on Image Processing*, 16:2096–2106, 2007.

[216] Chenyang Xu and Jerry L. Prince. Gradient Vector Flow: A New External Force for Snakes. In *Procs. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pages 66–71, 1997.

[217] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley, 2nd edition, 2001.

[218] Yonggang Shi and W. C. Karl. A real-time algorithm for the approximation of level-set-based curve evolution. *IEEE Trans. on Image Processing*, 17(5):645–656, 2008.

[219] L. J. Eshelman and J. D. Schaffer. Real-coded genetic algorithms and interval-schemata. In Darrell L. Whitley, editor, *Foundation of Genetic Algorithms 2*, pages 187–202, 1993.

[220] T. Bäck, D. B. Fogel, and Z. Michalewicz. *Handbook of Evolutionary Computation*. IOP Publishing Ltd and Oxford University Press, 1997.

[221] William H. Kruskal. Historical notes on the wilcoxon unpaired two-sample test. *Journal of the American Statistical Association*, 52(279):356–360, 1957.

[222] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

[223] Kalyanmoy Deb and Deb Kalyanmoy. *Multi-Objective Optimization Using Evolutionary Algorithms*. Wiley, 2001.

[224] S. Bandyopadhyay, S. Saha, U. Maulik, and K. Deb. A Simulated Annealing-Based Multiobjective Optimization Algorithm: AMOSA. *IEEE Trans. on Evolutionary Computation*, 12(3):269–283, 2008.

[225] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[226] Salvador García, Francisco Herrera, and John Shawe-taylor. An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. *Journal of Machine Learning Research*, 9:2677–2694, 2008.

[227] Salvador García, Daniel Molina, Manuel Lozano, and Francisco Herrera. A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC'2005 Special Session on Real Parameter Optimization. *Journal of Heuristics*, 15(6):617–644, 2009.

[228] Joaquín Derrac, Salvador García, Daniel Molina, and Francisco Herrera. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation*, 1(1):3–18, 2011.

[229] Milton Friedman. A comparison of alternative tests of significance for the problem of m rankings. *The Annals of Mathematical Statistics*, 11(1):86–92, 1940.

[230] O. J. Dunn. Multiple comparisons among means. *Journal of the American Statistical Association*, 56:52–64, 1961.

[231] Yosef Hochberg. A Sharper Bonferroni Procedure for Multiple Tests of Significance. *Biometrika*, 75(4):800–802, 1988.

[232] G. Hommel. A stagewise rejective multiple test procedure based on a modified Bonferroni test. *Biometrika*, 75(2):383–386, 1988.

[233] BS Holland. An improved sequentially rejective Bonferroni test procedure. *Biometrics*, 43:417–423, 1987.

[234] D Rom. A sequentially rejective test procedure based on a modified Bonferroni inequality. *Biometrika*, 77:663–665, 1990.

[235] Peter Nemenyi. *Distribution-free multiple comparisons*. PhD thesis, Princeton University, 1963.