N° d'ordre: 2013-isal-0058 Année: 2013

Thèse

Structural Priors for Multiobject Semiautomatic Segmentation of Three-dimensional Medical Images via Clustering and Graph Cut Algorithms

Présentée devant
L'Institut National des Sciences Appliquées de Lyon

Pour obtenir
Le grade de docteur

Formation doctorale: Traitement de signal et d'image
École doctorale: Électronique, Électrotechnique, Automatique (EEA) de Lyon

Par
Razmig Kéchichian
Ingénieur

Soutenue le 2 juillet, 2013 devant la Comission d'examen

Jury MM.

Philippe Bolon	Professeur, Polytech Annecy-Chambéry	Président
Fabrice Heitz	Professeur, Université de Strasbourg	Rapporteur
Jean-Philippe Thiran	Professeur, EPF, Lausanne	Rapporteur
Sébastien Valette	Chargé de recherche, CNRS	Co-encadrant
Michel Desvignes	Professeur, Grenoble INP	Co-directeur
Rémy Prost	Professeur, INSA de Lyon	Co-directeur

Summary

We develop generic methods for the efficient semiautomatic multiobject segmentation of high-resolution 3D medical images. We formulate segmentation as a Bayesian labeling problem defined in terms of a prior distribution, image-derived evidence and user constraints and solve it by minimizing an energy with multilabel Graph Cut algorithms which produce provably good approximate solutions and in some cases global optima.

The most widely used prior in Graph Cut segmentation is the Potts model. It implies the presence of a single boundary in the image, which is untrue in multiobject segmentation. We investigate extensions of this model to account for multiple boundaries in multiobject segmentation which allow the minimization of resulting energies by Graph Cut algorithms. Our main contribution is a structural prior model, called the vicinity prior, which captures the spatial configuration of objects. It is formulated as shortest-path pairwise constraints on a graph model of interobject adjacency relations.

Graph Cut algorithms use grid graphs which become complex on large input making optimization impractical. We reduce memory budget and optimization runtime by clustering image voxels prior to segmentation by an adaptive centroidal Voronoi tessellation which strikes a good balance between the conflicting goals of cluster compactness and object boundary adherence and help to place segmentation boundaries precisely. We formulate segmentation according to the graph of clusters such that its energy is independent of clustering resolution. This constitutes our second contribution.

Qualitative and quantitative evaluation and comparison with the Potts model on synthetic, simulated and real medical images confirm that the vicinity prior model achieves significant improvements in the correct segmentation of distinct objects having identical intensities, the accurate placement of structure boundaries and the robustness of segmentation with respect to clustering resolution. Comparative evaluation of our clustering method with competing ones confirms its benefits in terms of runtime and quality of produced partitions. Compared to voxel segmentation, the clustering step improves both overall runtime and memory footprint of the segmentation process up to an order of magnitude virtually without compromising the segmentation quality.

Structural prior models can be derived from other relationships as well, such as distance, area of common surface and spatial directionality. Graph models describing such relations are fairly easy to define at an arbitrary level of detail from simple specifications of relationships on pairs of objects, and can be generated from existing anatomical models in medical applications. Furthermore, relations can be learned from training data or several variants of a model thus capturing its variabilities.

Acknowledgements

First and foremost, I would like to thank my thesis advisors Messrs. Rémy Prost and Michel Desvignes for their guidance and trust throughout the past three years and half, and especially my supervisor and close collaborator Mr Sébastien Valette without the assistance, the meticulous attention to detail and the encouragement of whom this work would have never come to fruition. The insight and the invaluable feedback these gentlemen generously provided me with over the course of elaborating the present treatise undoubtedly shaped its scientific content and vigor, keeping it relevant to the times and current research trends. It should not go without notice that these three men were also instrumental in helping me meet and fulfill many deadlines and ridiculous administrative chores the life of a PhD candidate is never free of, often borrowing time from weekends and holiday leaves.

I extend my thanks and gratitude to the funding organization La Région Rhône-Alpes (France) and particularly to Mr Desvignes for generously and patiently allowing me the time to complete the thesis manuscript after recently recruiting me into his research group. I am looking forward to new research endeavors with him.

Finally, I address very special words of gratitude to my better half, Lilie. It is unavoidably cliché to thank beloved ones and family members; their support and inspiration can never be fully appreciated and returned. While the writer agonizes over his oeuvre, they suffer through his absence. Creation is beautiful agony, the wait is not. By nature, writing is a selfish process, and the writer only hopes to achieve salvation by the consciousness that his work would offer some inspiration and insight to readers.

Table of contents

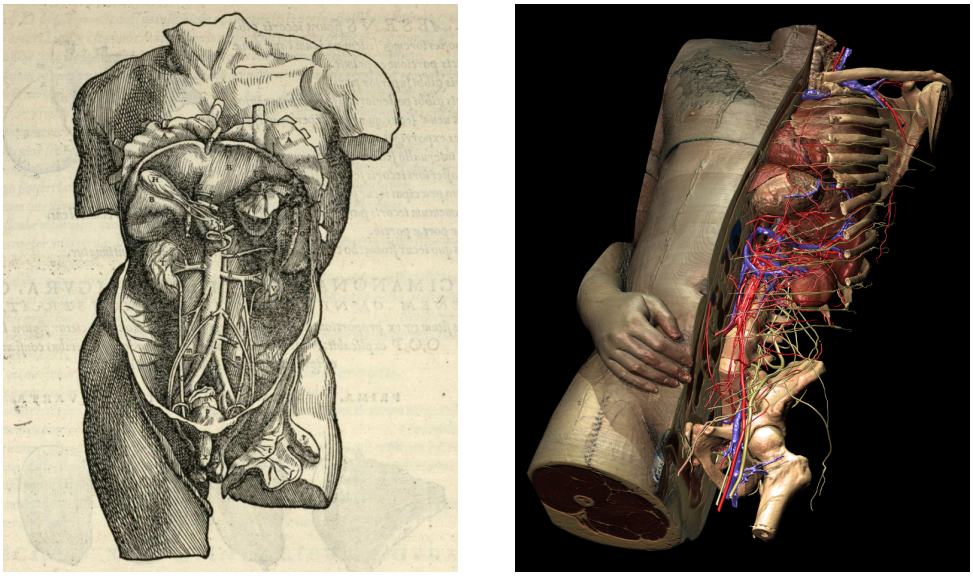
Chapter 1

Introduction

1.1 From Vesalius to Visible Human

Medicine owes much of its developments to advances in anatomy throughout the history of modern science. Although the study of anatomy begins at least as early as 1600 BC in ancient Egypt, the treatise of Vesalius [Ves43] was the first published work on anatomy to overthrow Galenic and Arabian medical doctrines which prevailed until the Renaissance. Prioritizing dissection, Vesalius established a scientific method introducing what is today called an anatomical view of the body which sees the human internal essentially as a corporeal structure filled with organs arranged in a three-dimensional space. Vesalius would not have been able to accomplish his seminal work without the advances in art during the Renaissance and the technical developments of printing press at the same time. The latter technology in particular allowed him to produce highly detailed illustrations of human anatomy accurately depicting his findings. In the 19th century, the finalization and the systematization of descriptive human anatomy developed during the previous centuries, along with growing contributions from new sources of knowledge such as histology and developmental biology, culminated in Gray's Anatomy [Gra58], an influential anatomical atlas which continues to be revised and republished to the present day in print and digital media [Sta08]. It is considered one of the most comprehensive and detailed references on the subject of human anatomy.

With the revolution of medical sciences in the 20th century, anatomical research has taken advantage of technological developments and expanding knowledge in new disciplines, such as endocrinology, evolutionary and molecular biology, to achieve a thorougher understanding of the human body's organs and structures. Importantly, medical imaging devices, such as magnetic resonance imaging (MRI) and computed tomography (CT) scanners, have enabled researchers to inspect the anatomy of living as well as of dead organisms after being limited to dissection studies on cadavers for centuries. Advances in medical imaging technologies simultaneously with the development of computing platforms and computer graphics technologies together with standardization efforts, such as DICOM [Pia11], moved many medical disciplines to



(a) (b)

Figure 1.1: Anatomical representation between the past and the present. (a) A dissectional drawing of internal organs and genitals from [Ves43], (b) a 3D dissectional view of the Voxel-Man inner organs virtual model [PHP+01] constructed from the Visible Human male image dataset [SASW96].

the digital realm, while anatomical research and education increasingly relied on digital photography and three-dimensional computer-generated representations. A major milestone at the turn of the century, perhaps of equal significance to Vesalius' contribution for its time, was the release of the Visible Human dataset by the U.S. National Library of Medicine [SASW96], which is an open-access digital image library of the adult male and female anatomy comprising high-resolution cryosectional color photographs of unprecedented detail, as well as the corresponding MR and CT images for the same subjects. This dataset has formed the basis of many biomedical and engineering applications and medical education platforms[1] which have all exploited computer and visualization technologies for the examination of anatomy in its natural three-dimensional setting. In particular, three-dimensional representations reconstructed from images in the Visible Human dataset have made it possible to view human anatomy from perspectives unavailable in dissections and thus have helped to identify and correct several inaccuracies in two-dimensional illustrations of conventional anatomical atlases [VIM04]. Today, as the macroscopic aspects of human anatomy have been largely cataloged, the research focus is on the development, the evolution, and the function of anatomical features, particularly on microscopic scales.

[1] For recent bibliography, refer to the Visible Human project web site [U.S12]

1.2 From the archetype to the subject-specific

The importance of anatomical understanding in modern medical practice cannot be stressed enough. A thorough working knowledge of gross anatomy, and sometimes of microscopic anatomy as well, is required for most medical workers; physicians, dentists, physiotherapists, nurses, paramedics, radiographers and especially surgeons, not only to support diagnosis but also in preventive care, therapy and particularly in preoperative planning, surgical intervention and postoperative care. Such clinical procedures usually involve an imaging step by the appropriate modality, depending on the tissue or structure type, which often constitutes the only means to discover a disease and proceed with diagnosis, particularly in the case of cancerous tumors. Obviously the accuracy of diagnosis and the effectiveness of surgical intervention depend crucially on the quality of the acquired image, the sufficiency of its coverage of the pathology and, more importantly, the relevance and the precision of information that a practitioner can extract from the image. Although, in general, a trained practitioner is considered of superior image analysis and pathology identification abilities, relegating some limited tasks to computers is of great practical interest; for example, the location and the isolation of a tumour from affected structures in the image for measurement and the outlining of moving structures for tracking purposes are routinely performed tasks on present-day imaging devices. Such tasks are carried out by "learned" software programs entailing, in almost all cases, structure recognition and delineation mechanisms, that is, image segmentation algorithms.

In addition to their critical importance in clinical practice, medical imaging technologies are exploited by other disciplines of medical science as well, notably in the study of anatomy and physiology. In many applications in these fields, medical images of different modalities are frequently analyzed for structure segmentation in preparing anatomical and physiological models for the diverse purposes of research, medical education and visualization. A noteworthy example is the creation of anthropomorphic phantoms for applications in health physics and nuclear medicine, such as whole-body and individual-organ radiation dosimetry for radiation protection and radiotherapy. Computational procedures of such studies usually involve Monte Carlo simulations to track radiation interactions and energy deposition in the body. Needless to say, the accuracy of such simulations largely depend on the precision and the realism of underlying anatomical representations. The construction of whole-body anatomical models from real images naturally entails a nontrivial task of identification and segmentation of, for the applications just mentioned, at least several tens of structures from a series of whole-body images, in some cases originating from different modalities to account for different anatomical structures. Unsurprisingly, the Visible Human image dataset has formed the basis of several such digital phantoms, notably the voxel-based VIP-Man [XCB00] and Voxel-Man [PHP+01] phantoms and the NURBS-based Cardiac-Torso (NCAT) phantom [Seg01]. All of these phantoms were constructed from the cryosectional color photographs in the dataset by trained anatomists mostly by a painstaking process of manual contouring and image volume editing complemented by elemen-

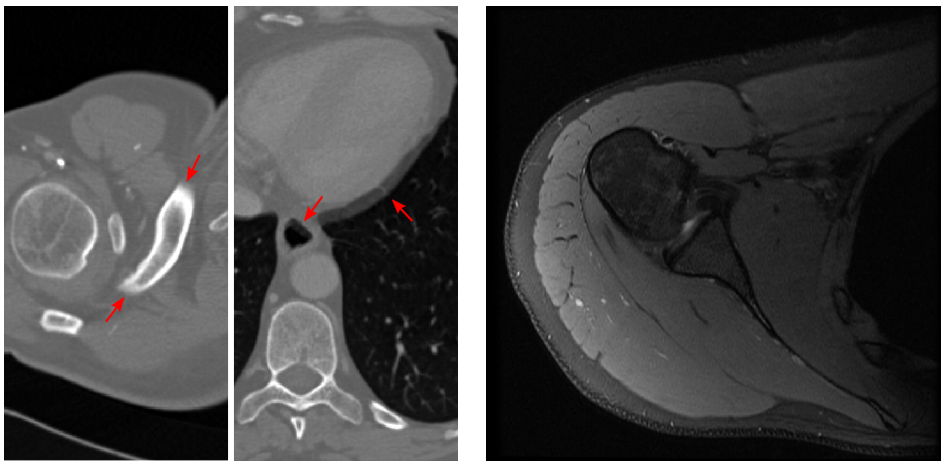
tary computer-assisted techniques such as thresholding and color-space classification. However, unlike the dataset from which they were created, none of these phantoms are available to the scientific community on an open-access basis.

While in the past cadavers were the definitive source of anatomical information, it seems ironic today that with the availability of open-access digital anatomical datasets, such as the Visible Human image library, and the effective means to process and visualize them, that is, powerful computing platforms and computer graphics technologies, open-access three-dimensional anatomical atlases and models are lacking. The sheer volume and anatomical coverage of image data to process, coupled with difficulties associated with the analysis and the segmentation of such images by automatic unattended methods prompts researchers and engineers to resort to manual methods in the construction of three-dimensional whole-body models which require significant investments in time and labor making it often infeasible to give away its outcome.

Clinical practice today, especially whole-body CT and MRI scanning, often generates large numbers of high-resolution images, which makes tasks of efficient data access, transfer, analysis and visualization challenging, especially in today's distributed computing environments which have seen growing use of handheld terminals for interactive data access and visualization of anatomy. With increasing availability and sophistication of industrial-grade medical imaging technologies, unprecedented amounts of patient-specific anatomical data is now at the disposal of medical practitioners. Traditionally single-object or pathology oriented, recent image processing methods, discussed later in this chapter, have made the simultaneous analysis and segmentation of multiple anatomical structures increasingly possible. Moreover, remote processing and visualization technologies for large datasets, such as [JJAM11], have matured and opened the door for a number of popular applications, notably the semantic navigation of full-body human anatomy [BCK+11]. Thus, the creation of patient-specific anatomical models, at least systemic or disease-specific, is becoming of great interest.

1.3 Why is it difficult to segment medical images?

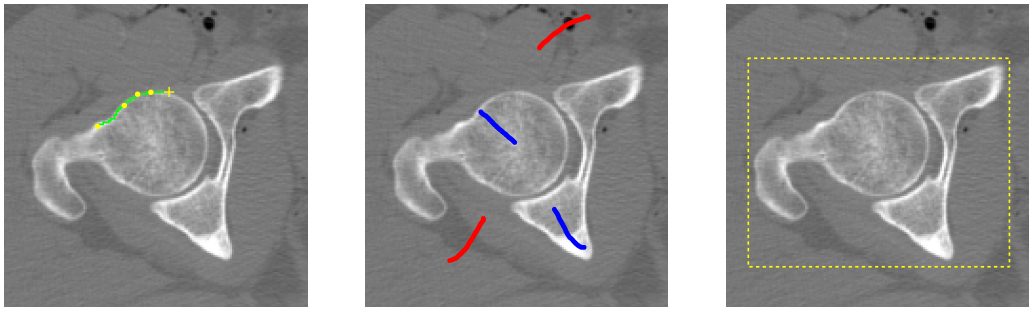
Despite more than 40 years of progress, image segmentation remains one of the major challenges and among most active research topics in image processing and computer vision [FP11]. As we have seen, segmentation of objects in images is a prerequisite to their effective analysis, manipulation and visualization. In medical applications in particular, segmentation of anatomical structures in medical images is the back-bone of many clinical procedures from diagnosis to surgical planning and intervention, and is crucial to other fields of medical science involving the study and the representation of human anatomy. The segmentation of high-resolution medical images for the purpose of creating application-specific reduced representations of targeted structures, such as surface and volume meshes, is also interesting from the viewpoint of efficient data storage, dissemination and three-dimensional visualization.



(a) (b)

Figure 1.2: Commonly encountered artifacts in CT and MR images. (a) (left) Partial volume effect on the clavicle and (right) motion artifacts on lung and esophagus boundaries in a thoracic-abdominal CT scan. (b) Intensity inhomogeneity field affecting muscular and adipose tissues in an MR image of the shoulder. Image courtesy of David Sarrut of Léon Berard cancer research center and Jean-Baptiste Pialat of Édouard Herriot hospital, Lyon, France.

CT and MR images have intrinsic characteristics that render its analysis and segmentation, especially in automatic unattended mode, particularly challenging [BSJB11]. In theory, they are regarded as piecewise-constant intensity maps over a number of tissue classes. In practice, however, the piecewise-constance property is degraded considerably by a number of factors. Even with significant technological advances, current image acquisition systems often yield low signal-to-noise ratio data, consequently produced images are commonly degraded by various noise sources. The limited resolution of imaging systems often results in the dual effects of partial volume and spillover leading to structure volume under- and overestimation respectively. These, in addition to other image artifacts due, for example, to patient motion or limited acquisition time, reduce the prominence of intensity edges in images creating diffuse or poorly defined ones which make it difficult to identify true structure boundaries. MR images may, in addition, suffer from spatial distortion of structure intensity due to magnetic field inhomogeneity and gradient field nonlinearity. Good contrast between tissue class intensities, which greatly facilitates segmentation, is usually achieved by using contrast agents depending on the imaging modality, in the lack of which contrast enhancement post-processing methods are resorted to, which often give less satisfactory results. However, regardless of the imaging modality and related artifacts, many anatomically and functionally distinct structures, especially those corresponding to soft tissues, have similar intensity levels in images and, furthermore, blend into surrounding tissues which have intensities close to their own. It is impossible for an automatic process to identify and segment such structures on the basis of intensity information only, hence most advanced segmentation methods exploit some form of prior information on structure location, spatial extent or interrelations to achieve greater robustness and precision.



(a) (b) (c)

Figure 1.3: Three commonly encountered user constraints applied to the semiautomatic segmentation of femoral head and hip bones. (a) Object-boundary constraints with part of the boundary traced by the algorithm, (b) object-region constraints for object (blue) and background (red), (c) image-region constraints.

1.4 The choice of a segmentation approach

Segmentation, as the reader can tell, consists of the two closely-interrelated tasks of structure recognition and delineation. Recognition involves identifying the location of the targeted object and distinguishing it from others that are present in the image, and delineation involves designating its precise spatial extent. According to the degree of automation in these two tasks, existing segmentation methods cover a wide spectrum of approaches from completely manual tracing or painting of object boundaries or regions to fully automatic recognition and delineation of objects. Automatic methods have had considerable success in many well-defined application contexts, usually coinciding with fixed image acquisition protocols and little variation in object pose, shape and location in the image. Due to the availability of image datasets acquired in such conditions by routine clinical or laboratory procedures, most automatic segmentation approaches rely on image and object features learned from presegmented datasets. This makes them prone to be bound to the characteristics of training sets and therefore to have difficulty in accounting for previously unencountered but possible large variations in object pose, location and shape, typically arising in pathological cases, and the photometric features of images acquired by different protocols in different conditions.

Semiautomatic segmentation methods are usually resorted to in order to lighten the burden of manual delineation and as an alternative to fully-automatic methods when the latter are inapplicable, for example due to the singularity of the studied case or the lack of training sets. Semiautomatic methods achieve a balance between manual and automatic methods exploiting the synergy between a trained expert's superior object recognition abilities and a computer's faster delineation capacities. Many methods relying on different interaction paradigms have been devised to allow a user to initialize, steer a segmentation process and refine its result with a reasonable amount of input. User interaction and feedback provide clues on what he intends to segment which can be exploited by a segmentation algorithm in a number of ways, such as to constrain segmentation boundary placement or the assignment of a pixel to some object, and to collect intensity statistics of objects and derive prior information on their interrelations.

User interaction is usually carried out through a traditional human-computer interface using a pointing device to mark certain image pixels in order to indicate an object or a boundary type. Often called "seeds", such pixel markings are consequently interpreted by the segmentation algorithm as delineation guides or pixel assignment constraints. Commonly encountered user-specified constraints fall in three categories: object boundary, object region and image region constraints. A popular object-boundary constraint formulation for semiautomatic segmentation was introduced in 2D by [MB98, FUS+98] and generalized to surface constraints in 3D by [FU00]. In these approaches, the user roughly traces the boundary of each object indicating a number of seed points through which each segmentation boundary should pass. The segmentation algorithm is tightly coupled with user gesture and finds the best boundary as the minimum-cost path between the current and the last specified seed points on the image graph according to some image gradient-based local cost. If a computed path deviates from the desired one, due to a weak intensity edge for example, the user specifies additional seeds. The intricacy of this method is obvious; the user is required to place seeds accurately on the object boundary. By contrast, approaches relying on object-region constraints require that the user places seeds anywhere inside targeted objects. It is the segmentation algorithm's duty to ensure that marked pixels are part of corresponding objects in the result. Assuming that objects consist of fairly homogeneous regions, such constraints are quite robust to placement precision and, compared to object-boundary seeds, their specification requires much less time and effort and can be carried out in 3D in a straightforward fashion by placing seeds in 2D slices of the image volume corresponding to a chosen cross-sectional view. The interpretation and the use of such seeds often depends on the algorithm. For example, [VK05] define seeds as the starting point of their cellular automata-based region growing algorithm. In [GFL04], given a number of user-specified seeds with label types corresponding to targeted objects, the algorithm uses an analytic method to determine, for every pixel in the image, the probability that a random walker starting its walk at that pixel first reaches a seed of a given label type. The pixel is then assigned the label with the greatest probability. Combinatorial Graph Cut methods [BF06], which describe a segmentation in terms of optimal regional and boundary properties of objects, interpret object-region seeds as hard constraints for pixels that should be part of the corresponding object in the result, also using them to derive intensity statistics of objects in addition to allowing the user to edit and refine an initial segmentation efficiently by corrective seeds. Image region-based constraints entail interaction methods that are even less stringent than object-region based ones and often do not involve the marking of any particular pixel belonging to some object. For example, [RKB04] only require that the user draws a bounding box around the targeted object in the image separating it from the background. The algorithm then refines background and foreground statistics iteratively as it converges to the optimal separating boundary. Figure 1.3 illustrates the three discussed interaction paradigms. Recent gesture-based multitouch interfaces offer new user interaction possibilities the use of which in simultaneous semiautomatic segmentation and visualization of anatomical data is undoubtedly an interesting research venue in its own right.

A number of successful variational and combinatorial segmentation methods have matured in the last decade, notably level-set based active contour [OF03] and maximum-flow based Graph Cut [BV06] methods. While the first group of approaches formulate segmentation as a continuous optimization of contour or surface functionals and the latter as a discrete optimization of cost functions on the image graph, both groups of methods rely on implicit nonparametric representations of object boundaries. Level-set methods define the implicit representation of a contour or a surface as the zero-level set of a higher dimensional function and describe its evolution towards the desired object boundary by a partial differential equation. The contour movement is driven by external image-based and internal contour-based speed functions, which, respectively evolve the contour towards image features, such as intensity edges, and smooth its boundary. In Graph Cut segmentation, targeted objects are represented by labels and a discrete labeling cost function of variables corresponding to image pixels is formulated in terms of label likelihood and spatial regularization criteria. The cost function is then represented by a flow network the minimum cut of which, calculated by a maximum flow algorithm, finds the optimal segmentation boundary. As research in both fields has progressed, state-of-art Level-set and Graph Cut segmentation methods have come to incorporate different object appearance, boundary and shape based cues and prior information [CRD07] (see Sections 2.2.1, 2.2.2 and 3.2 for corresponding Graph Cut literature) as well as topological constraints [HXP03, BV06], while Level-set methods too were shown to lend themselves to user-steered segmentation [YPH+06].

Simultaneous multiobject segmentation approaches are superior to those formulated as a sequence mono-object segmentations because they allow joint specifications of object properties and interrelations and entail simultaneous optimization with respect to these criteria. As such, they raise questions neither on the best segmentation sequence to follow nor on how to avoid the propagation of errors on individual segmentations [FAB12]. Multiobject generalizations of the basic Level-set method usually present problem or application-dependent tradeoffs in either representing every object independently with its own internal and external speeds and relying on soft coupling terms to avoid overlaps and gaps at the expense of high memory requirements as the number of objects increases, or using so-called multiphase representations with a number of level-sets inferior to the number of objects which reduce storage requirements and ensure the absence of overlaps and gaps at the expense of reduced flexibility in the specification of internal and external speeds which often arise in practice [BPB13]. By contrast, the basic Graph Cut segmentation method can be generalized to multiobject settings without modifying the form of the cost function and without representation overhead, although the complexity of the underlying graph structure is directly proportional to image size. In this case, the multilabel optimization of the cost function is expressed as a tightly coupled sequence of maximum-flow optimizations [BVZ01].

Although the continuous level-set representation and the associated evolution strategies have a number of attractive properties, such as subpixel accuracy and natural handling of topological changes, the solution of underlying partial differential equations has to be addressed carefully in order to ensure numerical stability which often limits

contour evolution speeds making Level-set methods rather slow in practice, especially in multiobject formulations [TT11]. By comparison, maximum flow algorithms underlying Graph Cut methods are deterministic polynomial-time algorithms which are numerically stable. For a particular class of cost functions covering a wide range of likelihood and prior formulations frequently arising in segmentation applications [KZ04], Graph Cut methods produce provably good approximate solutions in multiobject and global optima in mono-object segmentation. The relative ease and the efficiency of multiobject formulation along with the speed and attractive numerical properties of associated optimization algorithms thus justify our choice to rely on Graph Cut methods as the foundation for our semiautomatic multiobject segmentation approach.

1.5 The mission statement and the structure of dissertation

As we have seen, inherent limitations of quality and information content in medical images, their dependence of acquisition protocols and conditions coupled with tissue variability across patient populations make designing automatic general-purpose segmentation algorithms difficult, if not impossible. Therefore, our aim is to develop a general Graph Cut-based semiautomatic multiobject segmentation method, principally for medical images, that can be adapted to a particular application with minimal effort according to any or all of the following criteria and tradeoffs that exist between them: the number of objects to segment, the required anatomical level of detail, interrelations between objects to be respected, the accuracy of object and boundary retrieval, available computing resources of runtime and memory budget. The projected use is in routine applications involving low-resolution 2D to high-resolution 3D images on moderate to fairly hard multiobject segmentation tasks of up to few dozens of objects at a time. With the goal of minimizing user involvement and thus making the method viable in practical situations, we choose object-region style user interaction limited to initialization, allowing postsegmentation refinement if necessary. With a trained expert in mind at the human-computer interface, user input will also be relied upon as the principal source of information to collect intensity statistics of targeted objects as well as to constrain the assignment of user-marked pixels to some object.

Among segmentation criteria to adapt to a given application context, we address, in particular, the introduction of a fairly general graph-based prior model capturing spatial relationships between objects that, on one hand, is easy to specify for different relationships such as structure adjacency and distance, and on the other hand allows efficient optimization of resulting cost functions with multilabel Graph Cut algorithms. Targeting multiobject segmentation applications on high-resolution 3D images, we also address runtime and memory efficiency of Graph Cut segmentation with an image-adaptive clustering method paying particular attention to the robustness of segmentation and the invariance of its theoretical properties to the reduced data representation. The following list opens a window on the work's landscape summarizing its major contributions on a high level.

Chapter 2: Theoretical background, presents and discusses the theoretical foundations of our work, in particular the theory of Bayesian labeling and Markov random field modeling in image segmentation, concepts of flow networks, maximum flow techniques and Graph Cut algorithms in binary and multilabel optimization.

Chapter 3: A structural prior model for Graph Cut segmentation, investigates extensions of one of the most widely used prior models in Graph Cut segmentation, the Potts model, to account for multiple boundary interactions in multiobject segmentation allowing to minimize resulting multilabel cost functions by Graph Cut algorithms. Our main contribution is a structural prior model, called the vicinity prior, which captures the spatial configuration of objects. It is defined as shortest-path constraints on a graph model of interobject adjacency relations [KVDP13].

Chapter 4: Reducing Graph Cut complexity, addresses runtime and storage efficiency of Graph Cut segmentation on high-resolution input and proposes to improve them both by clustering image voxels prior to segmentation by an image-adaptive centroidal Voronoi tessellation and reformulating Graph Cut segmentation according to the graph of clusters such that it is independent of the clustering resolution [KVDP11]. The chapter also gives the results of a comparative evaluation of the clustering method with competing ones confirming its benefits in terms of runtime and quality of produced image partitions.

Chapter 5: Evaluations and applications, presents and discusses results of qualitative and quantitative evaluations of the proposed segmentation approach of multiobject Graph Cut segmentation on synthetic, simulated and real medical images confirming its advantages over its Potts prior-based counterpart, also showing that, compared to the application of segmentation on voxels, the clustering step improves both overall runtime and memory footprint of the segmentation process up to an order of magnitude without compromising the segmentation quality.

Chapter 6: Conclusions and perspectives, concludes the work outlining its major results and contributions and discusses future venues of research.

1.6 Remarks on notation

A few remarks on general conventions we follow in spelling and notation are in order. We capitalize proper nouns when they refer to a particular method, an algorithm or a problem, designating them with small letters otherwise, for example, when speaking of the Maximum flow problem and maximum flow algorithms. In some contexts, we use the terms voxel and pixel interchangeably when such use does not affect the precision and the clarity of presentation. To simplify asymptotic notation of algorithm complexity, instead of $O(|N|)$, we write $O(N)$, where N is some set the cardinality of which is proportional to the problem size. As it is customary in many treatments of discrete mathematics, we use braces to denote an unordered pair of indices or mathematical objects i, j as $\{i, j\}$ using parentheses for ordered pairs (i, j).

Chapter 2

Theoretical background

2.1 Introduction

In this chapter, we present and discuss the theoretical foundations of our work for a fuller understanding and appreciation of our contributions. The presentation of methods and underlying theories is concise yet includes sufficient detail to enable unfamiliar readers to follow subsequent chapters without having to consult external references.

In Section 2.2 we present the theory of Bayesian labeling and Markov random field modeling in image segmentation discussing underlying probability distributions and how energies describing these models can be solved. Section 2.3 presents the theory of flow networks and discusses the algorithms that solve the Maximum flow problem. Maximum flow techniques constitute the foundation of our energy optimization method of choice, Graph Cut algorithms, which are discussed in Section 2.4.

Throughout the presentation of methods, related literature is discussed in the appropriate context with state-of-art closely related to our contributions relegated to later chapters. The choice of a method over another for our problem is discussed and argued.

2.2 Image segmentation as Bayesian labeling

We shall formulate segmentation as a labeling problem, defined as the assignment of a label from a set of m labels $L = \{l_1, \ldots, l_m\}$, representing objects to be segmented, to each of the variables in a set of n variables, corresponding to image primitives indexed by $\mathcal{S}$. In this chapter as well as in the following one, we shall assume that these primitives correspond to image voxels. Later in Chapter 4 we shall redefine them to correspond to clusters of image voxels. Assume that each variable $i \in \mathcal{S}$ is associated with a node in an undirected graph on image voxels $\mathcal{G} = \langle \mathcal{S}, \mathcal{E} \rangle$, where pairs of adjacent image voxels define the set of edges $\mathcal{E} = \left\{ \{i, j\} \mid i, j \in \mathcal{S} \right\}$. Consequently, the neighborhood of a node i is defined as $\mathcal{N}_i = \left\{ j \mid j \in \mathcal{S}, \exists \{i, j\} \in \mathcal{E} \right\}$. Note that $i \notin \mathcal{N}_i$ because there is no edge from a node to itself. Although it can be arbitrarily complex, we assume that only immediate

neighbors of a node i constitute its neighborhood $\mathcal{N}_i$. Thus, the neighborhood of a variable is defined according to the directions parallel to the planes of a 3D image to get a 6-connected (4-connected in 2D) system. When voxels on diagonals are also taken into account, the neighborhood becomes a 26-connected (8-connected in 2D) system.

An assignment of labels to all variables is called a configuration, and is denoted by $\ell \in \mathcal{L}$, where $\mathcal{L} = L^n$ is the space of all possible configurations. The assignment of a label to a single variable $i \in \mathcal{S}$ is denoted by ℓ_i. The space of admissible configurations, which corresponds to possible segmentations, can be identical to $\mathcal{L}$ or, if user-specified constraints are taken into account, to a subset of $\mathcal{L}$.

A label configuration $\ell \in \mathcal{L}$ can also be expressed in probabilistic terms. Let $X = \{X_1, \ldots, X_n\}$ be a set of random variables on $\mathcal{S}$ taking its values in L. A configuration ℓ can be defined as the joint event $X_1 = \ell_1, \ldots, X_n = \ell_n$, abbreviated as $X = \ell$. Defining a joint probability distribution $\Pr(X = \ell)$ in the space of configurations $\mathcal{L}$, however, is rather difficult. Markov random field (MRF) theory [Li09] is usually resorted to in order to define such joint probability distributions from interactions between neighboring variables which can be specified conveniently and meaningfully.

Given a joint probability distribution $\Pr(X)$, we recall that the set of random variables X is an MRF with respect to the graph $\mathcal{G}$ if it satisfies the following condition:

$$\Pr(X_i = \ell_i \mid X_j = \ell_j, j \in \mathcal{S}, j \neq i) = \Pr(X_i = \ell_i \mid X_j = \ell_j, j \in \mathcal{N}_i). \tag{2.1}$$

This is called the Markov local property. It states that the probability of a label assignment to a variable is conditionally dependent on assignments to its neighbors only; in other words, the state of a variable depends only upon the state of its neighbors.

It still is not a trivial task to specify the set of conditional probabilities for an arbitrary joint probability distribution to satisfy the Markov local property (2.1). Fortunately, the Hammersley-Clifford theorem [Bes74] provides a solution. According to this theorem, a positive joint probability distribution of label configurations $\Pr(X) > 0$ satisfies the Markov local property with respect to $\mathcal{G}$ if and only if it is a Gibbs distribution with respect to $\mathcal{G}$, that is, if it can be factorized over the cliques[1] of $\mathcal{G}$.

A Gibbs distribution of configurations ℓ with respect to the graph $\mathcal{G}$ is given by:

$$\Pr(\ell) = Z^{-1} \times e^{-\frac{1}{t} U(\ell)} \tag{2.2}$$

where Z is a normalizing constant, t is a hyperparameter called the temperature and $U(\ell)$ is an energy function specified as a sum of functions of label configurations, called clique potentials, defined on up to an arbitrary order of graph cliques $\mathcal{C} = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \ldots$:

$$U(\ell) = \sum_{c \in \mathcal{C}} V_c(\ell) = \sum_{\{i\} \in \mathcal{C}_1} V_1(\ell_i) + \sum_{\{i,j\} \in \mathcal{C}_2} V_2(\ell_i, \ell_j) + \ldots . \tag{2.3}$$

Thus, the joint probability distribution $\Pr(X = \ell)$ can be conveniently defined by specifying appropriate clique potentials that encode prior domain knowledge about plausible configurations through local interactions between labels assigned to neighboring sites. Hence, $\Pr(X)$ is called a prior probability distribution.

[1] A clique in an undirected graph is a subset of vertices, each pair of which is connected by an edge. The order of a clique is the number of its vertices.

Since we concern ourselves with finding the "best" segmentation with respect to some user-provided evidence, we are interested in the probability distribution of configurations $\Pr(X = \ell \mid O)$ conditional on some, usually image-derived, observation O. This is called a posterior probability distribution. The observation constitutes a combination of spatially distinct records at each variable, for example, observed image intensity values. Evidence from observation is defined as the probability distribution $\Pr(O \mid X = \ell)$ and is called a likelihood distribution. Assuming that observations are local and spatially independent, it is reasonable to define likelihood probabilities at each variable $\Pr(O_i \mid X_i = \ell_i)$ to be conditionally independent. Thus:

$$\Pr(O \mid X = \ell) = \prod_{i \in \mathcal{S}} \Pr(O_i \mid X_i = \ell_i). \tag{2.4}$$

Note that the posterior distribution $\Pr(X = \ell \mid O)$ gives the probability of all possible configurations. The "best" configuration with respect to user-provided evidence can be considered the one with the highest probability given the observation, which corresponds to the highest mode of the posterior probability distribution:

$$\ell^* = \arg\max_{\ell \in \mathcal{L}} \Pr(X = \ell \mid O). \tag{2.5}$$

ℓ^* is called the maximum a posteriori probability (MAP) solution.

Recall that the posterior probability can be stated, according to the Bayes rule, in terms of prior, likelihood and observation probabilities as:

$$\Pr(X = \ell \mid O) = \Pr(O \mid X = \ell) \Pr(X = \ell) / \Pr(O).$$

Since $\Pr(O)$ does not depend on ℓ and therefore does not affect the MAP solution, the posterior probability is proportional to the joint likelihood and prior probabilities:

$$\Pr(X = \ell \mid O) \propto \Pr(O \mid X = \ell) \Pr(X = \ell). \tag{2.6}$$

Substituting the right-hand side of (2.6) from (2.2) and (2.4), taking the logarithm of both sides, ignoring the constant term $\ln z^{-1}$ and multiplying both sides by $-t$ we get:

$$-t \ln \Pr(X = \ell \mid O) \;\propto\; t \sum_{i \in \mathcal{S}} -\ln \Pr(O_i \mid X_i = \ell_i) + \sum_{c \in \mathcal{C}} V_c(\ell). \tag{2.7}$$

Given likelihood and posterior probabilities, the negative logarithm of the posterior can be regarded as an energy function of label configurations, thus we write (2.7) as:

$$E(\ell) \;=\; t \sum_{i \in \mathcal{S}} -\ln \Pr(O_i \mid X_i = \ell_i) + \sum_{c \in \mathcal{C}} V_c(\ell). \tag{2.8}$$

Recall that the negative logarithm is a monotonically decreasing function. This means that the negative logarithm of the posterior $-\ln \Pr(X \mid O)$ achieves its minimum at the maximum posterior probability. Thus, the MAP solution can be equivalently found by:

$$\ell^* = \arg\min_{\ell \in \mathcal{L}} E(\ell). \tag{2.9}$$

Clearly, the maximum posterior probability configuration (2.5) is identical to the minimum posterior energy configuration (2.9).

Retaining only second-order clique potentials from the prior energy sum, we rewrite (2.8) as follows:

$$E(\ell) = t \sum_{i \in \mathcal{S}} D_i(\ell_i) + \frac{1}{2} \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{N}_i} V_{i,j}(\ell_i, \ell_j). \tag{2.10}$$

This is the actual energy function we minimize to obtain the best segmentation with respect to user-supplied evidence and constraints, and some problem domain-dependent prior. Note that since either (i, j) or (j, i) is in the set of second-order binary cliques $\mathcal{C}_2$, either $V_{i,j}(\ell_i, \ell_j)$ or $V_{j,i}(\ell_j, \ell_i)$ should be in the prior energy sum (2.3). We consider that $V_{i,j}(\ell_i, \ell_j) = V_{j,i}(\ell_j, \ell_i)$, and in order to use the double-sum notation over variables and their neighborhoods in (2.10), we multiply the double-sum by $\frac{1}{2}$.

2.2.1 Likelihood-based data cost and user constraints

The unary energy term[2] $D_i(\ell_i)$ in (2.10), which is also called a data term reflecting the fact that it is often derived from observed data, measures the cost of assigning a label to the variable $i \in \mathcal{S}$ so that unlikely label assignments incur higher penalties. This property stems from its definition as a negative log-likelihood function (2.8), since the higher a likelihood probability is, the lower its negative logarithm would be.

In many MAP-MRF formulations, the observation is taken to be the spatially distinct records of image intensity values at each variable, thus the likelihood probability for a variable $i \in \mathcal{S}$ given the label assignment ℓ_i becomes $\Pr(O_i = I_i \mid X_i = \ell_i)$.

A variety of approaches are followed to estimate such conditional probability distributions. For example, they can be represented as: object-specific intensity histograms [BF06], kernel density estimates [LPPS12], Gaussian distributions [GTC11] and Gaussian mixture models (GMM) with hard [RKB04] and soft membership probabilities [ZBS01]. The latter approach sometimes follows an iterative Expectation-Maximization style two-step estimation [DLR77] whereby, given an initial estimate of GMM parameters, the MAP solution is used in the expectation step to calculate membership probabilities, then GMM parameters are calculated in the maximization step in a standard fashion and the process is repeated until some convergence criterion is met. Such iterative schemes need to be carefully designed in order to ensure numerical stability and algorithm convergence [RRBCK11]. In more recent conditional random field (CRF) image segmentation approaches [GRC+08, FVS09, LLB+11], observations often take the form of feature vectors over a set of different image features of intensity, texture and shape statistics [Low04]. For each label class, feature vectors are constructed from ground-truth data and fed to classifiers, the output score of which, normalized as necessary, is used to estimate likelihood probabilities.

[2] Likelihood-based unary terms in (2.10) should not be confused with prior-based unary energy formulations frequently encountered in the MAP-MRF literature which stem from the first-order clique potentials of the prior energy sum (2.3). In the context of segmentation, such terms are derived, for example, from a probabilistic atlas of tissue location [STAG06], a shape similarity measure [VM08] or a distance transform of a shape template [KRBT08].

In all results reported in our work, conditional probability distributions of intensity $\Pr(I \mid L = l)$ for every label are estimated as Gauss-smoothed and normalized intensity histograms from user-supplied "seeds", described in Section 1.4, for every structure to be segmented. We could have employed a more elaborate approach, such as kernel density estimation [Par62], to estimate likelihood densities. However such schemes require a choice of parameters, a good setting of which often entails a separate selection mechanism with no guarantees of yielding substantially better estimates in practice. We therefore have opted for a straightforward nonparametric approach.

Unary terms of the energy (2.10) can also encode user constraints on acceptable configurations. Recall that user-supplied seed labels indicate image voxels which are known to be part of some object. In addition to their use in deriving likelihood densities, in similar spirit to [BF06], we take advantage of such seeds in reducing the solution space of the labeling problem by prohibitively increasing the cost of invalid configurations. In order to ensure the attribution of the respective label type to seed-marked voxels, the data term should create a maximum bias for the seed's label and, at the same time, incur the maximum penalty for the attribution of all other labels. Denoting the set of all seeds by $\mathcal{H} = \{h_i^l \mid i \in \mathcal{S}, l \in L\}$, we define the data term as follows:

$$D_i(\ell_i) = \begin{cases} 0, & \exists h_i^k \in \mathcal{H}, k = \ell_i \quad (2.11a) \\ \infty, & \exists h_i^k \in \mathcal{H}, k \neq \ell_i \quad (2.11b) \\ -\ln \Pr(I_v \mid \ell_i), & \nexists h_i^k \in \mathcal{H}, \forall k \in L \quad (2.11c) \end{cases}$$

where h_i^k is a seed of label type $k \in L$ attributed to voxel $i \in \mathcal{S}$ and I_v is the gray-level of voxel v. Note that (2.11) is a continuous function where (2.11a) and (2.11b) correspond to negative log-likelihood values at maximum and zero probability respectively. In practice, we represent the infinite cost in (2.11b) by a large constant several orders of magnitude (10^9–10^{12}) greater than the value of (2.11c) corresponding to the smallest estimated likelihood probability, so that for practical settings of temperature $t > 0$, (2.11a) and (2.11b) amount to imposing hard constraints for acceptable configurations. By comparison, in order to ensure the invariance of user constraints with respect to temperature settings, [BF06] avoid multiplying (2.11b) by t in (2.10) and define a constant K representing its cost such that prior terms do not accidentally impose a labeling decision on corresponding variables. This, however, does not rule out the possibility of (2.11c) taking a value greater than K for a sufficiently small likelihood probability. In our formulation, as t greatly decreases, temperature-attenuated terms (2.11b) approach the order of magnitude of prior terms until the sum of data terms in (2.10) vanishes at $t = 0$, even though such a temperature setting is hardly interesting in practice.

In subsequent refinements of an initial segmentation, we allow the user to introduce additional "corrective" seeds to constrain the label preference of incorrectly segmented image regions. In order not to disrupt initial appearance statistics, we do not take such seeds into account for likelihood estimation.

2.2.2 Prior models for segmentation

Recall that the binary terms $V_{i,j}(\ell_i, \ell_j)$ in (2.10) encode prior information on pairwise interactions between labels assigned to pairs of neighboring sites. They encourage the spatial consistency of labeling with respect to a problem-dependent reference model, hence they are sometimes called spatial regularization terms.

Many classes of pairwise interaction models have been proposed in the MRF modeling literature for different image processing and computer vision problems [Li09]. For image segmentation, the pairwise interaction has to encourage piecewise-constant labeling, which is the partitioning of the image into several regions where voxels within individual regions are identically labeled. Furthermore, it has to be discontinuity preserving, meaning that, on one hand, it should attribute identical labels only to image regions which do not contain intensity edges, and, on the other hand, it should not excessively penalize sharp label transitions. Convex functions, which have a number of advantageous properties for mathematical optimization [BV04], unfortunately are not discontinuity-preserving because they do not have a bound on the largest possible penalty and potentially overpenalize large label differences. Optimizing an energy function with a convex pairwise potential would therefore lead to oversmoothing at object boundaries where abrupt transitions occur [BVZ01], hence the use of convex priors in segmentation is limited to special contexts, such as the oversegmentation algorithm in [IG98b]. Smooth convex priors however are desirable in some vision problems such as image restoration [BK04] where labels correspond to image gray-levels and stereo matching [IG98a] where labels represent disparities.

The earliest discontinuity-preserving piecewise-constant prior model is the Ising model in binary labeling where $|L| = 2$, usually $L = \{0, 1\}$. It has first been introduced in statistical mechanics as a model of ferromagnetism and later used in image processing and computer vision by analogy between images and statistical mechanics systems [GG84]. Given a pair of variables $i, j \in \mathcal{S}$ where $j \in \mathcal{N}_i$, the Ising model is defined as:

$$V_{i,j}(\ell_i, \ell_j) = 1 - \delta(\ell_i, \ell_j) \tag{2.12}$$

where δ is the Kronecker delta function. The identically-defined generalization of the Ising model for multilabeling problems where $|L| > 2$ is called the Potts model. We can readily see that both the basic model and its generalization incur a single level of penalization corresponding to the attribution of different labels to pairs of neighboring variables and incur no penalty for the identical labeling of such pairs. This penalization scheme, in fact, stems from the very nature of binary labeling where only a single type of boundary exists; the one between image regions labeled by either label.

Although the Potts prior is the most widely used spatial regularization model in MAP-MRF formulations of image segmentation and continues to enjoy popularity in recent CRF-based approaches [FVS09, SWRC09], many problem-specific pairwise models can be encountered in the literature, especially in the context of multiobject segmentation. For example, an ad-hoc extension of the Potts prior model to two penalization levels with respect to a label class cooccurrence model can be found in [CCB+05]. A shape template's distance transform has been used by [FZ05] to bias the placement

of segmentation boundary near the shape's contour. Spatial configuration prior is also quite common in the context of multiobject segmentation, where inter-object geometric and spatial relationships are often defined as pairwise relations. Such use has appeared in the context of part-based object detection [WS06] and scene layout estimation and segmentation where spatial relationships are either predefined on a limited number of classes [LVS10] or learned from annotated training data [LLB+11].

In Chapter 3 we propose and investigate a principled approach to extend the basic discontinuity-preserving piecewise-constant prior as defined by the Potts model to account for multiple types of boundaries in multiobject segmentation, incurring multiple levels of penalization, furthermore allowing the application of a particular energy minimization algorithm which converges rapidly to a provably good local minimum.

2.2.3 Solving posterior energies

Having addressed common prior models and likelihood estimation approaches for MAP-MRF image segmentation, we now turn our attention to the minimization of the energy function (2.10). In the following, we shall discuss several widely used energy minimization methods without delving into the details of algorithms. Our method of choice is presented in detail in subsequent sections of the current chapter.

Since many early vision problems are naturally expressed as pixel-labeling problems in terms of a likelihood probability and a prior distribution, energy functions of the form (2.10) frequently arise in practice. However, the minimization of such, often nonconvex, energies in high-dimensional spaces is not a trivial task. In fact, [BVZ01] show that the global minimization of (2.10) is NP-hard when $|L| > 2$ even with the simple discontinuity-preserving Potts prior model, since it reduces to the multiway cut problem which is known to be NP-complete [DJP+92]. This unfortunately implies that many similar energy formulations are intractable. Therefore, research has focused on efficient approximation algorithms which produce provably good local solutions. Having a bound on the local minimum is important, because when it is arbitrarily far from the global minimum, it would be difficult to determine the cause of the algorithm's unsatisfactory results, which could be due either to a bad energy formulation or a poorly performing minimizer.

Early energy minimization algorithms use an elementary search space exploration strategy where only a single pixel can change its label at a time. This is sometimes called a standard move. Starting with an initial estimate of the labeling, for each pixel, the algorithm chooses the label that gives the largest energy decrease either in a deterministic greedy fashion, like the Iterative Conditional Modes (ICM) algorithm [Bes86], or according to probabilistic metaheuristics, like the Simulated Annealing (SA) algorithm [KGV83], which both repeat the selection process for all pixels until convergence to a local minimum is achieved. The SA algorithm in particular has enjoyed popularity because of its implementation ease and ability to optimize an arbitrary energy. Theoretically, it would eventually find the global minimum if run for an infinite amount of time. In practice however, minimizing an arbitrary energy in finite time requires an

exponential cooling schedule to ensure convergence to a good local minimum [Cat92], which means that the algorithm can be rather slow to terminate.

ICM and SA remained in mainstream use in vision until late 1990's when energy minimization approaches had a renaissance with the introduction of flow networks techniques from combinatorial optimization for the exact global minimization of some widely used classes of energy functions in early vision. In particular, maximum flow algorithms of low-order polynomial time were used to find the global minimum in binary image restoration with the Potts prior [GPS89] and in stereo matching where $|L| > 2$ with linearly ordered labels and an absolute difference prior [IG98a] which was later extended to handle arbitrary convex priors [Ish03]. The binary exact optimization scheme in particular proved to be rather interesting in early vision and opened the door to a multitude of applications like interactive image and video segmentation, stereo, multiview reconstruction and texture synthesis (refer to the comprehensive list of references in [BF06]). Subsequently, [BVZ01] generalized the maximum flow-based binary optimization scheme to $|L| > 2$ for arbitrary labels and metric pairwise priors. In contrast to ICM and SA, their approximate labeling algorithm employed a search space exploration strategy, called an expansion move, whereby an arbitrarily large number of voxels can switch to a given label α simultaneously. The Expansion Moves algorithm, as we will refer to it henceforward, finds a local minimum such that no expansion move, for any label α, yields a labeling with lower energy. The local minimality criterion with respect to expansion moves is so strong that there are many fewer such minima in high-dimensional spaces compared to standard moves. [BVZ01] prove that any labeling locally optimal with respect to these moves is, in fact, within a known factor of the global minimum. Usually, maximum flow-based energy minimization approaches in vision are collectively referred to as Graph Cut methods.

More recently, the revival of older belief propagation methods [Pea88] has led to the development of the Sequential Max-Product Tree-Reweighted Message Passing algorithm (TRW-S) [Kol06]. It is often seen as a rival to the Expansion Moves and the Loopy Belief Propagation algorithms (LBP) [WF01] the latter of which has strong local minimality properties analogous to those of the former. [MYW05] has shown that TRW-S is capable of finding the global minimum for certain instances of the stereo problem. This algorithm can be applied to arbitrary energies on arbitrary label sets, although, like LBP, sometimes it might oscillate and fail to converge.

In order to understand the tradeoffs between different energy minimization methods, [SZS+08] have devised a set of benchmarks on common vision problems and used them to compare the solution quality and runtime of ICM, LBP, Graph Cut and TRW-S algorithms. They conclude that while the latter three come very close to the global minimum on different multilabeling problems, the Expansion Moves variant of Graph Cut algorithms consistently outperforms others on nonconvex energies in terms of both speed and quality of obtained solutions. [BVZ01] had earlier benchmarked SA and Expansion Moves algorithms and made similar conclusions. The comparative study of [SZS+08] also echoes the findings of [MYW05] in that better minimization methods are unlikely to produce significantly more accurate labelings, since the global minimum,

when known, is no more accurate than the ground-truth labeling, the energy of which is furthermore frequently higher than that of the former. This suggests, as pointed out by [MYW05], that often the problem is not in the optimization algorithm but in the energy formulation. Therefore, research should focus on finding better energy models.

Due to its optimality and runtime performance, we use Graph Cut algorithms for segmentation energy minimization. Closely following the method of [BF06], we use a maximum flow-based binary optimization scheme in mono-object segmentation where $|L| = 2$ and apply the Expansion Moves algorithm of [BVZ01] for multiobject segmentation with $|L| > 2$. In the remaining of this section, we shall give the necessary and sufficient conditions for the applicability of these methods to minimize the energy (2.10). Theoretical foundations and algorithmic details of maximum flow and Graph Cut methods are presented in Sections 2.3 and 2.4 respectively. Readers who are familiar with flow networks and Graph Cut methods can skip to Chapter 3 directly.

While the maximum flow-based binary energy minimization method is capable of finding the exact MAP solution [GPS89], the Expansion Moves algorithm produces a locally-optimal labeling $\hat{\ell}$ within a known factor of the global minimum ℓ^*, that is:

$$E(\hat{\ell}) \leq 2cE(\ell^*), \quad c = 2 \times \max V_{i,j}(a,b)/\min V_{i,j}(a,b), \quad \forall i,j \in \mathcal{S}, \forall a,b \in L. \tag{2.13}$$

For example, $c = 2$ for the Potts model. As mentioned previously, in the original work of [BVZ01], the Expansion Moves algorithm was shown to be applicable to energies of metric pairwise priors. [KZ04] subsequently characterized the classes of energy functions that can be minimized via Graph Cut algorithms and relaxed the metricity requirement. In particular, they showed that the Expansion Moves algorithm can be applied to minimize (2.10) with $|L| > 2$ if and only if the pairwise term is submodular, that is, if the following condition holds $\forall i,j \in \mathcal{S}, \forall a,b,c \in L$:

$$V_{i,j}(a,a) + V_{i,j}(b,c) \leq V_{i,j}(a,c) + V_{i,j}(b,a). \tag{2.14}$$

For $L = \{0,1\}$, the submodularity condition reduces to:

$$V_{i,j}(0,0) + V_{i,j}(1,1) \leq V_{i,j}(0,1) + V_{i,j}(1,0). \tag{2.15}$$

Submodular set functions and their properties and applications in combinatorial optimization are known to mathematicians for a long time [BKR96]. Importantly, optimization problems of functions having such properties are polynomial-time solvable. In fact, submodularity can be regarded as a property of discrete functions analogous to the convexity property of continuous functions [Muro3]. However, polynomial algorithms for minimizing arbitrary real-valued submodular functions have high complexity. For example, the algorithm proposed by [Orl09] runs in $O(n^5\gamma + n^6)$ time where γ is the time to evaluate the submodular function. Therefore, Graph Cut-minimizable functions can be regarded as a subclass of submodular functions which can be minimized with much faster algorithms. The optimization of submodular functions remains an active research field.

2.3 Flow networks

Flow networks are tools from combinatorial optimization used to solve problems that can be modeled as a steady-rate material flow through a directed capacitated network from a source node, where the material is produced, to a sink node where it is consumed. For example, we can think of liquid flows through pipes, and also electrical current and information through electrical and communication networks. Directed edges in such a network represent conduits each having a capacity indicating the maximum rate at which material can flow through it. Network nodes, other than the source and the sink, are conduit junctions through which material flows without collecting.

An important result in flow networks theory, due to [FF62], is the duality of Maximum flow and Minimum cut problems. The maximum flow is the greatest rate at which material can be sent from the source to the sink without violating any capacity constraints. A cut of the flow network is a partition of its nodes into two disjoint sets such that the source is in one set and the sink is in the other. The minimum cut has the minimum capacity among all cuts, measured as the sum of the capacities of edges going from the source set to the sink set. Importantly, the minimum cut capacity is equal to the maximum flow value which can be computed by efficient algorithms as we shall see in this section. Let us first define flow networks and their properties formally.

A flow network is a directed weighted graph $G = \langle V, E \rangle$ with no self-loops where we distinguish two special nodes, called terminals, the source s and the sink t, sometimes also called the target. Each node in V lies on a path from the source to the sink, therefore the graph G is connected and $|E| \geq |V| - 1$. In the special case of grid graphs with 4-connected and 6-connected neighborhood systems in 2D and 3D, $|E| \geq 2|V|$ and $|E| \geq 3|V|$, respectively. A flow network G is equipped with a positive capacity function of edges; $c \colon E \mapsto \mathbb{R}^+$ so that each edge $(u, v) \in E$ has a positive capacity $c(u, v) \geq 0$. Note that $c(u, v) = 0$ when $\nexists (u, v) \in E$.

Given a flow network $G = \langle V, E \rangle$ and a capacity function c, a flow f in G, is a real-valued function $f : V \times V \mapsto \mathbb{R}$ satisfying the following properties.

$$\text{Capacity constraint:} \quad \forall u, v \in V,\ f(u, v) \leq c(u, v) \tag{2.16a}$$

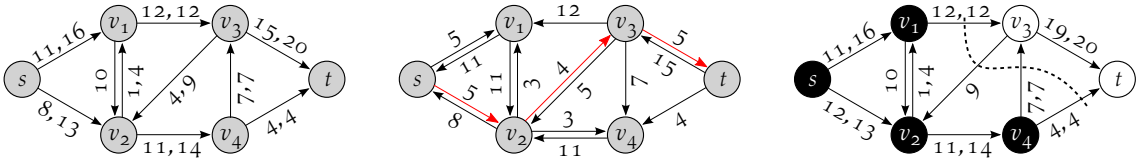
$$\text{Skew symmetry:} \quad \forall u, v \in V,\ f(u, v) = -f(v, u) \tag{2.16b}$$

$$\text{Flow conservation:} \quad \forall u \in V - \{s, t\},\ \sum_{v \in V} f(u, v) = 0 \tag{2.16c}$$

Obviously $f(u, v) = 0$ when $(u, v) \notin E$. The value of flow in the network G is given by:

$$|f| = \sum_{v \in V} f(s, v). \tag{2.17}$$

In the Maximum flow problem, given a flow network G we wish to find a flow of maximum value given by (2.17) respecting capacity constraints (2.16a) and the flow conservation property (2.16c). The skew symmetry property is merely a notational convenience indicating the direction of positive flow. Figure 2.1a illustrates a flow network having a valid flow.



(a) (b) (c)

Figure 2.1: Illustration of concepts of flow networks after [CLRS09]. (a) A flow network with $|f| = 19$ indicating, respectively, the flow and the capacity of each edge separated by a comma, negative flows are not shown, (b) the residual network induced by the previous flow with an augmenting path of capacity $c_f(p) = 4$ highlighted, (c) the flow network after augmentation through the highlighted path producing the indicated minimum cut with capacity $c(S,T) = 23$.

Two general methods exist for solving the Maximum flow problem; the Ford-Fulkerson method, sometimes also called the Augmented-paths method, and the Push-relabel method. Several algorithms of different running times implementing either approach have been proposed in the literature. In the following two sections, we outline each method providing more detail on the first since we use an algorithm that falls in this category. It is worth mentioning that the Maximum flow and Minimum cut problems can also be stated as two primal-dual linear programs [Law01]. However, such a formulation can be impractical for the large number of variables associated with vision and image processing applications.

2.3.1 The Ford-Fulkerson method

The Ford-Fulkerson method is characterized by three principal ideas upon which algorithms that implement it depend. They are furthermore essential to the Max-flow Min-cut theorem which ensures the correctness and the termination of such algorithms. In the following, we present these concepts before outlining the general strategy of finding a maximum flow by the Ford-Fulkerson method.

Residual networks. Starting from $|f| = 0$, the Ford-Fulkerson method iteratively increases the value of flow until no more flow can be sent from the source to the sink due to capacity constraints. The method maintains a residual network G_f whose topology is identical to that of G but the capacities of its edges represent the residual capacities of corresponding edges in G. The residual capacity of an edge (u,v) in G is the amount of additional flow we can push from u to v before exceeding $c(u,v)$. It is given by:

$$c_f(u,v) = c(u,v) - f(u,v). \tag{2.18}$$

The residual network G_f provides a roadmap for increasing flow in the original network with a simple rationale. Consider, for example, the edge (s,v_1) in Figure 2.1a with $c(s,v_1) = 16$ and $f(s,v_1) = 11$. We can increase flow on (s,v_1) by $c_f(s,v_1) = 5$ before exceeding the capacity constraint on the edge. Consider the negative flow $f(v_1,v_2) = -1$. In this case, the residual capacity $c_f(v_1,v_2) = 11$ is greater than the capacity $c(v_1,v_2)$, meaning that we may cancel the flow $f(v_2,v_1) = 1$ by pushing a flow of 1 from v_1 to

v_2, and subsequently push a flow of 10 before violating the capacity constraint $c(v_1, v_2)$. Note that $c_f(u,v) = c_f(v,u) = 0$ when neither edge is in G, therefore an edge (u,v) appears in G_f if at least one of (u,v) and (v,u) are in G. Formally then, given a flow network $G = \langle V, E \rangle$ and a flow f, the residual network of G induced by f is $G_f = \langle V, E_f \rangle$ where $E_f = \{(u,v) \in V \times V \mid c_f(u,v) > 0\}$ and $|E_f| \leq 2|E|$. To illustrate, the complete residual graph of the flow network in Figure 2.1a is given in Figure 2.1b.

Augmenting paths. Given a flow network $G = \langle V, E \rangle$ and a flow f, an augmenting path p is a simple path[3] from the source to the sink in the residual network G_f. We can increase the flow on an edge (u,v) of an augmenting path by up to $c_f(u,v)$ without violating the capacity constraint on whichever of (u,v) or (v,u) is in the flow network G. The maximum amount by which we can increase the flow through the path p is called the residual capacity of p and is given by $c_f(p) = \min\{c_f(u,v) \mid (u,v) \in p\}$. Highlighted edges in Figure 2.1b indicate an augmenting path with a residual capacity of $c_f(p) = 4$.

Cuts of flow networks. The last concept we introduce before presenting the Max-flow Min-cut theorem and outlining the Ford-Fulkerson general procedure, is that of cuts of flow networks. As mentioned in the beginning of this section, a cut of a flow network $G = \langle V, E \rangle$, denoted as (S,T), is a partition of its set of nodes V into two disjoint sets S and $T = V - S$ such that $s \in S$ and $t \in T$. The capacity of a cut is given by:

$$c(S,T) = \sum_{u \in S} \sum_{v \in T} c(u,v). \tag{2.19}$$

Obviously, the minimum cut of the network is the cut that has the minimum capacity among all cuts of the network. Figure 2.1c shows the minimum cut of the earlier flow network example having a capacity of $c(S,T) = 23$.

Theorem 2.3.1 (Max-flow Min-cut theorem). *Given a flow f in a flow network $G = \langle V, E \rangle$, the following conditions are equivalent:*

- *f is a maximum flow in G.*
- *There are no augmenting paths in the residual network G_f.*
- *$|f| = c(S,T)$ where (S,T) is the minimum cut of G.*

We refer the reader to [CLRS09] for the proof. Intuitively, the Max-flow Min-cut theorem states that the value of the flow in a network is bounded from above by the capacity of its minimum cut, and provides the termination condition for maximum flow algorithms following the Ford-Fulkerson method; the algorithm terminates and the flow is maximum if and only if its residual network contains no augmenting paths.

With all concepts in place, the general Ford-Fulkerson method of finding a maximum flow boils down to a simple procedure. Starting with $f(u,v) = 0 \forall (u,v) \in E$ giving an initial flow of $|f| = 0$, at each iteration, we find an augmenting path p in the residual network G_f induced by the current flow f in G, and for each edge (u,v) in p we

[3] A simple path in a directed graph is one the nodes of which are distinct.

increase the flow $f(u,v)$ in G by $c_f(p)$ and set the flow on the reverse edge $f(v,u)$ to the negative of the increased flow $-f(u,v)$. After each iteration, the value of flow in G equals $|f| + c_f(p)$. When no augmenting paths exist in G_f, this flow is a maximum flow. Figure 2.1 illustrates the last steps of the method's application. Given the network in Figure 2.1a with the flow $|f| = 19$, the path $p = (s, v_2, v_3, t)$ with $c_f(p) = 4$ is found in the residual network G_f given in Figure 2.1b and the flow in G is updated along the edges of p giving a new flow of $|f| = 23$ in Figure 2.1c, which is the maximum flow.

In practice, the Maximum flow problem arises with integer capacities. If capacities are rational numbers, they can be converted to integers with an appropriate scaling transformation. With irrational capacities however, the Ford-Fulkerson method might iterate infinitely, finding augmenting paths of increasingly lower capacity. Worse, the infinite sequence of augmentations may not even converge to the maximum flow value. [Zwi95] illustrates the smallest network on which this problem occurs. Therefore, irrational capacities have to be approximated by rational numbers or integers.

Implementations of the Ford-Fulkerson method address two questions; how augmenting paths are found, and in what order such paths are processed. Denoting the maximum flow by f^*, a straightforward implementation executes $|f^*|$ iterations at most, assuming that the flow increases by one unit at each iteration. If augmenting paths in the residual network are found using either a breadth-first or a depth-first search with a complexity of $O(V + E_f) = O(E_f)$ with no regard to path length, the total running time of the method would be $O(|f^*|E_f)$. The Edmonds-Karp algorithm [EK72] improves the bound on the number of iterations to $O(VE_f)$. Attributing unit weight to each edge in the residual network, in each iteration the algorithm choses the augmenting path corresponding to the shortest path from the source to the sink found by breath-first search. Therefore the algorithm runs in $O(VE_f^2)$ time. An algorithm due to [AO89] uses a scaling technique and runs in $O(E_f^2 \log_2 C)$ time where $C = \max_{\forall (u,v) \in E}\{c(u,v)\}$. Starting with $K = 2^{\lfloor \log_2 C \rfloor}$, and while $K \geq 1$, the algorithm finds augmenting paths of capacity at least K at each iteration halving K at the end of it.

In an attempt to improve the empirical performance of the Ford-Fulkerson method in vision applications involving grid graphs, [BK04] observe that building search trees to find augmenting paths in every iteration, which is a $O(E_f)$ operation, is prohibitively expensive especially in 3D where usually $|E| > 3|V|$. They propose reusing such trees. Their algorithm maintains two search trees rooted at the source and the sink respectively. In each iteration, each tree is grown through nonzero residual capacity edges acquiring new children from graph nodes until a leaf node from a tree comes in contact with a leaf node from the other tree resulting in an augmenting path. Flow is pushed through this path in the familiar manner, which breaks search trees into forests. Next, tree structures are restored by assigning nodes that have been disconnected by augmentation to new parents via nonzero capacity edges. If no parent can be found, disconnected nodes are made free nodes. The algorithm terminates when search trees cannot grow due to their leaves being separated by zero residual capacity edges. At this point, the flow is maximum and the nodes of either tree fall in the source and the sink set respectively.

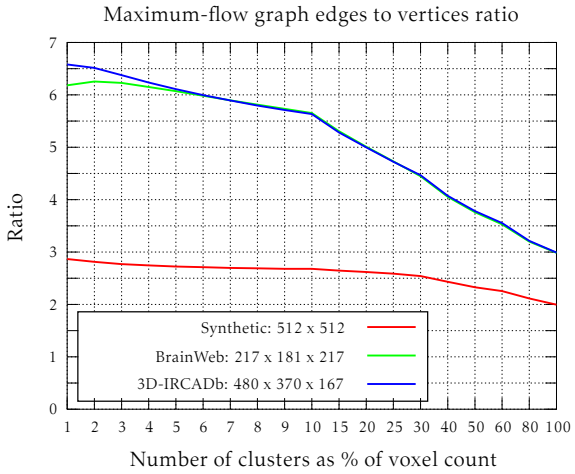


Figure 2.2: Number of flow network edges to number of nodes ratio for cluster graphs of 3 images, calculated at 19 increasing resolutions with 100% corresponding to the voxel grid graph. Note that edges connecting nodes to flow network terminals are not taken into account as their number is always $2|V|$.

The obvious drawback of the algorithm proposed by [BK04] is that augmenting paths found are not necessarily the shortest. As reported, the processing order of nodes at tree growth and restoration stages significantly impacts the algorithm's empirical runtime. Authors proceed in a First-In-First-Out (FIFO) fashion which ensures that the first path found is the shortest, with no guarantees on subsequently found paths since search trees may change unpredictably during the restoration stage. In order to make augmenting paths as short as possible, distance-to-root information is stored in each node so that disconnected nodes can be assigned to parents closer to a root. Although no asymptotic bound is given, the algorithm has supralinear empirical time in vision problems involving 2D and 3D grid graphs and outperforms the Edmonds-Karp algorithm and two algorithms among the best performing Push-relabel variants, namely those with queued and highest node selection rules discussed in the next section.

We use the algorithm of [BK04] in our maximum flow-based image segmentation applications. As we shall see in Chapter 4, in order to reduce the redundancy of voxel data and improve runtime performance, we formulate our segmentation method in terms of clusters of voxels. As shown in Figure 2.2, the neighborhood connectivity of such irregular cluster graphs is higher than that of image grid graphs. For example, we can see that with a clustered 3D image where the number of clusters equals 5% of the number of voxels, the number of neighbor edges of the cluster graph are as high as six times the number of voxels. This means that any algorithm that finds augmenting paths with breadth-first search would perform worse than the algorithm of [BK04].

2.3.2 The Push-relabel method

For the completeness of the presentation of maximum flow approaches, we outline the general Push-relabel method in this section pointing at common implementations. The Push-relabel method differs from the Ford-Fulkerson method in two primary ways. Firstly, the Ford-Fulkerson method examines the entire residual network in search for augmenting paths, and each such path gives rise to an additional stream of flow from the source to the sink until no more flow can be pushed. The Push-relabel method works in a localized fashion, examining a single node at a time and its neighbors in the residual network, thus the method readily admits parallel implementations. Secondly, while the Ford-Fulkerson method maintains the flow conservation property (2.16c) throughout its execution, the Push-relabel method allows each node $u \in V - \{s, t\}$ to have a positive flow excess given by:

$$e(u) = \sum_{v \in V} f(v, u) \geq 0. \tag{2.20}$$

In order to determine how flow is pushed from an overflowing node to its neighbors, the method maintains a height attribute for each node. Although flow from a lower node to a higher neighbor can be positive, flow can be pushed only from a higher node to a lower neighbor. In the beginning, the height of the source and the sink are set to $|V|$ and 0 respectively, and those of all other nodes start at 0 and increase as the algorithm progresses. The algorithm first sends the largest possible flow from the source towards the sink, equal to the capacity of the cut $(s, V - \{s\})$, which collects in intermediate nodes. In order to rid an overflowing node of excess flow, the algorithm increases its height to one unit above the height of the lowest of its neighbors with which it shares a nonzero capacity residual edge. This operation is called relabeling, after which the node can push excess flow to at least one of its neighbors. All flow that can reach the sink eventually arrives there and in order to make flow in the network a valid flow, undeliverable flows are drained back to the source by continuing to relabel nodes above the fixed height $|V|$ of the source. When all excess flow is drained, the flow in the network is a maximum flow.

The basic Push-relabel method, due to [Gol87], thus boils down to an initialization followed by a sequence of push and relabel operations, whichever is applicable, executed in no particular order. It has an asymptotic bound of $O(V^2E)$, which is better than that of the Edmonds-Karp algorithm mentioned above. [GT88] propose two improvements on this bound; $O(V^3)$ by maintaining overflowing nodes in a FIFO queue and always selecting the front node for discharging, that is applying push and relabel operations successively until the said node has no positive excess, and adding newly overflowing nodes to the rear of the queue, and $O(VE \log_2(V^2/E))$ by using an efficient dynamic tree data structure to reduce per push-operation time. [CG97] address practical aspects of implementing the Push-relabel method underlining the importance of using heuristics. They conclude that the best Push-relabel algorithm is one that repeatedly discharges the highest overflowing node. It has an asymptotic bound of $O(V^2\sqrt{E})$.

It is worth mentioning as we conclude this section that the asymptotically fastest deterministic sequential algorithm for the Maximum flow problem of which we are aware is due to [GR98] which is not a Push-relabel algorithm. It uses the concept of blocking flows and associates shorter lengths with higher-capacity edges in order for shortest $s-t$ paths to have high capacity and the algorithm to perform fewer iterations. The algorithm runs in $O(\min(V^{2/3}, E^{1/2})E\log_2(V^2/E)\log_2 C)$ time where C is the maximum edge capacity. Its performance on flow networks arising in vision is unknown to us.

2.4 Graph Cut methods

Having discussed the Maximum flow problem and algorithms to solve it, we return our attention to minimizing the energy (2.10) with Graph Cut algorithms. In Section 2.2.3 we outlined the general properties of this family of energy minimization algorithms mentioning that they apply to energy functions of the form (2.10) with submodular pairwise energy terms (2.14), and that in practice they outperform popular energy minimization algorithms, in terms of both runtime and quality of produced minima, on high-dimensional nonconvex energies frequently encountered in vision. In this section we examine the algorithmic details and the properties of Graph Cut algorithms.

Maximum flow methods lie at the heart of Graph Cut algorithms. In fact, every such algorithm entails a method of representing the energy (2.10) as a flow network so that a cut on the network and its capacity correspond to a label configuration and its energy respectively, furthermore ensuring that the minimum cut of the network minimizes the energy among all possible cuts.

Graph Cut methods give two interpretations to cuts of flow networks: 1) a geometrical interpretation, typically arising in binary problems, which views such cuts as optimal separating hypersurfaces in an N-D manifold embedding the graph equipped with a cut metric approximating any continuous Riemannian metric [BK03, KB05], and 2) as a classifier, finding the optimal attribution of two or more label classes to a set of variables [GPS89, BVZ01, Ish03]. In both cases, maximum flow algorithms constitute the numerical optimization tool of energies describing such hypersurfaces and classes.

Maximum flow methods, as we have seen in Section 2.3, are inherently binary techniques; they produce an optimal bipartition of the nodes of a flow network in terms of the capacity of the cut separating the two sets. In the following section, we shall show that binary problems in vision like mono-object segmentation, binary image restoration etc. can be most naturally and directly formulated and efficiently solved by maximum flow algorithms. Importantly, for energies of binary variables, the MAP solution (2.9), that is the global minimum of the energy (2.10), can be obtained via Graph Cut methods in polynomial time matching that of the underlying maximum flow algorithm. Moreover, multilabel energies, discussed in Section 2.4.2, can be minimized with Graph Cut methods by transforming the multilabling problem to a sequence of binary subproblems each formulated as a Maximum flow problem on an appropriate network.

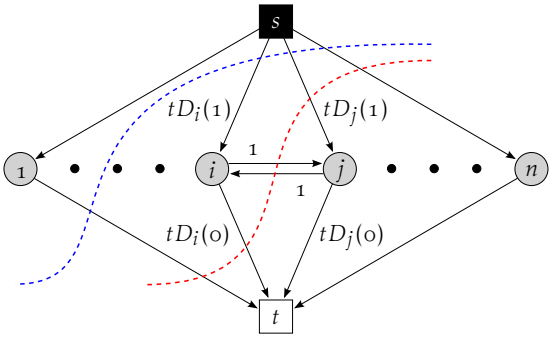


Figure 2.3: A flow network representing the energy (2.10) with binary variables $x_i \in \{0,1\}$. The configuration induced by the blue cut is $x_i = x_j = 1$ and that of the red cut is $x_i = 0, x_j = 1$

2.4.1 Graph Cut in binary optimization

In this section, we discuss one of the most straightforward Graph Cut applications of mono-object segmentation showing how the minimization of the energy (2.10) with binary variables can be formulated as a Maximum flow problem. The original formulation is due to [GPS89] in the context of binary image restoration. Their approach remained relatively unknown to the vision community until it was reintroduced and popularized by [BJ01] in mono-object interactive segmentation. Our discussion in this section closely follows the method presented in the latter work.

Let us begin by defining the flow network. As before, let $G = \langle V, E \rangle$ be a directed connected graph with a positive capacity function of edges. Let $\mathcal{S}$ be the set of integers indexing the variables of the energy (2.10) and the corresponding image primitives which are considered to be image voxels in this discussion. The set of nodes of the graph G is $V = \mathcal{S} \cup \{s,t\}$ where s and t are the source and the sink terminals of the flow network respectively. The set of edges E consists of two types of edges; those connecting nodes in $\mathcal{S}$ to either terminal, called t-links (for terminal links), and those connecting pairs of nodes in $\mathcal{S}$, called n-links (for neighbor links). The latter are dictated by the neighborhood system of the energy (2.10). The set of edges is therefore given by $E = \left\{\bigcup_{i\in\mathcal{S}}\{(s,i),(i,t)\}, \bigcup_{i,j\in\mathcal{S}}\{(i,j) \mid j \in N_i\}\right\}$. An illustration of the graph G is given in Figure 2.3 for a one-dimensional image.

Let us formulate mono-object segmentation according to the Bayesian labeling framework presented earlier in Section 2.2. For this binary problem, the set of labels is $L = \{0,1\}$. We assume that "0" represents the background and "1" the sought object. We take the observation to be image intensity values at each variable, and assume that likelihood densities are defined for both labels. Disregarding user constraints, data terms of the energy (2.10) are given by (2.11c). For pairwise energy terms, the Potts prior model (2.12) suffices to ensure the spatial consistency of this instance of binary labeling. Note that data terms are positive by definition and that if a variable $i \in \mathcal{S}$ prefers a label, say the object label "1", then $D_i(1) < D_i(0)$.

The one-to-one correspondence between cuts of the graph G and binary configurations of variables in $\mathcal{S}$ is fairly obvious. A cut of G, which partitions its nodes into

the source S and the sink T sets respectively, can be interpreted as a binary configuration ℓ which assigns labels $\{0,1\}$ to variables in $\mathcal{S}$ corresponding to nodes such that $\ell_i = 0$ if $i \in S$ and $\ell_i = 1$ if $i \in T$. Clearly, any cut (S,T) on G corresponds to a configuration ℓ. There remains to show that the capacities of the edges of G can be chosen so that the capacity of a cut is equal to the energy of the configuration induced by it. Given a cut (S,T), we denote the set of edges separating the source and the sink sets by $E^{(S,T)} = \{(u,v) \mid u \in S, v \in T\}$. Consider the graph in Figure 2.3. Cuts on G can include t-links and n-links, however for each node, exactly one t-link is in $E^{(S,T)}$. Thus, if $i \in S$, $(i,t) \in E^{(S,T)}$, and if $i \in T$, $(s,i) \in E^{(S,T)}$. Choosing $c(s,i) = tD_i(1)$ and $c(i,t) = tD_i(0)$, every node $i \in \mathcal{S}$ contributes the data penalty associated with the chosen label to the t-link part of the cut capacity. Note that an n-link (i,j) is in $E^{(S,T)}$ if and only if $i \in S$ and $j \in T$, thus by setting $c(i,j) = 1$, every n-link $(i,j) \in E$ accounts for the Potts prior penalty in the cut capacity. The capacity of a cut on the network G thus defined is given by:

$$c(S,T) = t \sum_{i \in \mathcal{S}} D_i(\ell_i) + \sum_{\substack{(i,j) \in E \\ i \in S, j \in T}} 1 \tag{2.21}$$

which is clearly equal to the energy of the configuration ℓ induced by the cut (S,T) as defined by (2.10) with the condition $\ell_i \neq \ell_j$ for pairwise terms. The minimum cut of G therefore yields the MAP configuration ℓ^* and its capacity is equal to the energy $E(\ell^*)$.

2.4.1.1 General-purpose constructions for binary energies

The graph construction we outlined in Section 2.4.1 assumes that unary energy terms are positive and that pairwise terms follow the Potts model. The natural question that arises consequently is: could we construct such graphs representing arbitrary energies? The seminal work [KZ04] answers this question giving a necessary graph representability condition for functions that can be written as a sum of terms of arbitrary order. Importantly, it shows that this condition is sufficient for functions which can be written as a sum of up to tertiary terms. In our work, we formulate mono and multiobject segmentation with binary and multilabel energies written as a sum of up to binary terms, therefore, in the following we shall present the main result of [KZ04] for such energies.

Theorem 2.4.1 (Graph Representability). *Let F be a function of n binary variables indexed by $\mathcal{S}$ written as:*

$$F(x_1, \ldots, x_n) = \sum_{i \in \mathcal{S}} F_i(x_i) + \sum_{(i,j) \in N} F_{i,j}(x_i, x_j), \tag{2.22}$$

where $N \subset \mathcal{S} \times \mathcal{S}$ is a neighborhood system containing ordered pairs (i,j) for which $i < j$. F is graph representable if and only if each term $F_{i,j}$ satisfies the following inequality:

$$F_{i,j}(0,0) + F_{i,j}(1,1) \leq F_{i,j}(0,1) + F_{i,j}(1,0). \tag{2.23}$$

Note that (2.23) is the submodularity condition (2.15) we mentioned earlier in Section 2.2.3. To make the distinction between functions of sets and functions of binary

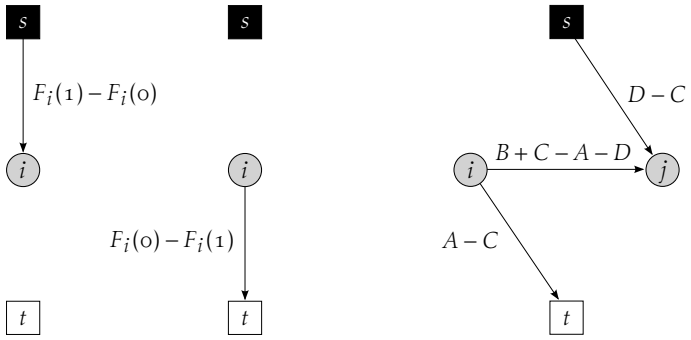


Figure 2.4: Graph representations due to [KZ04] for (a) unary and (b) pairwise terms of the function (2.22).

$$F_{i,j} = \begin{array}{|c|c|}\hline F_{i,j}(0,0) & F_{i,j}(0,1) \\ \hline F_{i,j}(1,0) & F_{i,j}(1,1) \\ \hline \end{array} = \begin{array}{|c|c|}\hline A & B \\ \hline C & D \\ \hline \end{array} = A + \begin{array}{|c|c|}\hline 0 & 0 \\ \hline C-A & C-A \\ \hline \end{array} + \begin{array}{|c|c|}\hline 0 & D-C \\ \hline 0 & D-C \\ \hline \end{array} + \begin{array}{|c|c|}\hline 0 & B+C-A-D \\ \hline 0 & 0 \\ \hline \end{array}$$

Table 2.1: Factorization of a pairwise function of binary variables into a constant, two unary functions and a pairwise function respectively.

variables, [KZ04] call it the regularity condition, hence binary functions satisfying (2.23) are called regular functions. In our discussion, we use the more widely used terminology of submodularity and submodular functions.

In the following, we shall outline the constructive proof of theorem 2.4.1 showing that submodularity is a sufficient condition for graph representability by describing the graph that represents an arbitrary submodular function of the form (2.22). We refer the reader to [KZ04] for the proof of the reverse direction as well as for graph constructions representing submodular functions written as a sum of up to tertiary terms. Another important theoretical result reported in this work is that the minimization of nonsubmodular functions of the form (2.22) is an NP-hard problem.

Given a cut (S,T) on the graph we defined earlier in Section 2.4.1, which represents a binary energy function with positive data and Potts prior terms, the energy of the configuration ℓ induced by the cut is exactly equal to the latter's capacity $E(\ell) = c(S,T)$. Observing that the configuration which minimizes the binary function (2.22) does not change if we add a constant to it, for the constructive proof of the theorem 2.4.1 we shall relax the last equality requiring that $F(x_1,\ldots,x_n) = c(S,T) + K$, where K is some constant. Obviously, F is exactly represented by a graph if $K = 0$.

Let F be a submodular binary function of the form (2.22) and $G = \langle V, E\rangle$ a flow network. We shall construct a subgraph representing each term of F separately and merge all subgraphs into G afterwards. This is justified by the additivity theorem outlined in [KZ04] which states that the sum of two graph-representable functions is graph-representable. Thus, we can illustrate the construction on the minimal example of a pair of variables indexed by i, j. As before, the set of nodes of the graph G is $V = \{s, t, i, j\}$ where each nonterminal node encodes a binary variable. For each term in F, we shall add one or more edges to E so that edge capacities are positive as required.

Consider first the unary terms of F. As illustrated in Figure 2.4a, the subgraph corresponding to a single variable i contains the nodes $\{s, t, i\}$. If $F_i(0) < F_i(1)$, we add the edge (s, i) to the subgraph with capacity $c(s, i) = F_i(1) - F_i(0)$, otherwise we add the edge (i, t) with capacity $c(i, t) = F_i(0) - F_i(1)$. We can see that the constructed subgraph represents the function F_i in both cases so that $F_i(x_i) = c(S, T) + K$ with $K = F_i(0)$ in the former and $K = F_i(1)$ in the latter case.

Consider now the pairwise terms of F. The subgraph corresponding to a pair of variables i, j naturally contains the nodes $\{s, t, i, j\}$. It is convenient to represent such a term in tabular form according to the four combinations of its variables' values as in Table 2.1. As shown, we rewrite the pairwise term in four terms and consider each for edge addition separately. The first term A is a constant, therefore no edges need to be added for it. The second and third terms are unary functions which depend on x_i and x_j respectively, therefore we can use the previous construction for unary terms to represent these terms. To represent the fourth term, we add the edge (i, j) with capacity $c(i, j) = B + C - A - D$. Note that $B + C - A - D \geq 0$ since F is submodular. Figure 2.4b illustrates the construction corresponding to the case $A - C > 0$ and $D - C > 0$. Note that in this case $F_{i,j}(x_i, x_j) = c(S, T) + F_{i,j}(1, 0)$.

As mentioned previously, we obtain the graph G that represents the function F by merging separately constructed subgraphs for unary and pairwise terms and adding edge capacities together. For referencing purposes, Table 2.2 summarizes graph constructions for submodular binary functions written as a sum of up to binary terms.

2.4.2 Generalized Graph Cut for multilabel energies

Many interesting vision tasks can be formulated as labeling problems with $|L| > 2$ and solved by multilabel optimization of energies of the form (2.10). To mention few instances, the set of labels L can represent disparities in stereo matching, motion vectors in motion estimation, intensity levels in grayscale image restoration and object identifiers in multiobject segmentation.

We have reported in Section 2.2.3 that minimizing nonconvex multilabel energies of the form (2.10) is NP-hard, even with the simple discontinuity-preserving Potts prior model. We are aware of only two Graph Cut methods for exact multilabel minimization of the energy (2.10): [IG98a] for stereo matching with an absolute difference prior and [Ish03] which generalizes the first method to handle arbitrary convex priors. Both of these methods require that labels are linearly ordered integers $L = \{1, \ldots, k\}$ with $k = |L|$, which limits their applicability to few problems. Unsurprisingly, therefore, for many interesting instances of multilabel minimization, researchers resort to approximate labeling algorithms which produce provably good local solutions. One Graph Cut method in particular, called the Expansion Moves algorithm [BVZ01], produces local minima within a known factor of the global minimum. Its practical advantages have been demonstrated in a range of vision problems on which it was shown to outperform popular multilabel minimization algorithms in terms of runtime and quality of obtained minima [SZS+08]. In our work, we use the Expansion Moves algorithm to

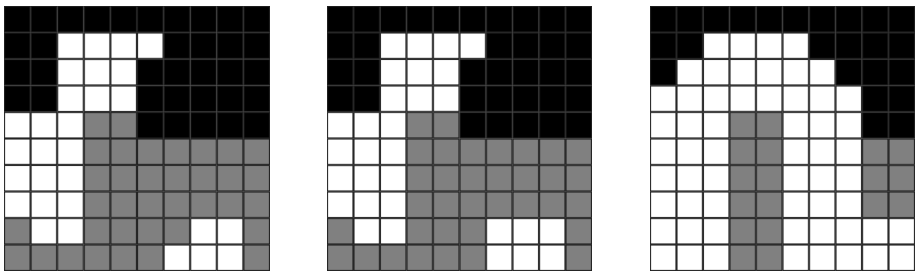
term	V	condition	c	K
$F_i(x_i)$	$\{s,t,i\}$	$F_i(1)-F_i(0)>0$	$c(s,i)=F_i(1)-F_i(0)$ $c(i,t)=0$	$F_i(0)$
		$F_i(0)-F_i(1)>0$	$c(s,i)=0$ $c(i,t)=F_i(0)-F_i(1)$	$F_i(1)$
$F_{i,j}(x_i,x_j)$	$\{s,t,i,j\}$	$F_{i,j}(1,0)-F_{i,j}(0,0)>0$ $F_{i,j}(1,1)-F_{i,j}(1,0)>0$	$c(s,i)=F_{i,j}(1,0)-F_{i,j}(0,0)$ $c(s,j)=F_{i,j}(1,1)-F_{i,j}(1,0)$ $c(i,t)=c(j,t)=0$	$F_{i,j}(0,0)$
		$F_{i,j}(0,0)-F_{i,j}(1,0)>0$ $F_{i,j}(1,1)-F_{i,j}(1,0)>0$	$c(s,j)=F_{i,j}(1,1)-F_{i,j}(1,0)$ $c(i,t)=F_{i,j}(0,0)-F_{i,j}(1,0)$ $c(s,i)=c(j,t)=0$	$F_{i,j}(1,0)$
		$F_{i,j}(1,0)-F_{i,j}(0,0)>0$ $F_{i,j}(1,0)-F_{i,j}(1,1)>0$	$c(s,i)=F_{i,j}(1,0)-F_{i,j}(0,0)$ $c(j,t)=F_{i,j}(1,0)-F_{i,j}(1,1)$ $c(s,j)=c(i,t)=0$	$F_{i,j}(0,0)+F_{i,j}(1,1)$ $-F_{i,j}(1,0)$
		$F_{i,j}(0,0)-F_{i,j}(1,0)>0$ $F_{i,j}(1,0)-F_{i,j}(1,1)>0$	$c(i,t)=F_{i,j}(0,0)-F_{i,j}(1,0)$ $c(j,t)=F_{i,j}(1,0)-F_{i,j}(1,1)$ $c(s,i)=c(s,j)=0$	$F_{i,j}(1,1)$
		$F_{i,j}(1,0)+F_{i,j}(0,1)$ $-F_{i,j}(0,0)-F_{i,j}(1,1)\geq 0$	$c(i,j)=F_{i,j}(1,0)+F_{i,j}(0,1)$ $-F_{i,j}(0,0)-F_{i,j}(1,1)$ $c(j,i)=0$	

Table 2.2: Summary of graph constructions due to [KZ04] for submodular binary functions of the form (2.22). For every term, the subgraph definition corresponding to indicated positivity conditions is given along with the constant K so that $F(x)=c(S,T)+K$. The graph representing F is obtained by merging subgraphs for unary and pairwise terms and adding edge capacities and constants K together respectively.

minimize multilabel segmentation energies, therefore in the following we shall discuss its properties and present its algorithmic details.

The Expansion Moves algorithm takes its name from its search space exploration strategy, the expansion move, whereby an arbitrarily large number of variables can switch to a label α simultaneously. The algorithm finds a local minimum such that no expansion move, for any label α, yields a labeling with lower energy. This strong local minimality criterion is actually what bounds solutions found by the algorithm. By contrast, traditional energy minimization algorithms, like the Iterative Conditional Modes (ICM) algorithm [Bes86] and the Simulated Annealing (SA) algorithm [KGV83], use a much simpler search space exploration strategy, called the standard move, where only a single variable can change its label at a time. In the following, we formally define expansion and standard moves and compare their local minimality criteria.

A configuration ℓ can alternatively be seen as a partition of the set of variables, indexed by $\mathcal{S}$, on which the energy (2.10) is defined. It is given by $\mathcal{P}=\{P_l \mid l\in L\}$, where $P_l=\{i\in\mathcal{S}\mid \ell_i=l\}$ is the subset of variables that are assigned the label l in the configuration ℓ. Given a label $\alpha\in L$, a move from an initial configuration ℓ, mirrored by the partition $\mathcal{P}$, to a new configuration ℓ' attainable from ℓ, mirrored by the partition



(a) (b) (c)

Figure 2.5: A labeling instance with $L = \{$"*black*","*gray*","*white*"$\}$. (a) The initial configuration, (b) a standard move on the white label, (c) an expansion move on the white label.

$\mathcal{P}'$, is called an α-expansion if $P_\alpha \subset P'_\alpha$ and $P'_l \subset P_l$ for any label $l \neq \alpha$. In other words, an α-expansion allows any number of variables to change their labels to α growing the set P_α in the partition $\mathcal{P}$. Figure 2.5c illustrates an expansion move on the white label from the initial configuration given in Figure 2.5a. A move from an initial configuration ℓ, mirrored by the partition $\mathcal{P}$, to a new configuration ℓ' attainable from ℓ, mirrored by the partition $\mathcal{P}'$, is called a standard move if $\exists_{=1} l \in L$ such that $\left||P'_l| - |P_l|\right| = 1$. Figure 2.5b illustrates a standard move on the white label from the initial configuration.

Given a configuration ℓ, denote the set of possible standard or expansion moves from ℓ by M_ℓ. We say that $\hat{\ell}$ is a local minimum of the energy E with respect to M_ℓ if:

$$\forall \ell' \in M_\ell, E(\hat{\ell}) \leq E(\ell'). \tag{2.24}$$

It is easy to see that for standard moves $|M_\ell|$ is linear in the number of variables, which makes it easy to find a local minimum with respect to these moves. However, the local minimality condition (2.24) when standard moves are allowed is a very weak one, because when at a local minimum with respect to these moves, one cannot find a configuration with lower energy by changing the label of a single variable. This explains the low quality results frequently generated by untuned ICM and SA algorithms and their sensitivity to initialization especially in minimizing high-dimensional nonconvex energies which have many such local minima. By contrast, when expansion moves are allowed, $|M_\ell|$ is exponentially large since if the move involves m variables of a total of n, the number of expansion moves is n^m. This makes the condition (2.24) so demanding and the number of such local minima so few that [BVZ01] are able to prove that any configuration locally minimal with respect to expansion moves is within a known factor of the global minimum. The strong local minimality property also makes expansion moves less sensible to the initial labeling. [BVZ01] show that $E(\hat{\ell}) \leq 2cE(\ell^*)$ where $c = 2 \times \max V_{i,j}(a,b)/\min V_{i,j}(a,b), \forall i,j \in \mathcal{S}, a,b \in L$ and ℓ^* is the global minimum of E. Note that the algorithm has the best bound on local minima for the Potts model $c = 2$.

Given a configuration ℓ, there are, as we have seen, an exponential number of expansion moves from ℓ, therefore a direct search for the optimal expansion move is infeasible. This is where binary Graph Cut techniques discussed in Section 2.4.1 come

into play. [BVZ01] show that computing the optimal expansion on a label can be specified as a binary optimization problem, in effect, transforming the multilabling problem to a sequence of binary subproblems each formulated as a Maximum flow problem on an appropriate graph. In the following, using the results of [KZ04] we show how to formulate an expansion move as a binary labeling problem and find the optimal move, that is the move that gives the largest decrease in energy, by computing a minimum cut. We also derive the multilabel submodularity condition mentioned earlier (2.14) that must be met by the pairwise terms of the energy (2.10) so that the method applies.

Consider some initial configuration ℓ and a label we wish to expand on $\alpha \in L$. Recall that after an expansion move on α, a variable $i \in \mathcal{S}$ either keeps its current label ℓ_i, or switches to α and that variables that are already labeled α naturally keep their label. We can define such a label attribution as a function of binary variables

$$f_i(x_i) = \begin{cases} \ell_i, & x_i = 0 \\ \alpha, & x_i = 1 \end{cases} \tag{2.25}$$

and write the multilabel energy function (2.10) as follows:

$$E(x_1, \ldots, x_n) = t \sum_{i \in \mathcal{S}} D_i(f_i(x_i)) + \sum_{i,j \in \mathcal{S},\, j \in \mathcal{N}_i} V_{i,j}(f_i(x_i), f_j(x_j)). \tag{2.26}$$

As before, a graph with two terminal nodes can be constructed to represent the energy (2.26) with the following conventions. Given a cut (S, T), if $i \in S$, then $x_i = 0$, therefore $f_i(0) = \ell_i$ and the label keeps its current label, otherwise if $i \in T$, then $x_i = 1$, therefore $f_i(1) = \alpha$ and the label switches to α. The graph can then be constructed by the method of [KZ04] presented in Section 2.4.1.1 and summarized in Table 2.2.

The original work which proposed the Expansion Moves algorithm [BVZ01] relied on a graph construction requiring metric pairwise terms, that is $\forall \alpha, \beta, \gamma \in L$:

$$\begin{aligned} &V(\alpha, \beta) = 0 \Leftrightarrow \alpha = \beta, \\ &V(\alpha, \beta) = V(\beta, \alpha) \geq 0, \\ &V(\beta, \gamma) \leq V(\beta, \alpha) + V(\alpha, \gamma). \end{aligned} \tag{2.27}$$

In particular, symmetry is assumed by [BVZ01] only to simplify graph constructions. [KZ04] subsequently showed that metricity is an accidental property of elaborate graph constructions the original method employed. They conclude that the energy function (2.26) is graph representable if and only if each pairwise term satisfies the inequality:

$$V_{i,j}(f_i(0), f_j(0)) + V_{i,j}(f_i(1), f_j(1)) \leq V_{i,j}(f_i(0), f_j(1)) + V_{i,j}(f_i(1), f_j(0)) \tag{2.28}$$

that is

$$V_{i,j}(\beta, \gamma) + V_{i,j}(\alpha, \alpha) \leq V_{i,j}(\beta, \alpha) + V_{i,j}(\alpha, \gamma) \tag{2.29}$$

where $\beta = \ell_i$ and $\gamma = \ell_j$. Setting $V_{i,j}(\alpha, \alpha) = 0$, we obtain the triangle inequality, as required by [BVZ01].

Algorithm 2.4.1: Expansion Moves energy minimization

input : set of labels L, energy function of label configurations $E(\ell)$ on a set of variables indexed by $\mathcal{S}$, maximum cycles $maxCycles$
output: label configuration ℓ

```
   // Initialize configuration
1  l_init ← RANDOM(l ∈ L)
2  foreach i ∈ S
3  |  ℓ_i ← l_init
   // Perform energy minimization
4  cycles ← 0
5  repeat
6  |  success ← false
7  |  foreach α ∈ L
8  |  |  ℓ̂ = arg min E(ℓ'), ∀ℓ' ∈ M_ℓ
9  |  |  if E(ℓ̂) < E(ℓ)
10 |  |  |  ℓ ← ℓ̂
11 |  |  |  success ← true
12 |  cycles ← cycles + 1
13 until success = false or cycles = maxCycles
```

Having formulated individual expansion moves as binary subproblems, we now return our attention to the Expansion Moves algorithm. Its listing is given in Algorithm 2.4.1. We shall call the execution of lines 5–13 a cycle and an execution of lines 7–11 an iteration. Starting from a constant labeling by an arbitrarily chosen label (lines 1–3), the algorithm cycles through labels in a fixed or a random order. In each iteration, the optimal expansion on a label is computed with the method outlined in the above (line 8), and if the energy of the resulting configuration is strictly inferior (line 9), the current configuration is updated (line 10). A cycle is successful if a strictly better configuration is found in one of its iterations. The algorithm stops after the first unsuccessful cycle or if the required maximum number of cycles are executed.

The Expansion Moves algorithm 2.4.1 is guaranteed to terminate since every iteration retains the configuration that decreases the value of the energy. [BVZ01] mention that for an energy formulation which is independent of the number of variables, the algorithm performs $O(V)$ cycles, where V is the set of variables. Therefore, the running time of the algorithm is $O(VL)$. However, the authors report that in practice the algorithm stops after a few cycles. This observation is confirmed by our experimental setup as well. A precise description of the algorithm's total running time requires taking into account the running time of the minimum cut computation (line 8) for which a maximum flow algorithm is applied. For example, using a straightforward implementation of the Push-relabel method with the FIFO selection rule [GT88] which runs in $O(V^3)$ time, the total running time of the Expansion Moves algorithm becomes $O(V^4L)$.

2.5 Conclusions

In this chapter, we have laid the theoretical foundation of our approach that we shall elaborate in subsequent chapters formulating image segmentation as a Bayesian labeling problem which is solved by minimizing a posterior probability energy, described in terms of likelihood and Gibbs distribution energies, to obtain the optimal segmentation with respect to user-provided evidence, constraints and a problem-specific prior.

We have addressed energy minimization methods discussing Graph Cut algorithms in exact binary and approximate multilabel optimization as well as the underlying maximum flow techniques. We choose the Expansion Moves algorithm for minimizing multilabel segmentation energies which finds a local solution within a known factor of the global minimum. Having a bound on local minima helps us to explain whether an inadequate solution is due to a bad energy definition or a poorly performing minimizer.

Even though good local minima correspond to good quality solutions, the quality of a labeling result in segmentation is usually assessed with respect to a ground-truth labeling. As reported, the global minimum, when known, is not more accurate than the ground-truth, and the energy of the ground-truth is frequently higher than that of the global minimum, therefore the focus is not on the optimization of energies, but on finding more expressive energy models. This is the objective of the following chapter.

Chapter 3

A structural prior model for Graph Cut segmentation

3.1 Introduction

Most successful segmentation methods usually incorporate some image or domain-specific prior information. Medical image segmentation methods in particular should ideally be able to incorporate a wide range of anatomical and physiological prior information to ensure the consistency of the result with respect to anatomical properties and variabilities thereof. From the standpoint of Graph Cut optimization of energies of label configurations, prior information can be regarded as a set of constraints on labels defined by a reference model. In Section 2.2, we have encountered unary and pairwise constraints stemming from the energy potentials of a prior distribution expressed as a Markov random field (MRF) of label configurations with respect to the image graph. Such prior information is combined with data-derived likelihood and user constraints in an energy minimization framework to infer the optimal segmentation according to these criteria.

As mentioned in Section 2.2.2, in image segmentation, pairwise prior interaction potentials have to encourage a piecewise-constant discontinuity-preserving labeling. The most widely used such prior, called the Potts model (2.12), presumes the presence of a single type of boundary in the image. With the assumption of equal violation of piecewise constance, it penalizes attributions of pairs of unequal labels to neighboring variables equally while incurring no penalty for the identical labeling of such pairs. This assumption obviously does not hold in multiobject segmentation where multiple boundaries exist between the objects.

In this chapter, we investigate extensions of the Potts prior model to account for multiple boundaries in multiobject segmentation which incur multiple levels of penalization and furthermore allow the minimization of resulting energies by the Expansion Moves Graph Cut algorithm discussed in Section 2.4.2. We first consider a simple structure adjacency prior by adding a second level of penalization to the Potts model [JKD+12]. Due to the submodularity requirement, Expansion Moves optimization can

be applied only to a restricted form of this prior where the penalty of nonadjacency is at most twice that of adjacency. We outline a submodular extension of this prior for simple linear label layout models to penalization ratios greater than two by the introduction of auxiliary labels into the model. The main contribution of this chapter, motivated by a special case of the aforementioned submodular extension of the adjacency prior, is presented last. The vicinity prior, as we term it, is an intrinsically submodular prior model which incurs multiple levels of penalization, $|L|$ at most, and captures the spatial configuration of structures in multiobject segmentation. It is defined as shortest-path pairwise constraints on a graph model of interobject adjacency relations [KVDP13].

The following section reviews relevant literature and suggests a taxonomy of prior introduction approaches in Graph Cut segmentation.

3.2 Prior information in Graph Cut segmentation

Many forms of prior information have been introduced in Graph Cut segmentation for different applications. We broadly classify the most relevant of such approaches in three categories according to the nature of introduced prior.

a) Location prior, formulated as spatially-varying prior probability maps of individual structure locations in the image space. Such prior information is usually encoded by the unary constraints of the prior energy (2.3) to indicate the label preference of individual variables [STAG06, KRBT08, LPPS12] and sometimes via its pairwise constraints to favor the placement of the segmentation boundary near the object's contour [FZ05]. In simplest cases, probabilistic maps of location are defined as a distance transform of a shape template [FZ05, KRBT08]. In medical applications, probabilistic atlases of tissue location are usually constructed by averaging label estimates for each pixel over a ground-truth dataset registered to some anatomical space [STAG06, LPPS12]. Naturally, such probabilistic maps have to be learned in advance and registered to an image before it is segmented. Registration cannot trivially handle topological and geometrical changes, due to pathologies for example, and is often computationally intensive.

b) Shape prior, where explicit or implicit shape constraints are used to match the segmented object with predefined or learned shapes. [VM08] define an affine-invariant shape similarity measure and introduce it in a sequential multiobject Graph Cut segmentation framework. [MRT07] first learn a statistical shape model from a training set via kernel principal component analysis, then use the projection of an initial segmentation in the learned shape space as the shape prior for subsequent segmentations in an iterative process. In their Graph Cut framework for the optimization of hypersurface functionals, [KB05] show how to set unary constraints to favor cuts that maximize the flux of distance transform gradients for a shape's contour or its medial axis. Shape-prior segmentation approaches entail shape location initialization and alignment mechanisms, and often require model learning.

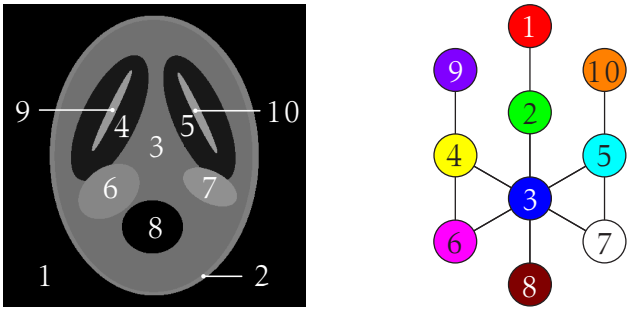
c) Spatial configuration prior, the use of which usually arises in the context of multiobject segmentation, where usually pairwise relations define interobject geometric and spatial relationships of the type "object a is above object b and next to object c". The

use of such prior information has appeared in the context of part-based object detection [WS06] and scene layout estimation and segmentation where spatial relationships are either predefined on a limited number of classes [LVS10] or learned from annotated training data [WS06, LLB+11]. [DB09] segment multiple or compound objects encountered in relative positions of containment and exclusion, also introducing graph-edge constraints on preferred distances between the boundaries of regions. Some useful configurations however cannot be represented by definitions therein, such as the inclusion of an adjacent pair of objects in a third.

There are a number of spatial configuration prior formulations closely related to our work which have appeared outside Graph Cut segmentation literature. In a theoretical framework similar to ours, although following a different numerical optimization scheme, [SSLS+01, CCB+05] use a simple three-level piecewise-constant prior model for tissue classification in brain MR images. According to a brain tissue model, it encourages the identical classification of pairs of neighboring pixels, allowing different class adjacency patterns which are consistent with the model while penalizing adjacencies which are not allowed to occur. [GRC+08] proposes a relative location prior represented by probabilistic maps of interclass offset preferences for pairs of classes. Such probabilistic maps are learned in advance from training data and are used to define a local relative location feature which is introduced into the segmentation energy via unary constraints. In [SBZ+09], pairwise geometric relationships are defined by an anatomical network on easily detectable body landmarks and used in the organ detection phase of a sequential segmentation framework for thoracic and abdominal organs in whole-body CT scans. [OYH+12] addresses the same problem as [SBZ+09] with an automatic two-stage method whereby once relatively stable organs are identified and segmented, organ interrelations are used to proceed with the segmentation of variable organs by fitting previously learned statistical organ atlases and shape models. Organ interrelations are defined according to canonical correlation analysis and represented by a directed graph. Intuitively, these relations represent the degree of influence of an organ's surface upon another. A generic, graph-based scene model representing spatial information explicitly is also employed by [FAB12] in the problem of segmenting multiple subcortical structures in brain MR images. Fuzzy representations of distance, adjacency and spatial directionality relations are introduced to account for model imprecisions and variabilities. The segmentation algorithm, however, is sequential and requires error detection and segmentation order rearrangement strategies.

3.3 An adjacency prior

Let us first consider the image in Figure 3.1a. It shows our synthetic phantom featuring 10 structures of different size, topology and adjacency patterns, labeled 1 to 10 as indicated. More details can be found in Section 5.3 where we present evaluation results on a dataset generated from the phantom by introducing variable Gaussian noise. In the remaining of this chapter, we shall use this phantom as a reference model to illustrate concepts as we develop definitions of prior models.



(a) (b)

Figure 3.1: (a) Our Synthetic phantom and (b) the graph model of adjacency relations on its structures.

In all our formulations, we shall rely on pairwise constraints, expressed by pairwise energy terms $V_{i,j}(\ell_i, \ell_j)$ in (2.10), to encode prior information on pairwise interactions between labels assigned to pairs of neighboring variables encouraging the spatial consistency of labeling with respect to a problem-specific reference model. Assuming a piecewise-constant prior model, the simplest manifestation of such consistency is that the spatial layout of image regions labeled differently should reflect the local spatial constraints defined by the prior model. In particular, given a pair of labels for which no close-range spatial interaction is described by the model, no pair of image regions labeled with this pair of labels should appear next to each other in the labeled image.

In a segmentation task of an image generated from the phantom in Figure 3.1a, the use of the Potts prior model incurs equal penalty for all but pairs of identical labels. For example, given the adjacent and the nonadjacent pairs $\{9,4\}$ and $\{9,8\}$ respectively, the Potts model implies that attributions of the corresponding pairs of nonidentical labels to pairs of neighboring variables violate the spatial consistency of labeling equally, since $\forall i,j \in \mathcal{S}$, $V_{i,j}(9,8) = V_{i,j}(9,4) = 1$.

In similar spirit to [SSLS+01, CCB+05], we extend the Potts prior by an additional constraint level to penalize label assignments which violate layout consistency with respect to structure adjacency relations as described by the reference model. Such a formulation is able to distinguish pairings of labels corresponding to adjacent and non-adjacent structures in the model and penalizes them differently while preserving the piecewise constance property of the prior since $V(a,a) = 0, \forall a \in L$. For the example just illustrated, the effect would be $\forall i,j \in \mathcal{S}$, $V_{i,j}(9,8) > V_{i,j}(9,4)$.

Let $\mathcal{R}$ be the set of symmetric binary relations on pairs of distinct labels, $\mathcal{R} = \{\mathrm{r} \mid a\,\mathrm{r}\,b, a,b \in L, a \neq b\}$. Such binary relations can represent many types of relationships between pairs of objects, such as adjacency, distance, area of common surface, etc. At present, we let relations $\mathrm{r} \in \mathcal{R}$ represent structure adjacency. A structure a is considered to be adjacent to another b, denoted as $a\,\mathrm{r}\,b$, if they share a boundary (in 3D, a surface). We define the adjacency prior as follows:

$$V_{i,j}\left(\ell_i, \ell_j\right) = \begin{cases} 0, & \ell_i = \ell_j \quad (3.1a) \\ \gamma_{soft}, & \exists r \in \mathcal{R}, \ell_i r \ell_j \quad (3.1b) \\ \gamma_{hard}, & \nexists r \in \mathcal{R}, \ell_i r \ell_j \quad (3.1c) \end{cases}$$

where γ_{soft} and γ_{hard} correspond to adjacency and nonadjacency penalties respectively and $\gamma_{hard} > \gamma_{soft}$. Ideally, $\gamma_{hard} \to \infty$, however, some energy terms would then violate the submodularity condition (2.14), namely those falling on the left-hand side of the inequality. One possible remedy to this problem is the "truncation" of nonsubmodular terms [RKKB05, WS06, DB09], which is the modification of energy terms on either side of an unsatisfied submodularity condition so that the inequality holds. As pointed out by [RKKB05], the Expansion Moves algorithm with truncated energy terms is a valid energy minimization technique for arbitrary functions and is guaranteed to converge. However, individual algorithm steps are no longer guaranteed to find the optimal labeling with respect to expansion moves and the final labeling does not necessarily have the strong local minimality property. They point out that the technique gives reasonable results as long as the number of nonsubmodular terms remains small. Another solution is using an efficient arbitrary energy minimization algorithm, like the TRW-S algorithm [Kol06], which however does not have a bound on local minima.

The adjacency prior as defined by (3.1) is submodular when $\gamma_{hard} \leq 2\,\gamma_{soft}$, with the largest penalization ratio at equality. Clearly, this prior formulation is essentially limited to definitions where the penalty of nonadjacency is at most twice as much as the penalty of adjacency so that the prior is submodular and the energy (2.10) can be minimized via the Expansion Moves algorithm. In other words, it is impossible to penalize nonadjacencies by some large constant without penalizing adjacencies half as severely. In the following section, we outline a submodular extension of this prior for simple linear label layout models to penalization ratios greater than two by the introduction of auxiliary labels into the model.

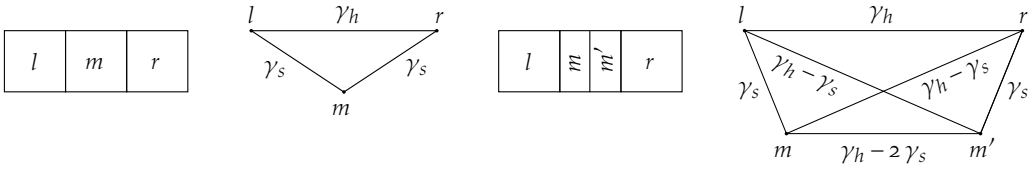
3.3.1 Arbitrary penalization submodular formulations

Consider the simple three-label model illustrated in Figure 3.2a with the label set $L = \{l, m, r\}$. The adjacency prior for this model is defined according to (3.1) as follows:

$$\begin{aligned} \forall a \in L,\, V(a,a) &= 0, \\ V(l,r) = V(l,r) &= \gamma_{hard}, \\ V(l,m) = V(m,l) = V(m,r) = V(r,m) &= \gamma_{soft}. \end{aligned} \quad (3.2)$$

Let us suppose that $\gamma_{hard} > 2\,\gamma_{soft}$. Consider the set of inequalities of the submodularity condition for all permutations of labels in L. Because of symmetric pairwise definitions in (3.2), there are three such inequalities where each pair of distinct labels in $\{l, m, r\}$ falls on the left-hand side of an inequality exactly once. The inequality which does not hold for $\gamma_{hard} > 2\,\gamma_{soft}$, is the following:

$$V(m,m) + V(l,r) \leq V(m,r) + V(l,m). \quad (3.3)$$



(a) (b) (c) (d)

Figure 3.2: (a) 3-label structure adjacency model, (b) geometrical representation of the adjacency prior defined on the model, (c) the model after shadowing the label m, (d) geometrical representation of the extended adjacency prior on the new model. Note that trapezoid side lengths are set for illustration clarity and do not precisely correspond to indicated quantities.

To satisfy (3.3), we can either increase one of $V(m,r)$ or $V(l,m)$, or decrease $V(m,m)$, as pointed out by [RKKB05], or use an arbitrary energy minimizer, for example [Kol06].

Without modifying the prior definition, in particular the ratio $\gamma_{hard}/\gamma_{soft}$, we write the violated inequality (3.3) as two inequalities that can be satisfied on the set of labels $L \cup m'$. We call m' the shadow label of m representing the same image structure. Consider the graphical representation of the triangle inequality in Figure 3.2b, where triangle vertices correspond to labels l, m and r and lengths of its sides to prior penalties such that $\forall a, b \in L$, $\overline{ab} = V(a,b)$. Note that the illustration in 3.2b corresponds to $\gamma_{hard} < 2\,\gamma_{soft}$ and that the geometrical representation for $\gamma_{hard} = 2\,\gamma_{soft}$ corresponds to collinear points. In the following, we shall denote γ_{hard} and γ_{soft} shortly as γ_h and γ_s.

Our intuition is pulling the vertices l and m of the triangle $\triangle lrm$ apart so that the side $\overline{lr}$ is longer, in particular $\overline{lr} > \overline{lm} + \overline{rm}$. As illustrated in Figure 3.2d, this splits the vertex m and results in two triangles, $\triangle lrm$ and $\triangle lrm'$, according to which we rewrite the violated inequality (3.3) as follows using line-segment notation:

$$\overline{lr} \leq \overline{lm} + \overline{mr} \tag{3.4a}$$

$$\overline{lr} \leq \overline{lm'} + \overline{m'r} \tag{3.4b}$$

where $\overline{lr} = \gamma_h$ and $\overline{lm} = \overline{m'r} = \gamma_s$ and $\gamma_h > 2\,\gamma_s$. We shall derive bounds on unknowns $\overline{mr}$, $\overline{lm'}$ and $\overline{mm'}$.

From (3.4a), $\overline{mr} \geq \overline{lr} - \overline{lm}$, that is, $\overline{mr} \geq \gamma_h - \gamma_s$. Also from $\triangle lrm$, $\overline{mr} \leq \overline{lr} + \overline{lm}$, that is, $\overline{mr} \leq \gamma_h + \gamma_s$. Thus:

$$\gamma_h - \gamma_s \leq \overline{mr} \leq \gamma_h + \gamma_s. \tag{3.5}$$

Similarly, from (3.4b) and $\triangle lrm'$ we find:

$$\gamma_h - \gamma_s \leq \overline{lm'} \leq \gamma_h + \gamma_s. \tag{3.6}$$

We now consider $\overline{mm'}$. From $\triangle lmm'$ we have, $\overline{mm'} \leq \overline{lm} + \overline{lm'}$, $\overline{mm'} \leq \gamma_s + \gamma_h - \gamma_s = \gamma_h$ assuming lower bound on $\overline{lm'}$. We also have $\overline{mm'} \geq \overline{lm'} - \overline{lm}$, that is, $\overline{mm'} \geq \gamma_h - 2\,\gamma_s$ under the same assumption. Hence:

$$\gamma_h - 2\,\gamma_s \leq \overline{mm'} \leq \gamma_h. \tag{3.7}$$

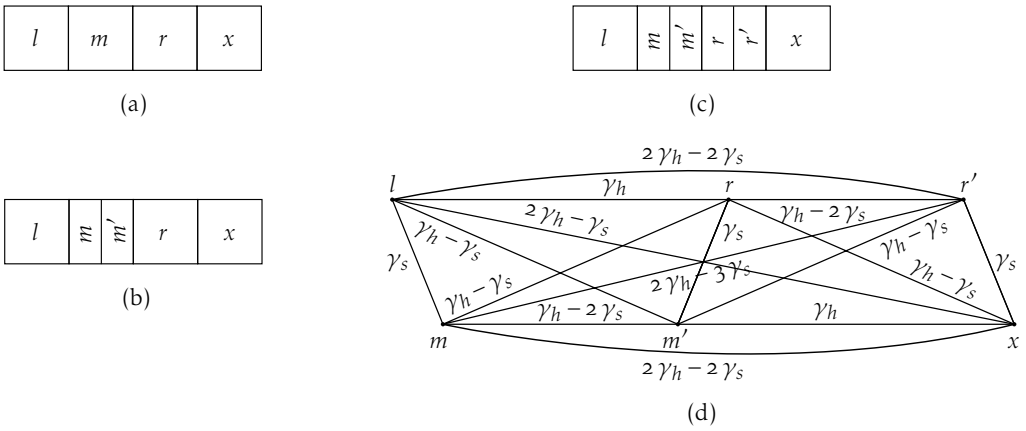


Figure 3.3: (a) 4-label structure adjacency model, (b) the model after shadowing the label m and (c) after shadowing the label r', (d) geometrical representation of the extended adjacency prior on the new model. Note that line segment lengths are set for illustration clarity and do not precisely correspond to indicated quantities. Segments $\overline{lr'}$ and $\overline{mx}$ are drawn curved for the same purpose.

Since m and m' represent the same structure, the upper bound γ_h, equal to $\overline{lm}$ which is the highest penalty in the model, is too severe. Therefore, we shall assume lower bound on $\overline{mm'}$. This completes the construction of the extended adjacency prior that accounts for arbitrary ratios $\gamma_{hard}/\gamma_{soft}$ for the simple three-structure model in Figure 3.2a. Its definition is summarized in Figure 3.2d. Note that for a ratio of two, $\gamma_{hard} = 2\,\gamma_{soft}$, the extended definition gives the original (3.1). In particular, vertices m and m' coincide and $\overline{mm'} = 0$, and the diagonals of the trapezoid $lrm'm$ in Figure 3.2d coincide with the sides lm and rm of the triangle $\triangle lrm$ in Figure 3.2b with $\overline{lm'} = \overline{rm} = \gamma_{soft}$.

Now we shall consider a more elaborate four-structure model, illustrated in Figure 3.3a, defined on $L = \{l, m, r, x\}$ by the addition of a label x adjacent to r to the previous three-label model in Figure 3.2a. The adjacency prior definition for this model follows (3.2), with the following addition:

$$\begin{aligned} V(r,x) = V(x,r) &= \gamma_{soft}, \\ \forall a \in \{l,m\},\ V(a,x) = V(x,a) &= \gamma_{hard}. \end{aligned} \tag{3.8}$$

Once again, we assume that $\gamma_{hard} > 2\,\gamma_{soft}$. Let us consider the triple $\{l, m, r\}$ first. Shadowing the label m gives the five-label construction shown in Figure 3.3b. The definition of the extended adjacency prior for the set $\{l, m, m', r\}$ follows the derivation discussed above which is shown in Figure 3.2d. Furthermore, triples involving labels $\{l, m, r, x\}$ satisfy submodularity inequalities. However, the condition does not hold for the triple $m'rx$, as expected. As before, we shadow the label r producing the six-label model illustrated in Figure 3.3c. The geometrical representation of the extended adjacency prior for this model is given in Figure 3.3d. Definitions of trapezoids $lrm'm$ and $xm'rr'$, each corresponding to an extended adjacency prior on a three-structure model, are identical to that in Figure 3.2d. There remains to derive bounds on $\overline{lr'}$, $\overline{xm}$, $\overline{lx}$ and $\overline{mr'}$.

From $\triangle lrr'$ we have, $\overline{lr'} \leq \overline{lr} + \overline{rr'}$, that is, $\overline{lr'} \leq \gamma_h + \gamma_h - 2\gamma_s = 2\gamma_h - 2\gamma_s$. From the same triangle, we also have $\overline{lr'} \geq \overline{lr} - \overline{rr'}$, that is, $\overline{lr'} \geq \gamma_h - \gamma_h + 2\gamma_s = 2\gamma_s$. Therefore:

$$2\gamma_s \leq \overline{lr'} \leq 2\gamma_h - 2\gamma_s. \tag{3.9}$$

Similarly, from $\triangle xmm'$ we find that:

$$2\gamma_s \leq \overline{xm} \leq 2\gamma_h - 2\gamma_s. \tag{3.10}$$

Upper bounds of (3.9) and (3.10) can also be obtained from $\triangle lr'm'$ and $\triangle xrm$ respectively. Let us derive bounds on $\overline{lx}$ and $\overline{mr'}$. From $\triangle lxr'$ we have, $\overline{lx} \leq \overline{lr'} + \overline{r'x}$, that is, $\overline{lx} \leq 2\gamma_h - 2\gamma_s + \gamma_s = 2\gamma_h - \gamma_a$ assuming upper bound on $\overline{lr'}$. We also have, $\overline{lx} \geq \overline{lr'} - \overline{r'x}$, which is, $\overline{lx} \geq 2\gamma_h - 2\gamma_s - \gamma_s = 2\gamma_h - 3\gamma_s$. Hence:

$$2\gamma_h - 3\gamma_s \leq \overline{lx} \leq 2\gamma_h - \gamma_a. \tag{3.11}$$

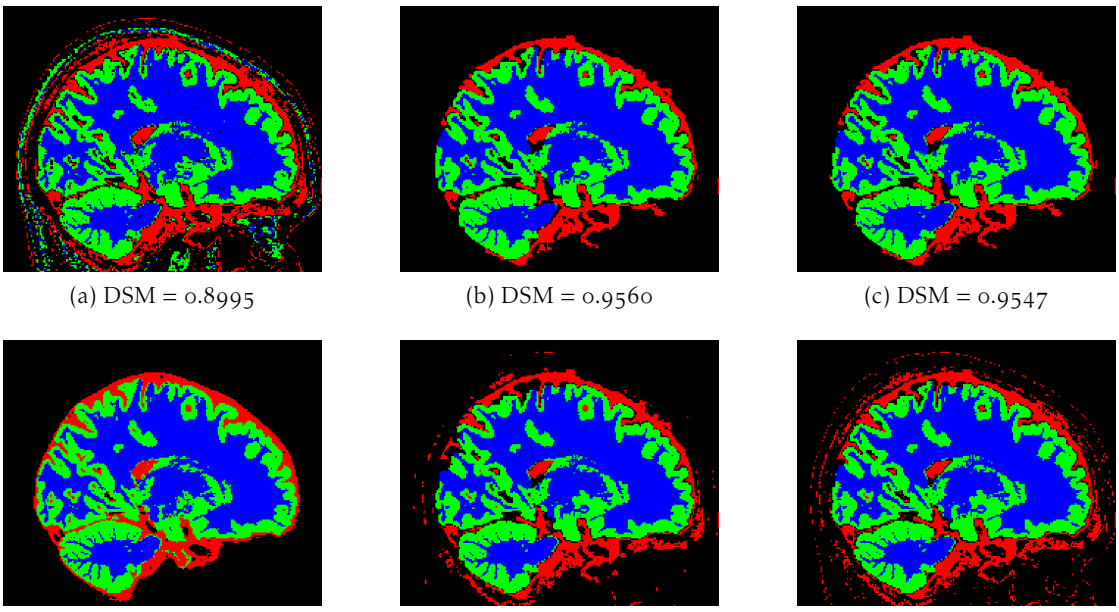
From $\triangle mrr'$ we have, $\overline{mr'} \leq \overline{mr} + \overline{rr'}$, which is, $\overline{mr'} \leq \gamma_h - \gamma_s + \gamma_h - 2\gamma_s = 2\gamma_h - 3\gamma_a$. We also have, $\overline{mr'} \geq \overline{mr} - \overline{rr'}$, that is, $\overline{mr'} \geq \gamma_h - \gamma_s - \gamma_h + 2\gamma_s = \gamma_s$. Hence:

$$\gamma_s \leq \overline{mr'} \leq 2\gamma_h - 3\gamma_a \tag{3.12}$$

which completes the construction of the extended adjacency prior accounting for arbitrary ratios $\gamma_{hard}/\gamma_{soft}$ for the four-structure linear model in Figure 3.3a. Figure 3.3d summarizes the definition.

In order to understand the practical merit of extended versus basic adjacency prior models, we conduct an experiment of brain tissue segmentation on a single simulated 3D MR image from the second-generation BrainWeb simulated MRI dataset. The dataset is described in Section 5.4.2. To test the four-label extended adjacency prior defined above, we formulate Graph Cut segmentation on the label set $L = \{$"BKG", "CSF","GM","WM"$\}$ representing background, cerebrospinal fluid, gray and white matters respectively. The background comprises all nonbrain tissues in addition to air. The spatial layout of structures in L largely corresponds to that of the four-label model illustrated in Figure 3.3a. We estimate likelihood distributions for the last three labels from user-supplied "seeds" as described in Section 2.2.1, using distributions identical to those of the original label for shadow labels and a uniform distribution for the background label. We measure the quality of segmentation with the overall Dice similarly metric (DSM) (5.2) which indicates the agreement between the segmentation and the ground-truth as a mean weighted measure of individual-label DSM measures.

Figure 3.4 gives results of qualitative and quantitative evaluation using Potts, basic adjacency prior and its extension for three ratios $\gamma_{hard}/\gamma_{soft}$ equal to 2, 3 and 5. Presented cross-sections and DSM figures suggest that there are no real advantages of using the extended adjacency prior compared to the basic adjacency prior on this instance of segmentation described by the four-label model. Furthermore, segmentation quality deteriorates with increasing gap between the two penalty levels and false positives of the label class "CSF" begin to appear, which however obey adjacency constraints.



(a) DSM = 0.8995 (b) DSM = 0.9560 (c) DSM = 0.9547

(d) (e) DSM = 0.9467 (f) DSM = 0.9368

Figure 3.4: Qualitative and quantitative comparison of best brain tissue segmentations, as measured by overall DSM, in a second-generation BrainWeb simulated MR image. Sagittal cross-sections show segmentations using 5 different prior formulations: (a) Potts, (b) adjacency prior (3.1) with $\gamma_{hard} = 2\,\gamma_{soft}$ and arbitrarily-penalization submodular adjacency prior for the 4-structure model with (c) $\gamma_h = 2\,\gamma_s$, (e) $\gamma_h = 3\,\gamma_s$ and (f) $\gamma_h = 5\,\gamma_s$. (d) Shows the ground-truth. Color legend; black: background, red: cerebrospinal fluid, green: gray matter, blue: white matter.

It is interesting to note that for $\gamma_{hard}/\gamma_{soft} = 2$, the extended adjacency prior gives a definition that accounts for relative structure distance for the linearly ordered four-label model:

$$\begin{aligned} V(l,m) = V(m,r) = V(r,x) &= \gamma_{soft} \\ V(l,r) = V(m,x) &= 2\,\gamma_{soft}, \\ V(l,x) &= 3\,\gamma_{soft}. \end{aligned} \tag{3.13}$$

Given the complexity of extended adjacency prior constructions, we seek a simpler specification of the idea of stepwise penalization which would furthermore be easier to define for more complex nonlinear models, counting several tens of labels.

3.4 A shortest-path formulation: the vicinity prior

The adjacency prior discussed in Section 3.3 ensures the spatial consistency of labeling by penalizing pairings of labels corresponding to adjacent and nonadjacent pairs of structures differently (3.1). The formulation however makes the assumption that pairings corresponding to nonadjacent structures are equally unfit, independently of the mutual distance between structures. Higher penalization for farther away pairs of structures while maintaining the submodularity property of the prior is desired.

We have observed that a special case of the extended adjacency prior, corresponding to $\gamma_{hard} = 2\,\gamma_{soft}$, accounts for relative structure distance for the simple linearly ordered four-label model 3.3a incurring multiple levels of penalization (3.13). In this section, we present a different formulation of the idea of variable penalization according to stepwise relative distance which can be furthermore specified easily for much more complex nonlinear models defined on an arbitrary number of labels.

Consider the phantom in Figure 3.1a again, and observe the layout of structures "9", "4", "6" and "8". Notice how the structure labeled "9" is nested in the structure "4" which, in turn, is adjacent to the structure "6". Due to this adjacency pattern, it makes sense to encourage the assignment of the pair of labels $\{9,4\}$ while penalizing the attribution of the pair $\{9,6\}$ to neighboring variables in a segmentation task of an image generated from the phantom. Hence, $\forall i,j \in \mathcal{S}$, $V_{i,j}(9,6) > V_{i,j}(9,4)$. Furthermore, since the structure "8" is farther still, $V_{i,j}(9,8) > V_{i,j}(9,6) > V_{i,j}(9,4)$ should hold. By comparison, the adjacency prior penalizes nonadjacent pairs $\{9,6\}$ and $\{9,8\}$ equally implying that they are equally unfit.

Recall that $\mathcal{R}$ is the set of symmetric binary relations on pairs of distinct labels representing adjacency relations of corresponding structures. Assume furthermore that each relation $\mathrm{r} \in \mathcal{R}$ has a corresponding characteristic function $\hat{\mathrm{r}}$ which gives a quantitative measure for the relationship r represents, $\hat{\mathrm{r}}\colon L \times L \setminus \{(a,a)\} \mapsto \mathbb{R}^+$.

The set of relations $\mathcal{R}$ can be equivalently represented by a weighted undirected graph $\mathcal{A} = \langle L, W\rangle$ without self-loops[1] on labels L, where the set of edges is given by $W = \{\{a,b\} \mid \exists \mathrm{r} \in \mathcal{R},\, a\,\mathrm{r}\,b,\, a \neq b\}$. Edge weights are defined by $w(\{a,b\}) = \hat{\mathrm{r}}(a,b)$, such that $w(\{a,b\}) = \infty$, $\nexists \mathrm{r} \in \mathcal{R}$, $a\,\mathrm{r}\,b$.

Define a path of length[2] k from a vertex l_0 to another l_k in $\mathcal{A}$ by a sequence $\pi = (l_0, l_1, \ldots, l_k)$. We say that the vertex l_k is reachable from the vertex l_0 via the path π and denote it as $l_0 \overset{\pi}{\rightsquigarrow} l_k$. Define the weight of a path $w(\pi)$ as the sum of the weights of its constituent edges $w(\pi) = \sum_{i=0}^{k-1} w(\{l_i, l_{i+1}\})$. The shortest-path weight from a vertex a to another b is defined by:

$$\varpi(a,b) = \begin{cases} \min w(\pi), & \forall \pi,\, a \overset{\pi}{\rightsquigarrow} b \qquad (3.14a) \\ \infty, & \text{otherwise} \qquad (3.14b) \end{cases}$$

where (3.14b) corresponds to the case where there is no path π through which the vertex b can be reached from the vertex a. Note that the shortest path from a to b is defined as any path π with weight $w(\pi) = \varpi(a,b)$ and that the shortest-path weight from a vertex to itself is $\varpi(a,a) = 0$.

Given the graph $\mathcal{A}$, we define the pairwise prior term in (2.10) as:

$$V_{i,j}\left(\ell_i, \ell_j\right) = \varpi(a,b), \quad \ell_i = a, \ell_j = b. \qquad (3.15)$$

There are several algorithms to calculate shortest-path weights the choice of which depends on the underlying graph and the application [CLRS09]. For weighted graphs with positive edge weights, shortest paths from a given vertex to all others can be found

[1] A self-loop is an edge from a vertex to itself.

[2] The length of a path is equal to the number of edges on it.

via Dijkstra's algorithm with a running time of $O(L^2)$. If the graph is sparse and every vertex can be reached from the starting vertex, the running time becomes $O(W \log L)$ when the algorithm's priority queue is implemented with a binary heap. For graphs in which each edge has unit weight, shortest paths from a given vertex to all others can be found via the breadth-first search (BFS) algorithm with a running time of $O(L + W)$.

In our formulation, relations $\mathrm{r} \in \mathcal{R}$ represent structure adjacency. For two such structures a and b, the adjacency relation's characteristic function is given by:

$$\hat{\mathrm{r}}(a,b) = \begin{cases} 1, & \exists \mathrm{r} \in \mathcal{R},\ a\,\mathrm{r}\,b \quad (3.16a) \\ \infty, & \text{otherwise.} \quad (3.16b) \end{cases}$$

This results in a connected undirected graph $\mathcal{A}$ with unit-weight edges between adjacent vertices, where the degree of each vertex $l \in L$ is equal to the number of structures adjacent to the structure associated with it. Shortest paths from a vertex to all others, conveniently calculated by an application of BFS to the graph $\mathcal{A}$, define the vicinity of the associated structure with respect to all others, which is introduced in the energy function of segmentation (2.10) via pairwise terms (3.15). Clearly, the graph $\mathcal{A}$ constitutes a prior model of explicit spatial configuration of objects. To illustrate, the graph prior model of our synthetic phantom in Figure 3.1a is given in Figure 3.1b. Note that the graph $\mathcal{A}$ is not limited to unit-weight edges. It can encode any quantity that accumulates linearly along a path that we would want to minimize.

The vicinity prior as defined by (3.15) is intrinsically submodular. Therefore, the Expansion Moves algorithm can by applied to minimize segmentation energies using this prior without resorting to model truncation, unlike [RKKB05, WS06, DB09]. The following theorem establishes the submodularity property of the vicinity prior.

Theorem 3.4.1 (Submodularity of vicinity prior). *Given the undirected weighted graph* $\mathcal{A} = \langle L, W \rangle$ *and the prior definition* $V_{i,j}(a,b) = \varpi(a,b)$*, the inequality* $V_{i,j}(a,a) + V_{i,j}(b,c) \leq V_{i,j}(a,c) + V_{i,j}(b,a)$ *holds* $\forall i,j \in \mathcal{S}$ *for any* $a,b,c \in L$*.*

Proof. By definition, $V_{i,j}(a,a) = \varpi(a,a) = 0$. Substituting (3.15) in (2.14), we try to prove

$$\varpi(b,c) \leq \varpi(b,a) + \varpi(a,c), \quad \forall a,b,c \in L. \quad (3.17)$$

First assume that the graph $\mathcal{A} = \langle L, W \rangle$ is connected, that is $\forall a,b \in L, \exists \pi, a \overset{\pi}{\rightsquigarrow} b$. Let π_{ba} be the shortest path with weight $w(\pi_{ba}) = \varpi(b,a)$, similarly, let π_{ac} be the shortest path with weight $w(\pi_{ac}) = \varpi(a,c)$. The shortest path π_{bc} either passes by a, that is, $\pi_{bac} = (b,\ldots,a,\ldots,c)$, in which case its weight is $w(\pi_{bac}) = w(\pi_{ba}) + w(\pi_{ac})$, hence $\varpi(b,c) = \varpi(b,a) + \varpi(a,c)$, or it passes by a vertex $q \in L$ not on π_{bac} through a different path $\pi_{bqc} = (b,\ldots,q,\ldots,c)$ with weight $w(\pi_{bqc}) < w(\pi_{ba}) + w(\pi_{ac})$, therefore $\varpi(b,c) < \varpi(b,a) + \varpi(a,c)$.

If the graph $\mathcal{A}$ is unconnected, infinite shortest-path weight might appear on the right-hand side of (3.17) when either or both b and c are unreachable from a, and the inequality holds. However, infinite shortest-path weight cannot appear only on the left-hand side of (3.17), because if $b \overset{\pi_{ba}}{\rightsquigarrow} a$ and $a \overset{\pi_{ac}}{\rightsquigarrow} c$ then $\exists \pi_{bac} = (b,\ldots,a,\ldots,c)$ hence $b \overset{\pi_{bac}}{\rightsquigarrow} c$ with $w(\pi_{bac}) = w(\pi_{ba}) + w(\pi_{ac})$ thus completing the proof. □

3.5 Discussion and conclusions

In this chapter, we have investigated discontinuity-preserving piecewise-constant submodular prior formulations for multiobject segmentation which, unlike the Potts prior model, incur multiple levels of penalization accounting for multiple types of boundaries. Starting from the Potts model, we have devised a simple structure adjacency prior by introducing a second level of penalization for nonadjacent structures. The submodularity requirement restricts this model to definitions where the penalty of nonadjacency is at most twice that of adjacency. We have outlined submodular extensions of this prior for linear spatial layout models to penalization ratios greater than two by the introduction of auxiliary labels into the model. A special case of the extended formulation gives rise to a variable penalization scheme accounting for relative structure distance on the linear spatial layout models. We proposed a graph-based formulation of this scheme which can be specified easily for complex nonlinear models defined on an arbitrary number of labels. The vicinity prior, as we name it, is defined as shortest-path pairwise constraints on a graph model of interobject adjacency relations. It is intrinsically submodular and incurs $|L|$ levels of penalization at most, furthermore capturing the spatial configuration of structures in multiobject segmentation. In Chapter 5, we give the results of quantitative and qualitative evaluation of Potts, adjacency and vicinity prior-based segmentation on synthetic, simulated and real medical images.

While theoretically interesting, we did not observe any practical advantages of the extended compared to the basic adjacency prior model in the problem of brain tissue segmentation. One explanation is that the Expansion Moves algorithm has a worse bound on local minima (2.13) for the extended model compared to that of the basic model, in addition to the labeling problem being defined in a higher-dimensional space in this case. At any rate, further empirical evidence is necessary to reach a conclusion on the practical merit of the extended adjacency prior model.

Compared to shape prior-based approaches, structural prior models capturing the spatial configuration of objects are quite robust to shape deformations since relative positions of objects remain largely stable. Graph models describing such priors are fairly easy to define at an arbitrary level of detail from simple specifications of spatial relationships on pairs of objects, and can be generated from existing anatomical models for medical applications in particular. Interobject adjacency patterns can be learned from training data or several model variants thus capturing model variabilities in which case continuous graph-edge weighs may be used, for example, to represent degrees of adjacency. Pairwise spatial configuration prior formulations learned from data are frequently nonsubmodular, therefore authors resort to model truncation to apply Graph Cut minimization [WS06] or use arbitrary-energy minimizers [GRC+08, LLB+11] thus sacrificing the accuracy of the model in the former and the quality of local minima in the latter case. Compared to such learning-based approaches, the vicinity prior formulation has the advantage that of all pairings on $L \times L$ only those corresponding to structures that exhibit adjacency in the dataset need to be explicitly specified or learned; on the remaining pairs, the prior is defined as shortest-path lengths on the prior graph.

Chapter 4

Reducing Graph Cut complexity

4.1 Introduction

We have seen in Section 2.4 that Graph Cut algorithms usually employ rectangular grid graphs as their data structure. In segmentation applications on high-resolution 3D medical images where several tens of millions of voxels are common, such graphs have to be defined on a number of nodes equal to the number of image voxels and, when 6-connected voxel neighborhoods are used, a number of edges at least thrice the number of nodes for neighborhood links plus twice the number of nodes for terminal links. With data structure size touching on the limit or exceeding the storage capacity of computer memory along with supralinear time complexity of best performing maximum flow algorithms [BK04], numerical optimization becomes impractical.

Several approaches have been proposed to address the complexity of Graph Cut and other graph-based image segmentation approaches on high-resolution input. They can be broadly classified into two categories; algorithm parallelization and image simplification. Parallel versions of some maximum flow algorithms of the Push-relabel variant discussed in Section 2.3.2 have been devised for shared-memory multiprocessor architectures [DB08] and graphics processing units (GPU) [VN08]. While these methods attain good speedups, they offer no solutions for the memory footprint problem since they operate on the entire image domain. Image simplification approaches address this problem directly observing that the full-resolution pixel representation is often redundant because objects usually comprise many similar pixels that could be grouped. In [LSGX05], Graph Cut segmentation is first applied to a low-resolution version of the image and the solution is propagated to the full resolution where Graph Cut segmentation is applied in a narrow band surrounding the projected foreground-background boundary. In order to accelerate feedback in their interactive Graph Cut segmentation scheme, [LSTS04] exploit the fact that Graph Cut algorithms lend themselves equally well to irregular nongrid graphs and oversegment the image into small regions via the Watershed algorithm [VS91]. Graph Cut segmentation is then formulated according to the watershed regions graph. Starting from a grid partition, [CA10] cluster image pixels

by an iterative k-means algorithm augmented by color similarity and shape compactness constraints. The image is then represented by a cluster graph which is partitioned via the Normalized Cuts approach [SM00] based on color information only yielding a coarse oversegmentation of the image. Recent conditional random field (CRF) image segmentation approaches such as [GRC+08, FVS09, LLB+11] rely on image clustering not only to reduce memory footprint and accelerate runtime, but also to collect image features from clusters and their neighborhood.

In order to reduce memory footprint and accelerate Graph Cut optimization, we cluster image voxels prior to segmentation by an intensity and gradient-adaptive centroidal Voronoi tessellation (CVT). Image partitions produced by this clustering method strike a good balance between the conflicting criteria of cluster compactness and structure boundary adherence furthermore allowing to control the tradeoff between them and helping towards the fine-grained placement of segmentation boundaries. We formulate Graph Cut segmentation according to the graph of CVT clusters so that the resulting labeling energy, defined after (2.10), is independent of the clustering resolution [KVDP11]. Compared to a direct application of segmentation on voxels, the clustering step improves overall runtime and memory footprint of the segmentation process up to an order of magnitude virtually without compromising the segmentation quality.

4.2 Related work

Several image tessellation algorithms have been developed by vision and pattern analysis communities for image classification and segmentation applications. Of special interest is a class of image partition algorithms which produce a dense oversegmentation of compact segments of relatively uniform size and shape which adapt to local intensity edges in the image. These are often called superpixels, a term introduced by [RM03] who proposed a Normalized Cuts-based algorithm that recursively bipartitions an image minimizing a cut cost function. Superpixels algorithms with better runtime performance and comparable oversegmentation quality have been introduced since, such as the TurboPixels algorithm of [LSK+09] and the Graph Cut superpixels algorithm of [VBM10]. The oversegmentation produced by these algorithms achieves a better balance between the conflicting goals of cluster compactness and structure boundary adherence than some well-known image partition algorithms which produce segments of highly variable shape and size, like Watershed [VS91] and Mean-shift [CM02] algorithms and the oversegmentation algorithm of [FH04]. Unsurprisingly, a large segment of irregular shape is more likely to span more than one object, especially in the absence of boundary cues in the image due to insufficient contrast or the presence of shadows. A compact, regular shape, on the other hand, is less likely to cross object boundaries unless they are too wiggly. Moreover, if such a segment does encompass more than one object, the overall error rate can be controlled if its size is not allowed to grow arbitrarily. Cluster compactness, however, comes at the expense of boundary adherence, therefore it is desirable to be able to control the tradeoff between the two criteria by an appropriate parameterization of the algorithm. Later in this chapter, we shall compare

the CVT clustering algorithm we employ with the aforementioned TurboPixels and Graph Cut superpixels algorithms, therefore we review them here before presenting our clustering method in the following section.

The TurboPixels algorithm [LSK+09] is a geometric flow algorithm cast in a banded level-set curve evolution that iteratively dilates an initial set of uniformly distributed contour seeds, each corresponding to one superpixel. Superpixel compactness is maximized by a term in the curve evolution equation which produces constant motion in the outward normal direction in image regions of uniform intensity maximizing the rate of area growth while maintaining the minimum possible isoperimetric ratio. The smoothness of superpixel boundaries and their alignment with local image edges are ensured, respectively, by curvature and gradient-based curve attraction terms in curve evolution equation velocity coefficients. To prevent expanding seeds from overlapping, background skeletonization is used to stop boundary evolution when two distinct dilating seeds are about to collide. We note that upon termination, the algorithm produces more superpixels than the initial number of seeds, since any remaining unassigned large connected regions are treated as superpixels.

Similar to our image segmentation theoretical framework presented in Section 2.2, [VBM10] formulate superpixel partition as a pixel labeling problem and optimize with the multilabel Expansion Moves Graph Cut algorithm [BVZ01]. The image is initially covered with overlapping square patches of the maximum allowed superpixel size, each corresponding to a label, and the optimal "stitching" of these patches is sought, in other words, the optimal assignment of each pixel to only one such patch. In order to align patch boundaries with intensity edges, the stitching cost at image intensity edges is made cheaper by using a weighted Potts pairwise model, which, for the assignment of pairs of neighboring pixels to different patches, incurs a penalty inversely proportional to their intensity difference. The explicit energy formulation allows for different unary constraints to be specified in order to encourage different properties of superpixels. A formulation which was experimentally found to perform best specifies an absolute intensity difference-based unary term that encourages intensity homogeneity inside superpixels at the expense of slightly less regular superpixel sizes and shapes. We note that, instead of the number of superpixels, the maximum superpixel size is considered to be a more natural parameter for the algorithm which could be chosen by the user as appropriate for an application. The number of patches is deduced subsequently by spreading square patches on the image at certain intervals.

4.3 Image clustering by centroidal Voronoi tessellation

Given a grayscale image $\mathcal{I}$ and n sites $c_i \in \mathcal{I}$ such that $1 \leq i \leq n$, a Voronoi tessellation partitions the image $\mathcal{I}$ into n disjoint clusters C_i associated with each site c_i, where every cluster C_i consists of the set of all voxels in the image which, according to a distance measure d, are closer to the associated site than to all other sites:

$$C_i = \{v \in \mathcal{I} \mid d(v, c_i) < d(v, c_j);\ 1 \leq j \leq n, j \neq i\}. \tag{4.1}$$

Unless otherwise mentioned, d is usually the Euclidean distance.

A Centroidal Voronoi Tessellation (CVT) [DFG99] has the additional property that each site c_i corresponds to the mass centroid of the associated cluster and is defined by:

$$c_i = \frac{\sum_{v \in C_i} \rho(v)\, v}{\sum_{v \in C_i} \rho(v)} \tag{4.2}$$

where $\rho(v)$ is a density function defined by (4.5). Moreover, a CVT minimizes the following energy function:

$$F(v; c_i) = \sum_{i=1}^{n} \left(\sum_{v \in C_i} \rho(v) \|v - c_i\|^2 \right). \tag{4.3}$$

We extend the cluster geometry compactness property expressed by (4.3) by adding an intensity-space norm as well:

$$E(v; c_i) = \sum_{i=1}^{n} \left(\sum_{v \in C_i} \rho(v) \left(\|v - c_i\|^2 + \alpha \|I_v - I_i\|^2 \right) \right) \tag{4.4}$$

where I_v is the gray-level of the voxel v, I_i is the gray-level of the cluster C_i defined as the mean intensity of its voxels, and α is a positive scalar allowing to control the intensity adaptivity of the CVT partition. Intuitively, minimizing (4.4) now corresponds to maximizing cluster compactness in terms of both geometry and gray-level.

In (4.2) and (4.4), we define $\rho(v)$ as a linear function of intensity-gradient magnitude at voxel v:

$$\rho(v) = m|\nabla I_v| + b. \tag{4.5}$$

As we shall see later in Section 4.3.2, the density weighting scheme encourages the alignment of CVT clusters with local intensity edges in the image and the formation of relatively small clusters nearby, thus allowing fine-grained placement of subsequent Graph Cut segmentation boundaries.

In order to understand the properties of CVT image clustering, in Figure 4.1 we give 2D illustrations of its output for different parameterizations on a 2D binary image of a solid circle shown in Figure 4.1a. Obviously, the intensity gradient magnitude $|\nabla I_v|$ is zero everywhere in this image except on the circumference of the circle. We show 4 different CVT clusterings of the circle image, along with the dual graph of the CVT for each case overlaid on the cluster map, which corresponds to the Delaunay triangulation of the tessellation's centroids. Each case in Figure 4.1 corresponds to a different setting of parameters α in (4.4), and m and b in (4.5). In Figure 4.1b, we have $\alpha = 0$, $m = 0$ and $b = 1$, therefore the clustering corresponds to a uniform geometric partition according to (4.1) with a constant density where cluster centroids coincide with the barycenters of hexagonal clusters. Figure 4.1c shows an intensity-adaptive clustering of the image according to (4.4) with $\alpha = 5$, $m = 0$ and $b = 1$. Here clusters have similar sizes and furthermore adapt to the domain of the circle since with $\alpha > 0$ both geometry and intensity space similarities are considered. Figure 4.1d shows

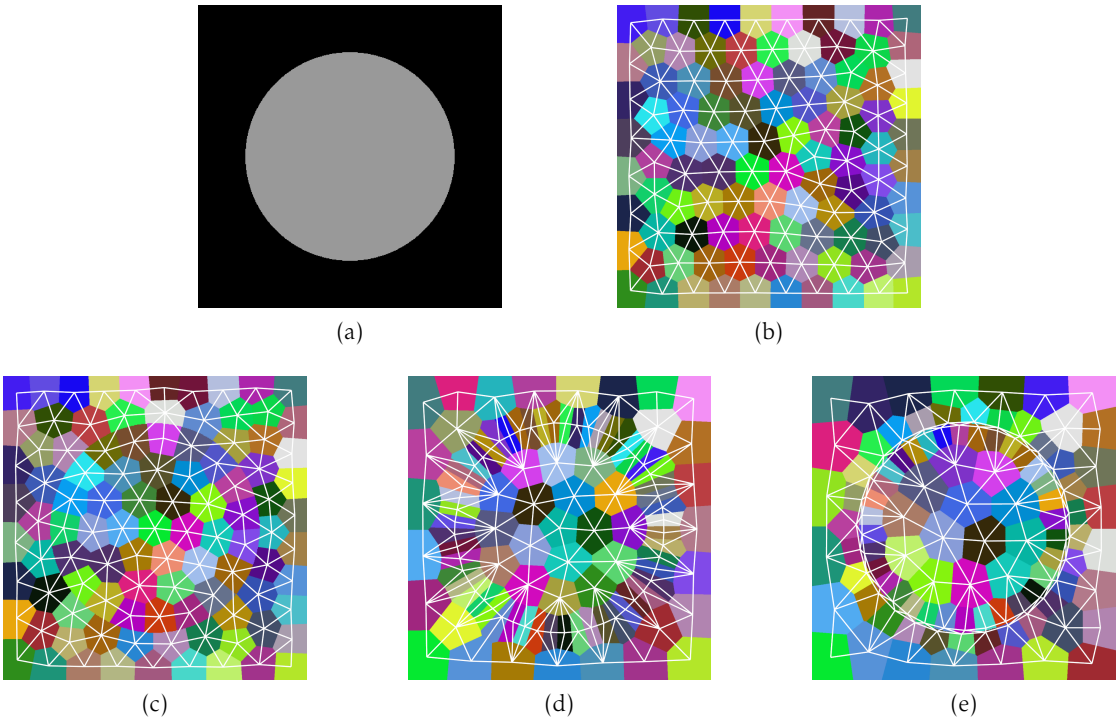


Figure 4.1: A binary image of a solid circle (a) and 4 different CVT clusterings corresponding to different parameter settings. (b) Uniform geometric clustering with no regard to intensity or gradient, (c) intensity-adaptive clustering with constant density, (d) gradient-adaptive clustering, and (e) intensity and gradient-adaptive clustering. For each clustering, the dual graph of the CVT is overlaid on the cluster map.

a gradient-adaptive clustering where $\alpha = 0$, $m = 5$ and $b = 1$. With variable density, cluster sizes are noticeably different, with smaller clusters located near intensity edges, i.e. the circumference of the circle, and larger ones in constant-intensity areas of the image. Notice how clusters run over the circumference without regard for intensity since $\alpha = 0$. Figure 4.1e shows an intensity and gradient-adaptive clustering of the image which produces the desired tessellation; clusters align with the circumference with smaller clusters forming on either side of it while image areas of constant-intensity are coarsely partitioned. Note that when variable density is involved, centroids no longer coincide with barycenters of polygonal clusters.

A number of algorithms exist to construct a CVT minimizing (4.4), including Lloyd's relaxation method for k-means clustering [Llo82]. We use a variant of the clustering algorithm first proposed by [VC04] which approximates a CVT in a computationally-efficient manner involving only local queries on voxels located on boundaries of pairs of clusters. The original algorithm was applied in the context of the uniform coarsening of triangle meshes, and was later extended to discrete image volumes by [DVS+09]. Results of quantitative analysis and qualitative comparison, reported in Section 4.3.2, with the TurboPixels and the Graph Cut superpixels algorithms on the Berkeley segmentation dataset [MFTM01] confirm that the CVT clustering algorithm is substan-

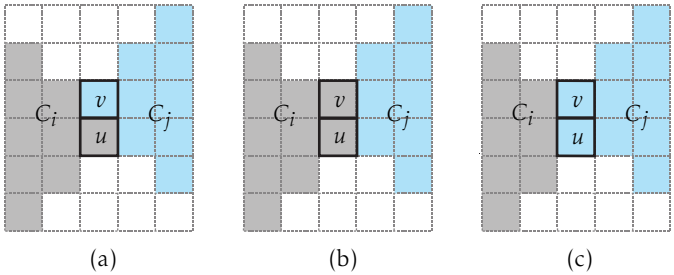


Figure 4.2: The three voxel assignment possibilities to a pair of clusters, in gray and blue, during energy minimization. (a) Initial configuration, (b) C_i grows; $C_i \leftarrow u, v$, and (c) C_j grows; $C_j \leftarrow u, v$.

tially faster than the other two algorithms while generating tessellations of comparable quality to theirs. Runtime evaluations on medical images, given in Section 5.6, show that the total runtime of CVT clustering and Graph Cut segmentation is 2–10 times better than voxel segmentation alone with no compromises in segmentation quality for most practical purposes. The following section outlines the clustering algorithm.

4.3.1 The clustering algorithm

Here we outline the CVT clustering algorithm of [VC04] which we use to partition an image before proceeding with Graph Cut segmentation. Algorithm 4.3.1 gives its listing.

The algorithm starts by randomly selecting n voxels and assigning each to one of the n projected clusters C_i (lines 1–2). It assigns all of the remaining voxels to a special null cluster representing the fact that these voxels do not yet belong to any cluster (lines 3–4). The algorithm then proceeds to minimize the energy (4.4) iteratively, examining all pairs of adjacent clusters and updating voxel membership in clusters according to boundary tests for local minimum-energy configurations (lines 7–26). If one of the clusters is the null cluster (lines 10–15), the algorithm assigns all of the voxels in its boundary set, denoted by ∂C, which are neighbors with voxels in the nonnull cluster to the latter without performing any energy calculations, thus growing the nonnull cluster. If both clusters are nonnull, the algorithm checks whichever of the three possible voxel assignments, illustrated in Figure 4.2, gives the largest decrease in energy and updates the current clustering configuration accordingly. Note that boundary tests rely on voxel neighborhood systems as defined in Section 2.2 for voxels in the boundary sets of clusters. The algorithm's main loop terminates either when no clusters are modified during its last iteration or when the maximum number of iterations are executed.

We note, without giving details, that [VC04] formulates the energy (4.4) so that it depends only on the assignment of voxels to clusters, therefore cluster energy computations on line 21 do not involve any explicit calculation of cluster centroids. Moreover, in order to ensure implementation efficiency, cluster energy terms are stored in computer memory and updated incrementally as the configuration changes so that the energy computation for a cluster is a constant-time operation. Hence, the asymptotic time complexity of the clustering algorithm is that of its inner loop: $O(\Delta n e)$, where $\Delta = \max_{\forall i,j \in \mathcal{S}} |\partial C_i \cap \partial C_j|$ is the greatest cluster common-boundary length, and e is the

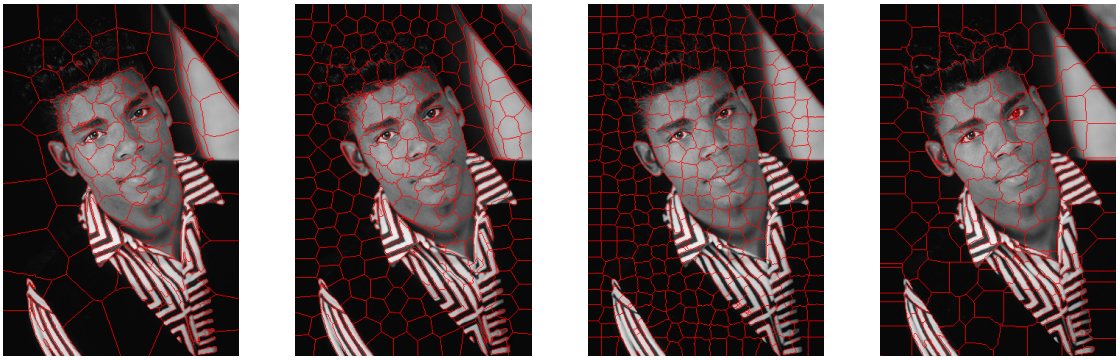
Algorithm 4.3.1: Image clustering via CVT approximation

input : image $\mathcal{I}$, number of clusters n, maximum iterations $maxIterations$
output: image partition $\mathcal{C} = \{C_1, \ldots, C_n\}$ indexed by $\mathcal{S}$

// Calculate initial clustering
for $i \leftarrow 1$ **to** n
  $C_i \leftarrow \textsc{Random}(v \in \mathcal{I})$
foreach $v \in \mathcal{I} \mid \nexists C_i, v \in C_i$
  $C_{null} \leftarrow v$

// Perform energy minimization
$\mathcal{S}' = \mathcal{S} \cup \{null\}$
$iterations \leftarrow 0$
repeat
  $modifications \leftarrow 0$
  foreach $\{i, j\} \in \mathcal{S}' \times \mathcal{S}' \mid |\partial C_i \cap \partial C_j| \neq 0$
    if $i = null$
      foreach $u \in \partial C_i \mid \exists v \in \partial C_j, v \in \mathcal{N}_u$
        $C_j \leftarrow u$
    else if $j = null$
      foreach $u \in \partial C_j \mid \exists v \in \partial C_i, v \in \mathcal{N}_u$
        $C_i \leftarrow u$
    else
      foreach $\{u, v\} \in \partial C_i \times \partial C_j \mid v \in \mathcal{N}_u$
        $conf_{init} \longleftarrow \{C_i \leftarrow u, C_j \leftarrow v\}$
        $conf_i \longleftarrow \{C_i \leftarrow u, v\}$
        $conf_j \longleftarrow \{C_j \leftarrow u, v\}$
        if $\textsc{Energy}(conf_i) < \textsc{Energy}(conf_{init})$ **or** $\textsc{Energy}(conf_j) < \textsc{Energy}(conf_{init})$
          $conf_{new} \longleftarrow \textsc{MinEnergy}\left(conf_i, conf_j\right)$
          $\textsc{UpdateCVT}(conf_{new})$
          $modifications \leftarrow modifications + 1$
  $iterations \leftarrow iterations + 1$
until $modifications = 0$ **or** $iterations = maxIterations$

ratio of the number of edges to the number of vertices in the dual graph of the CVT giving a mean estimate of the number of neighbors to consider per cluster during boundary tests. The convergence of the Algorithm 4.3.1 is guaranteed since the energy is positive and each modification decreases its value.



(a) Recall = 0.590
Seg. err. = 1.528

(b) Recall = 0.641
Seg. err. = 1.256

(c) Recall = 0.518
Seg. err. = 2.014

(d) Recall = 0.591
Seg. err. = 2.759

Figure 4.3: Qualitative and quantitative comparison of 300-cluster tessellations of a Berkeley dataset grayscale image via the CVT clustering algorithm with (a) $m = 1$ and (b) $m = 0$, (c) the TurboPixels and, (d) the Graph Cut superpixels algorithms.

4.3.2 Evaluating the clustering algorithm

In this section, we present the results of quantitative evaluation and qualitative validation of the CVT clustering algorithm comparing its performance with those of the TurboPixels algorithm [LSK+09] and the Graph Cut supepixels algorithm [VBM10].

We shall first define the two performance metrics used in quantitative analysis; undersegmentation error and boundary recall. Given a CVT-clustering of an image $\mathcal{C} = \{C_1, \ldots, C_n\}$ indexed by $\mathcal{S}$ and a ground-truth partition of it $\mathcal{T} = \{g_1, \ldots, g_k\}$, the undersegmentation error for segment g_j is defined as the total amount of surface bleeding induced by clusters which overlap the segment g_j, normalized by the segment's area:

$$\underset{j}{\text{seg err}}\,(\mathcal{C}, \mathcal{T}) = \frac{\left(\sum_{i:C_i \cap g_j \neq \emptyset} |C_i|\right) - |g_j|}{|g_j|} \tag{4.6}$$

It is obvious that the above fraction equals zero when all clusters overlapping the segment g_j fall inside it, that is, no cluster crosses the segment's boundary. When measuring the undersegmentation error of an algorithm for a given clustering, we average the above fraction over all segments of the corresponding ground-truth partition and give the mean error.

Boundary recall is defined straightforwardly as the fraction of ground-truth boundary pixels which fall within a given small distance from at least one cluster boundary. Since we seek high precision in segmentation boundary placement for medical applications, we use a distance of one pixel.

In order to stay in line with quantitative results reported in [LSK+09] and [VBM10], we also use the Berkeley segmentation database [MFTM01] for quantitative evaluation.

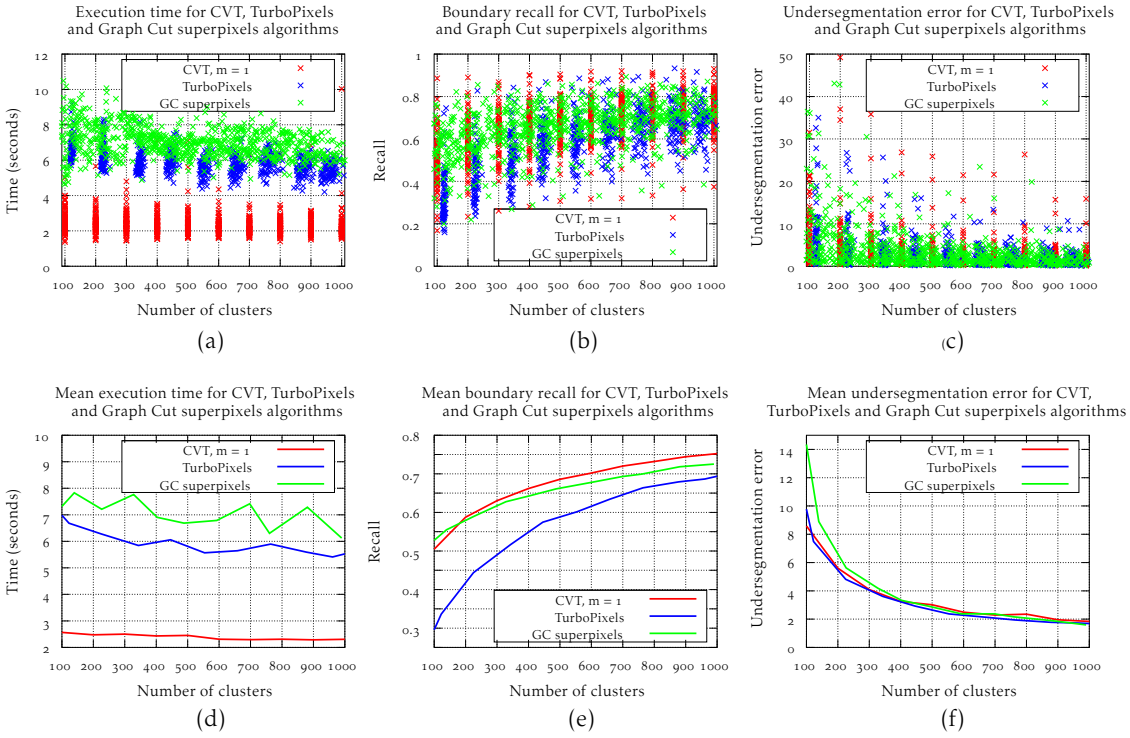


Figure 4.4: Quantitative evaluation of CVT clustering, TurboPixels and Graph Cut superpixels algorithms on 80 grayscale images from the Berkeley segmentation dataset.

This dataset comprises 300 generic photos of size 321×481 or 481×321 pixels which have been presented to human subjects both in color and grayscale modes for manual segmentation. For evaluation purposes, we select 80 of the 100 grayscale images from the test subset, and choose one ground-truth segmentation for each such image so that it does not contain small segments of few pixels. Such small ground-truth segments can create large outliers and disrupt mean undersegmentation error measurements.

Quantitative evaluations are performed with respect to increasing number of clusters in the range $[100, 1000]$. In all evaluations, the temperature hyperparameter of the Graph Cut superpixels algorithm is set to 50, a default recommended by the author's implementation, and two iterations of the algorithm are applied. The maximum number of iterations for the CVT clustering algorithm is set to 1000. In practice, however, the algorithm terminates much earlier. The intensity-norm weighting coefficient α in (4.3) is set to one, and the algorithm's performance is evaluated with respect to two definitions of the density function (4.5); $m = 1$, yielding a gradient and intensity adaptive clustering, and $m = 0$, yielding an intensity adaptive clustering.

Results of quantitative evaluation for TurboPixels, Graph Cut superpixels and the CVT clustering algorithms are presented in Figure 4.4. We first give scatter plots of boundary recall and undersegmentation error measures along with execution time for all 80 images in the dataset in Figures 4.4a–4.4c, then we give mean performance mea-

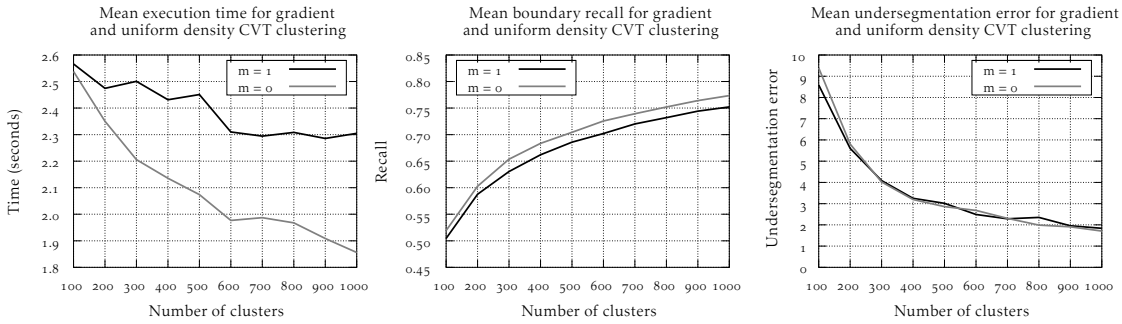
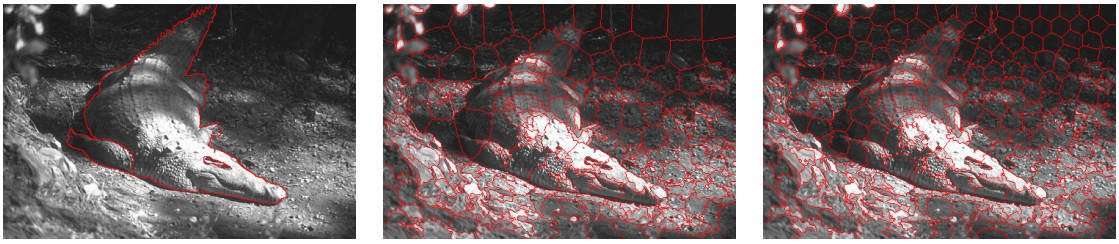


Figure 4.5: Quantitative evaluation of the CVT clustering algorithm on 80 grayscale images from the Berkeley segmentation dataset with respect to gradient-based and constant density functions.

sures, averaged over all images for a given clustering resolution in Figures 4.4d–4.4f. From these graphs, we can readily see that the CVT clustering algorithm outperforms the other two in terms of execution time by a wide margin and that its boundary recall is mostly higher than those of the other algorithms. Its mean undersegmentation error is comparable to that of the TurboPixels algorithm and is better than that of the Graph Cut superpixels algorithm with lower clustering resolutions.

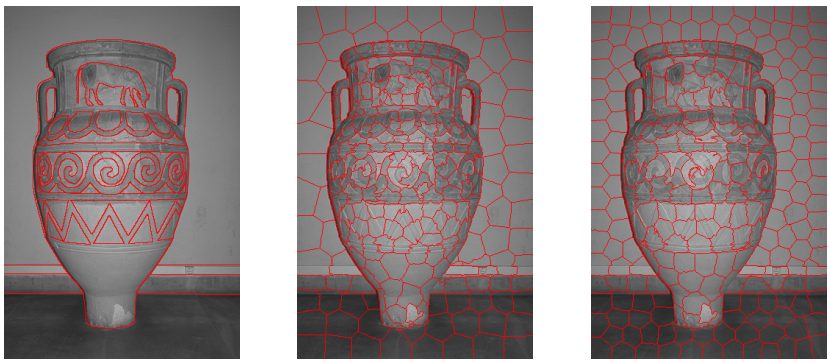
We can see in Figures 4.4a–4.4c that both TurboPixels and Graph Cut superpixels algorithms produce a variable number of output clusters. As previously mentioned, the TurboPixels algorithm assigns remaining connected regions to new superpixels upon termination and the Graph Cut superpixels algorithm deduces the number of superpixels from the specified maximum superpixel size, although, in theory, the number of superpixels could be specified directly as the number of labels to optimize. Mean performance figures for these algorithms in Figures 4.4d–4.4f are therefore reported with respect to the average number of output clusters for each input resolution. Note that the CVT clustering algorithm produces the exact number of clusters specified, which is important in applications where the required cluster budget has to be respected.

We present the results of quantitative evaluation of the CVT clustering algorithm with respect to different density function definitions separately in Figure 4.5. Here we can see that the average performance of the gradient-based CVT clustering is slightly lower, more significantly in terms of mean boundary recall, it also takes longer to execute. These results are not surprising, as the dataset on which evaluations are performed features highly textured images with nonuniform illumination and substantial presence of shadows. The gradient-based density function, while decreasing the size of clusters, increases their density in textured areas of the image. This reduces the cluster budget for other image areas, which, in the presence of shadows or with insufficient contrast, provide poor gradient information for precise cluster placement. This point is illustrated in Figure 4.6b. In Figure 4.6c we see that with a constant density function, the density of clusters across the image and their relative size are more uniform. Furthermore, clusters adapt to small local patches of homogeneous intensity better, es-



(a) (b) Recall = 0.458 Seg. err. = 3.273 (c) Recall = 0.576 Seg. err. = 3.159

Figure 4.6: Qualitative and quantitative evaluation of the CVT clustering algorithm on a textured image from the Berkeley segmentation dataset with respect to (b) gradient-based and (c) constant density functions. Ground-truth segmentation boundary is given in (a).



(a) (b) Recall = 0.571 Seg. err. = 3.055 (c) Recall = 0.541 Seg. err. = 3.680

Figure 4.7: Qualitative and quantitative evaluation of the CVT clustering algorithm on a nontextured image from the Berkeley segmentation dataset with respect to (b) gradient-based and (c) constant density functions. Ground-truth segmentation boundary is given in (a).

pecially near the tail and the right side of the alligator's trunk, thus retrieving a larger fraction of the object's boundaries compared to the gradient-based density CVT in 4.6b. Measures provided below each image confirm our observations quantitatively.

The gradient-based density function is however useful in retrieving shape detail with nontextured images of fairly uniform illumination. Importantly, most medical CT and MR images have such properties. This is illustrated in Figure 4.7b, where, despite the small cluster budget, 300, the CVT clustering with gradient-based density function has retrieved more boundaries of the pot's ornaments than the constant-density CVT given in Figure 4.7c. Once again, measures provided underneath each image confirm our observations quantitatively.

A side-by-side qualitative comparison of partitions by the three clustering algorithms for a Berkeley database photo of a human subject is given in Figure 4.3.

4.4 Segmentation energy redefined

In Section 2.2, we formulated segmentation as a labeling problem, defined as the assignment of a label from a set of m labels $L = \{l_1, \ldots, l_m\}$ representing objects to be segmented to each of the variables in a set of n variables corresponding to image voxels. Henceforward, we shall assume that these variables correspond to the clusters of a CVT-clustered image $\mathcal{C} = \{C_1, \ldots, C_n\}$ indexed by $\mathcal{S}$. We shall assume furthermore that each variable $i \in \mathcal{S}$ is associated with the corresponding node in the dual graph of the CVT, $\tilde{\mathcal{G}}$, illustrated in Figure 4.1, which we define in the following. Denote the boundary (surface, in 3D) of a cluster C_i by ∂C_i, and the length (area) of the common boundary (surface) of a pair of clusters $i, j \in \mathcal{S}$ by $|\partial C_i \cap \partial C_j|$. Therefore, for a pair of adjacent clusters $|\partial C_i \cap \partial C_j| \neq 0$. Define the dual graph of the CVT, $\tilde{\mathcal{G}} = \langle \mathcal{S}, \tilde{\mathcal{E}} \rangle$, as an undirected graph on cluster centroids where pairs of adjacent clusters define the set of edges $\tilde{\mathcal{E}} = \left\{ \{i, j\} \mid i, j \in \mathcal{S}, |\partial C_i \cap \partial C_j| \neq 0 \right\}$. Consequently, the neighborhood of a node $i \in \mathcal{S}$ is defined as $\tilde{\mathcal{N}}_i = \left\{ j \mid j \in \mathcal{S}, \exists \{i, j\} \in \tilde{\mathcal{E}} \right\}$ where $i \notin \tilde{\mathcal{N}}_i$.

Now the prior probability distribution of label configurations can be expressed as an MRF of label configurations with respect to the graph $\tilde{\mathcal{G}}$, and if image-derived likelihood densities can be defined for each label $l \in L$, then the MAP configuration yielding the best segmentation with respect to user-supplied evidence and constraints can be found by minimizing the energy function (2.10) with the obvious replacement of the voxel neighborhood system $\mathcal{N}_i$ by the cluster neighborhood system $\tilde{\mathcal{N}}_i$. We reproduce the energy function of segmentation here for convenience.

$$E(\ell) = t \sum_{i \in \mathcal{S}} D_i(\ell_i) + \frac{1}{2} \sum_{i \in \mathcal{S}} \sum_{j \in \tilde{\mathcal{N}}_i} V_{i,j}(\ell_i, \ell_j). \tag{4.7}$$

Recall that the unary term $D_i(\ell_i)$ in (4.7), also called the data term, is a negative log-likelihood function derived from observed data and measures the cost of assigning a label to the variable $i \in \mathcal{S}$, and that pairwise terms $V_{i,j}(\ell_i, \ell_j)$ encode prior information on pairwise interactions between labels assigned to pairs of neighboring sites.

In the following sections we redefine unary and pairwise energy terms of (4.7) to suit the Bayesian labeling of CVT-clustered images. Importantly, this redefinition maintains the clustering-resolution independence of segmentation energy, which we establish formally in Section 4.4.3.

4.4.1 Likelihood-based data cost redefined

Recall that the likelihood probability for a variable $i \in \mathcal{S}$ given the label assignment ℓ_i is given by the conditional probability $\Pr(O_i = I_i \mid X_i = \ell_i)$, where the observation O constitutes the spatially distinct records of image intensity levels at each variable. The conditional probability distribution of intensity $\Pr(I \mid L = l)$ for every label $l \in L$ is estimated as Gauss-smoothed and normalized intensity histograms from user-supplied "seeds", as described in Section 2.2.1. The definition of the term $D_i(\ell_i)$ is slightly different however. To facilitate the understanding of different cases, we define it as follows:

$$D_i(\ell_i) = \begin{cases} 0, & \exists h^k \in C_i, k = \ell_i \quad (4.8a) \\ \infty, & \exists h^k \in C_i, k \neq \ell_i \quad (4.8b) \\ -\ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i), & \nexists h^k \in C_i, \forall k \in L \quad (4.8c) \end{cases}$$

where h^k represents a seed of label type $k \in L$ attributed to a voxel $v \in C_i$ and I_v is the gray-level of voxel v. (4.8a) (respectively (4.8b)) corresponds to the negative log-likelihood value at maximum (zero) probability and indicates no cost (infinite cost) for assigning the label $k \in L$ (any label but k) to a variable i when there is at least one voxel in the associated cluster C_i marked by a seed of label type k. Recall that conditions (4.8a) and (4.8b) amount, in practice, to imposing hard constraints on configurations in the space $\mathcal{L}$, and thus encourage plausible segmentations. (4.8c) accounts for all other variables for which there is no seed to impose a labeling decision. This is the case for the vast majority of variables, since the user is expected to introduce only a small number of seeds with respect to the number of image voxels.

Note that in (4.8c) the likelihood probability for a variable $i \in \mathcal{S}$, which corresponds to a cluster C_i, is defined on intensity observations at image voxels $v \in C_i$ as $\Pr(O_i = I_i \mid X_i = \ell_i) = \prod_{v \in C_i} \Pr(I_v|\ell_i)$ under the assumption of probabilistic independence of such observations. An alternative specification of likelihood probability $\Pr(O_i = I_i \mid X_i = \ell_i) = \Pr(I_i|\ell_i)$ defined on intensity observations at clusters, the intensities of which could be calculated as the mean intensity of its voxels, would not take intensity variance within clusters into account and furthermore the resulting energy would not maintain resolution independence.

4.4.2 Prior redefined

We weight the pairwise prior term $V_{i,j}(\ell_i, \ell_j)$ in (4.7) by the length (area, in 3D) of the common boundary (surface) of adjacent clusters $|\partial C_i \cap \partial C_j|$ so that the sum of pairwise terms in the energy of some configuration $E(\ell)$ for any pair of labels would be equal to the length (area) of the common boundary (surface) between the corresponding pair of objects multiplied by a constant, equal to the length of the shortest path between corresponding nodes in the prior graph of adjacency relations as defined in Section 3.4. The definition of the pairwise prior thus becomes:

$$V_{i,j}\left(\ell_i, \ell_j\right) = |\partial C_i \cap \partial C_j| \, \omega(a, b), \quad \ell_i = a, \ell_j = b. \tag{4.9}$$

Note that clusters sharing longer boundaries (larger surfaces, in 3D) would prefer similar labels, or label assignments incurring smaller penalties.

In addition to its theoretical soundness, as shown in the following section, results of quantitative evaluation presented in Section 5.3 establish the practical advantage of this weighting scheme compared to the empirical inverse-distance prior weighting method frequently encountered in MRF-based segmentation literature, for example in [BF06] and [CCB+05].

4.4.3 Energy invariance

We have mentioned previously that the definitions of data and prior terms of (4.7) given in Sections 4.4.1 and 4.4.2 respectively maintain the clustering-resolution independence of segmentation energy. This means that the energy of identical label configurations on CVT clusterings of different resolution is invariant. In particular, given CVT clusterings of different resolution, the energies of segmentations corresponding to identical MAP solutions ℓ^* are identical.

The following theorem establishes the energy invariance property with respect to data energy formulations disregarding hard constraints as defined by (4.8a) and (4.8b). In the presence of hard constraints, energy invariance no longer holds. The corollary of the energy invariance theorem shows that when hard constraints are considered, the energy of a MAP solution ℓ^* on a higher resolution CVT clustering is greater than the energy of an identical MAP solution obtained on a clustering of lower resolution.

Theorem 4.4.1 (Energy Invariance). *Let $\mathcal{C} = \{C_1, \ldots, C_m\}$ indexed by S and $\mathcal{C}' = \{C_1, \ldots, C_n\}$ indexed by S' be two CVT clusterings of the image $\mathcal{I}$ with $m < n$, and let $T = \{M_1, \ldots, M_{|L|}\}$ and $T' = \{M'_1, \ldots, M'_{|L|}\}$ be two segmentations on $\mathcal{C}$ and $\mathcal{C}'$ respectively resulting from minimizing (4.7) in some labeling space $\mathcal{L}$ (= L^m and L^n respectively) with respect to label likelihood-based data and pairwise prior terms defined by (4.8c) and (4.9) respectively. If $T = T'$, then $E(\ell^*) = E'(\ell^*)$, where $\ell^* = \arg\min_{\ell \in \mathcal{L}} E(\ell)$.*

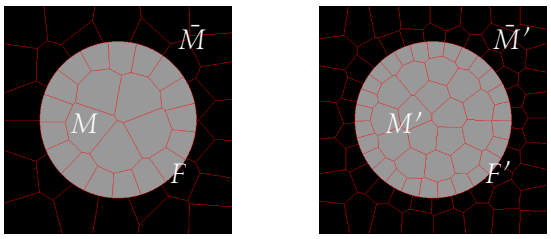


Figure 4.8: Identical binary segmentations on two CVT clusterings of different resolution.

Proof. We give the proof for $|L| = 2$, since the generalization to $|L| > 2$ is straightforward. As illustrated in Figure 4.8, let $T = \{M, \bar{M}\}$ and $T' = \{M', \bar{M}'\}$. Let S_M index clusters $C_i \in M$ and $S_{\bar{M}}$ index clusters $C_i \in \bar{M}$. Define $S'_{M'}$ and $S'_{\bar{M}'}$ for T' similarly.

Let us write the sums of unary energies of ℓ^* for T and T' as follows:

$$E_1(\ell^*) = t \sum_{i \in S} D_i(\ell_i) = t \sum_{i \in S_M} D_i(\ell_i) + t \sum_{i \in S_{\bar{M}}} D_i(\ell_i) \tag{4.10a}$$

$$E'_1(\ell^*) = t \sum_{i \in S'} D_i(\ell_i) = t \sum_{i \in S'_{M'}} D_i(\ell_i) + t \sum_{i \in S'_{\bar{M}'}} D_i(\ell_i). \tag{4.10b}$$

We shall show that the respective sums on the right hand side of (4.10a) and (4.10b) are equivalent.

If $T = T'$, then

$$M = M'$$

therefore

$$|M| = |M'|.$$

Expanding both sides on voxels

$$\sum_{i \in S_M} \sum_{v \in C_i} 1 = \sum_{i \in S'_{M'}} \sum_{v \in C_i} 1$$

and summing over the negative log-likelihood for every $v \in C_i$, we get

$$\sum_{i \in S_M} \sum_{v \in C_i} -\ln \Pr(I_v \mid \ell_i) = \sum_{i \in S'_{M'}} \sum_{v \in C_i} -\ln \Pr(I_v \mid \ell_i).$$

Rewriting the sum of logarithms on both sides as the logarithm of the product we get

$$\sum_{i \in S_M} -\ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i) = \sum_{i \in S'_{M'}} -\ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i)$$

which, by definition (4.8c), gives

$$\sum_{i \in S_M} D_i(\ell_i) = \sum_{i \in S'_{M'}} D_i(\ell_i).$$

Similarly, we can show that

$$\sum_{i \in S_{\bar{M}}} D_i(\ell_i) = \sum_{i \in S'_{\bar{M}'}} D_i(\ell_i).$$

Hence

$$E_1(\ell^*) = E'_1(\ell^*). \tag{4.11}$$

Now we show that the sums of pairwise energies of ℓ^* for T and T' are equivalent. As illustrated in Figure 4.8, denote the boundary between the pair of segments in T and T' by F and F' respectively, and let us define the corresponding sets of cluster pairs falling on either side of each boundary as $B = \{(i, j) \mid C_i \in M, C_j \in \bar{M}\}$ and $B' = \{(i, j) \mid C_i \in M', C_j \in \bar{M}'\}$.

If $T = T'$, then

$$F = F'$$

therefore

$$|F| = |F'|$$

which can also be expressed as the length of the common boundary between the pair of segments in either segmentation

$$|\partial M \cap \partial \bar{M}| = |\partial M' \cap \partial \bar{M}'|.$$

Expanding both sides on common boundaries between pairs of clusters on either side of each segmentation boundary, we get

$$\sum_{i,j\in B} |\partial C_i \cap \partial C_j| = \sum_{i,j\in B'} |\partial C_i \cap \partial C_j|.$$

Let $l \in L$ denote the label of segments M and M' and $\bar{l} \in L$ that of segments $\bar{M}$ and $\bar{M}'$. Multiplying both sides by a constant $k = V_{i,j}(\ell_i = l, \ell_j = \bar{l})/2$

$$\frac{1}{2}\sum_{i,j\in B} |\partial C_i \cap \partial C_j| V_{i,j}(\ell_i, \ell_j) = \frac{1}{2}\sum_{i,j\in B'} |\partial C_i \cap \partial C_j| V_{i,j}(\ell_i, \ell_j)$$

and expanding both sums on cluster neighborhood, we get

$$\frac{1}{2}\sum_{i\in S_M}\sum_{\substack{j\in\tilde{\mathcal{N}}_i \\ j\in S_{\bar{M}}}} |\partial C_i \cap \partial C_j| V_{i,j}(\ell_i, \ell_j) = \frac{1}{2}\sum_{i\in S'_{M'}}\sum_{\substack{j\in\tilde{\mathcal{N}}_i \\ j\in S'_{\bar{M}'}}} |\partial C_i \cap \partial C_j| V_{i,j}(\ell_i, \ell_j).$$

Hence

$$E_2(\ell^*) = E'_2(\ell^*). \tag{4.12}$$

From (4.11) and (4.12) we get $E(\ell^*) = E'(\ell^*)$, thus completing the proof. □

Corollary 4.4.1. *Given two CVT clusterings $\mathcal{C}$ and $\mathcal{C}'$ of the image $\mathcal{I}$ indexed by S and S', and two identical segmentations on them T and T' resulting from minimizing* (4.7) *in some labeling space with respect to hard constraints and label likelihood-based data expressed by* (4.8) *and pairwise prior defined by* (4.9)*. If $|\mathcal{C}| < |\mathcal{C}'|$, then $E(\ell^*) < E'(\ell^*)$.*

Proof. Denote the energy of the optimal labeling ℓ^* with respect to label likelihood-based data (4.8c) and pairwise prior(4.9) by $E(\ell^*)$. Let S_H index the set of variables subject to hard constraints $S_H = \{i \in S \mid \exists h_k \in C_i, k \in L\}$. The energy of ℓ^* with respect to label likelihood *and* hard constraints-based data (4.8) and pairwise prior(4.9) is

$$E_c(\ell^*) = E(\ell^*) - \sum_{i\in S_H} -t \ln \prod_{v\in C_i} \Pr(I_v \mid \ell_i) \tag{4.13}$$

because $\forall i \in S_H, D_i(\ell_i) = 0$. Obviously, $E_c(\ell^*) < E(\ell^*)$.

Let $|\mathcal{C}| = m$ and $|\mathcal{C}'| = n$. If $m < n$, then the average cluster size of $\mathcal{C}$ is greater than that of $\mathcal{C}'$

$$\frac{1}{m}\sum_{i\in S} |C_i| > \frac{1}{n}\sum_{i\in S'} |C_i|.$$

Therefore, for S_H and S'_H

$$\sum_{i\in S_H} |C_i| > \sum_{i\in S'_H} |C_i|.$$

Expanding both sides on voxels

$$\sum_{i \in S_H} \sum_{v \in C_i} 1 > \sum_{i \in S'_H} \sum_{v \in C_i} 1$$

and summing over the negative log-likelihood for every $v \in C_i$, we get

$$\sum_{i \in S_H} \sum_{v \in C_i} -\ln \Pr(I_v \mid \ell_i) > \sum_{i \in S'_H} \sum_{v \in C_i} -\ln \Pr(I_v \mid \ell_i).$$

Rewriting the sum of logarithms on both sides as the logarithm of the product we get

$$\sum_{i \in S_H} -\ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i) > \sum_{i \in S'_H} -\ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i).$$

From the Energy Invariance theorem we have $E(\ell^*) = E'(\ell^*)$. Multiplying both sides of the above inequality by $-t$ and adding $E(\ell^*)$ to both sides we get

$$E(\ell^*) - \sum_{i \in S_H} -t \ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i) < E(\ell^*) - \sum_{i \in S'_H} -t \ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i)$$

hence $E_c(\ell^*) < E'_c(\ell^*)$, thus completing the proof. □

4.5 Conclusions

In this chapter, we have addressed the complexity of Graph Cut image segmentation on high-resolution input, in particular on 3D medical images often comprising several millions of voxels. We presented an image simplification approach to partition such an image prior to segmentation by an efficient intensity and gradient-adaptive centroidal Voronoi tessellation (CVT) method. It allows to control the tradeoff between the conflicting criteria of cluster compactness and structure boundary adherence, and furthermore produces dense partitions of relatively small clusters near intensity edges helping towards the fine-grained placement of subsequent segmentation boundaries.

We have redefined the Bayesian labeling problem and the Graph Cut minimization of its energy according to the graph of CVT clusters. As we have established formally, our redefinition maintains the clustering-resolution independence of the segmentation energy, which means that identical image segmentations on CVT partitions of different resolution have identical energies.

Comparative evaluation results of our CVT image clustering method with closely-related TurboPixels and Graph Cut superpixels algorithms on the Berkeley segmentation dataset confirmed the advantages of our method in terms of both runtime and quality of produced partitions. In the next chapter, quantitative and qualitative evaluation of our CVT-clustered segmentation approach on synthetic, simulated and real medical images confirm that, compared to a direct voxel segmentation approach, the clustering step improves the total runtime and memory footprint of the segmentation process up to an order of magnitude without compromising the segmentation quality for most practical purposes.

Chapter 5

Evaluations and applications

5.1 Introduction

This chapter summarizes the results of quantitative analysis and qualitative validation of the vicinity prior-based Graph Cut segmentation approach proposed in Section 3.4 on synthetic, simulated and real medical CT and MR images. In all evaluations, comparisons are drawn with the standard Potts prior-based segmentation and in some cases also with the adjacency prior-based approach introduced in Section 3.3. Let us first outline the conditions under which evaluations presented in this chapter are performed.

For comparison purposes with clustered image segmentation as formulated in Section 4.4, we perform some of the evaluations on image pixels (voxels). We apply the basic framework of Bayesian labeling outlined earlier in Section 2.2 by letting variables $i \in \mathcal{S}$ correspond to image pixels (voxels) $v \in \mathcal{I}$ and defining an 8-connected (6-connected) neighbourhood $\mathcal{N}_i$ for each variable (respectively in 3D).

In performance measurements with respect to varying number of clusters, we use 19 clustering resolutions, reported in different figures and tables in this chapter as percentages of the number of image voxels. In all evaluations, we set the coefficients m and b of the clustering density function (4.5) to $m = b = 1$, although we have seen in Section 4.3.2 that in some cases, especially with images having high noise levels, careful setting of these parameters can improve the clustering aspect and yield higher boundary recall and lower undersegmentation error.

We minimize the energy (4.7) using the aforementioned three prior models with the Expansion Moves algorithm discussed in Section 2.4.2. In all evaluations, we set the algorithm's maximum number of cycles to 40. In practice, however, the algorithm converges to a local minimum rapidly in 3–5 cycles. We note that all our software implementations use the multilabel energy optimization library developed by the Computer Vision Research Group at the University of Western Ontario [BVZ01, KZ04, BK04].

Before presenting the results in detail, we shall define performance metrics used in quantitative analysis in the following section.

5.2 Performance measures

We measure the quality of segmentation with respect to ground-truth via the widely-used Dice similarity metric (DSM) [Dic45]. Denote the set of voxels labeled with $l \in L$ in the segmented image $\mathcal{I}$ and the ground-truth $\mathcal{T}$ by S_l and T_l respectively, the DSM for label class $l \in L$ is defined as:

$$\underset{l}{\mathrm{dsm}}(\mathcal{I},\mathcal{T}) = \frac{2\,|S_l \cap T_l|}{|S_l| + |T_l|}. \tag{5.1}$$

It is obvious that the DSM measures the overlap between the segmentation and the ground-truth. It takes its values in $[0,1]$ where 1 indicates a perfect match and 0 no match at all. In addition to the DSM metric for individual label classes, we employ an "overall" segmentation quality metric calculated from mean weighted DSM measures for all labels in L:

$$\underset{L}{\mathrm{dsm}}(\mathcal{I},\mathcal{T}) = \frac{\sum_{l\in L} \mathrm{dsm}_l(\mathcal{I},\mathcal{T})\,|T_l|}{|\mathcal{T}|}. \tag{5.2}$$

The two other classification performance metrics we use in quantitative analysis are sensitivity and specificity, defined respectively as the true-positive and the true-negative rates for individual label classes [Faw06]:

$$\underset{l}{\mathrm{sens}}(\mathcal{I},\mathcal{T}) = \frac{|S_l \cap T_l|}{|T_l|}, \tag{5.3}$$

$$\underset{l}{\mathrm{spec}}(\mathcal{I},\mathcal{T}) = \frac{\left(\sum_{k\in L} |S_k \cap T_k|\right) - |S_l \cap T_l|}{|\mathcal{I}| - |T_l|}. \tag{5.4}$$

Similarly to (5.2), overall sensitivity and specificity metrics are defined from mean weighted sensitivity and specificity measures for all labels in L.

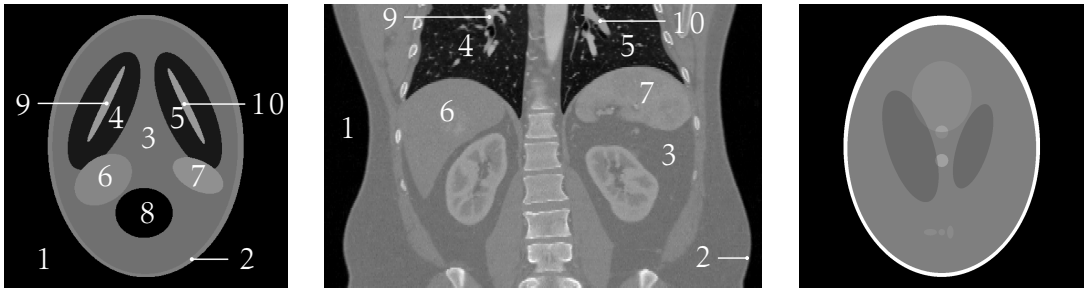
In order to justify the significance of observed improvements brought by the vicinity prior on the aforementioned performance measures, we complement comparison results on medical case studies with statistical testing according to Welch's t test [Wel47], which is an adaptation of Student's t-test intended for use with samples having possibly unequal variances. For two such samples, the test statistic

$$t = \frac{\mu_1 - \mu_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}} \tag{5.5}$$

approximately follows Student's t-distribution with degrees of freedom

$$df = \frac{\left(\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}\right)^2}{\frac{\sigma_1^4}{n_1^2(n_1-1)} + \frac{\sigma_2^4}{n_2^2(n_2-1)}} \tag{5.6}$$

where μ_i, σ_i^2 and n_i are respectively the i^{th} sample mean, variance and size. Given t and df, we calculate a p-value using the t-distribution in a standard fashion and perform



(a) (b) (c)

Figure 5.1: (a) Our synthetic phantom with structures numbered. (b) A coronal cross-section of a CT image showing corresponding structures. (c) The Shepp-Logan phantom.

a one-tailed test of the null hypothesis $H_0 : \mu_{Potts} \geq \mu_{vicinity}$, where μ_{Potts} and $\mu_{vicinity}$ are the sample means of some performance measure calculated on two sets of images segmented with Potts and vicinity priors respectively. Rather than reporting p-values, we highlight vicinity prior performance measures indicating significant improvements according to two significance levels $\alpha = 0.05$ and 0.01.

In order to understand how reliable the segmentation algorithm is with a given prior at attaining a given overall DSM value d, we evaluate a reliability function of DSM over the interval $[0, 1]$ on a set of segmented images $\mathbb{I}$. Define the subset of images attaining an overall DSM level superior to d as $\mathbb{I}_d = \{\mathcal{I} \in \mathbb{I} \mid \mathrm{dsm}_L(\mathcal{I}, \mathcal{T}) > d\}$, and let D be a random variable on $[0, 1]$. The reliability at DSM value d is given by:

$$\mathrm{rel}(d) = \Pr(D > d) = \frac{|\mathbb{I}_d|}{|\mathbb{I}|}. \tag{5.7}$$

We evaluate the precision of segmentation boundary placement with an error metric measuring the Euclidean distance of each point on the surface of a segmented structure from the surface of the corresponding ground-truth structure. Let $M^l_{\mathcal{S}}$ and $M^l_{\mathcal{T}}$ be the triangle meshes representing isosurfaces of structure volumes labeled by $l \in L$ in the segmented image $\mathcal{I}$ and the ground-truth $\mathcal{T}$. We obtain such meshes by applying the Marching Cubes algorithm to labeled volumes [LC87]. The error at $x \in M^l_{\mathcal{S}}$ is given by:

$$\mathrm{err}(x) = \inf\{d(x, y) \mid \forall y \in M^l_{\mathcal{T}}\}. \tag{5.8}$$

Later in this chapter, we give the cumulative histogram of errors calculated for all points on the surface of a given structure. The histogram representation gives a clear indication of the percentage of segmented surface lying within a given distance from the true surface, and consequently allows us to understand how precise the segmentation is in retrieving true structure boundaries.

5.3 Synthetic phantom and dataset

In this subsection, we present the results of quantitative and qualitative evaluation of segmentation on a dataset generated from a synthetic phantom we have created for the very purpose inspired by the Shepp-Logan phantom [SL74] which has been used in tomographic reconstruction. We have already encountered this phantom in Chapter 3. For convenience, its structure is reproduced in Figure 5.1a. Due to the small number of structures, their simplistic spatial arrangement and intersection patterns, the Shepp-Logan phantom, illustrated in Figure 5.1c, is not well-suited for evaluating a segmentation approach using prior information on spatial relationships of structures. The layout of the 10 structures our phantom features mimics the spatial configuration of some thoracic and abdominal organs in a coronal cross-section of a CT scan. This can be seen in Figure 5.1b. Roughly speaking, identically-colored phantom structures 4 and 5 correspond to lungs, right and left, 9 and 10 to pulmonary arteries in each lung, 6 and 7 to the liver and the spleen and 8 to the cavity of the digestive tract, which has zero intensity in areas filled by air.[1] Visceral fat, 3, fills the space between these structures and the whole is covered by a thin envelope of skin, 2, and is placed against a background of air, 1, which is distinct from the digestive tract cavity, although both have identical intensities. The intensity levels of the 10 structures are set from mean intensity values of corresponding structures in a CT image for which ground-truth estimates for these structures are available.

Let us outline the conditions under which evaluations discussed in this section are performed. Graph Cut segmentation with Potts, adjacency and vicinity priors is applied to the pixels as well as to clusterings of 512×512 images generated from the synthetic phantom by the introduction of additive zero-mean Gaussian noise of variable standard deviation σ sweeping the range $[0, 0.2]$ with a step of 0.01. For each noise level, we generate 40 images clamping intensity values to $[0, 1]$. Of ranges swept by the temperature parameter t of the energy (2.10), we report results in the representative window $[0, 1]$ sampled at 0.05. The reason behind the choice is that in preliminary experiments, where we probed 0.005-sampled temperature ranges with an upper bound much higher than 1, we did not find any subranges where Potts prior segmentation, as measured by mean overall DSM, outperformed the best result of vicinity prior segmentation over all subranges. Moreover, segmentation quality with both priors usually deteriorated with temperature settings $t > 1$. The set of "seeds" used to estimate likelihood densities as well as to impose hard constraints on segmentation in all evaluations on the synthetic dataset is given in Figure 5.2. The number of seed-marked pixels for each label is in the range $[32, 160]$ and roughly proportional to the size of the corresponding structure, otherwise hard constraints of equal image support for all labels would create uneven bias of segmentation ease for smaller structures, like 9 and 10.

In the remaining of this section as well as in Figures 5.2, 5.3, 5.5, 5.4, 5.6 and 5.7 any references to DSM are actually for overall DSM measures as defined by (5.2).

[1] The digestive tract cavity is not visible in the same plane as the other structures in the coronal cross-section in Figure 5.1b, however it is visible in the sagittal cross-section of the same image in Figure 5.16c

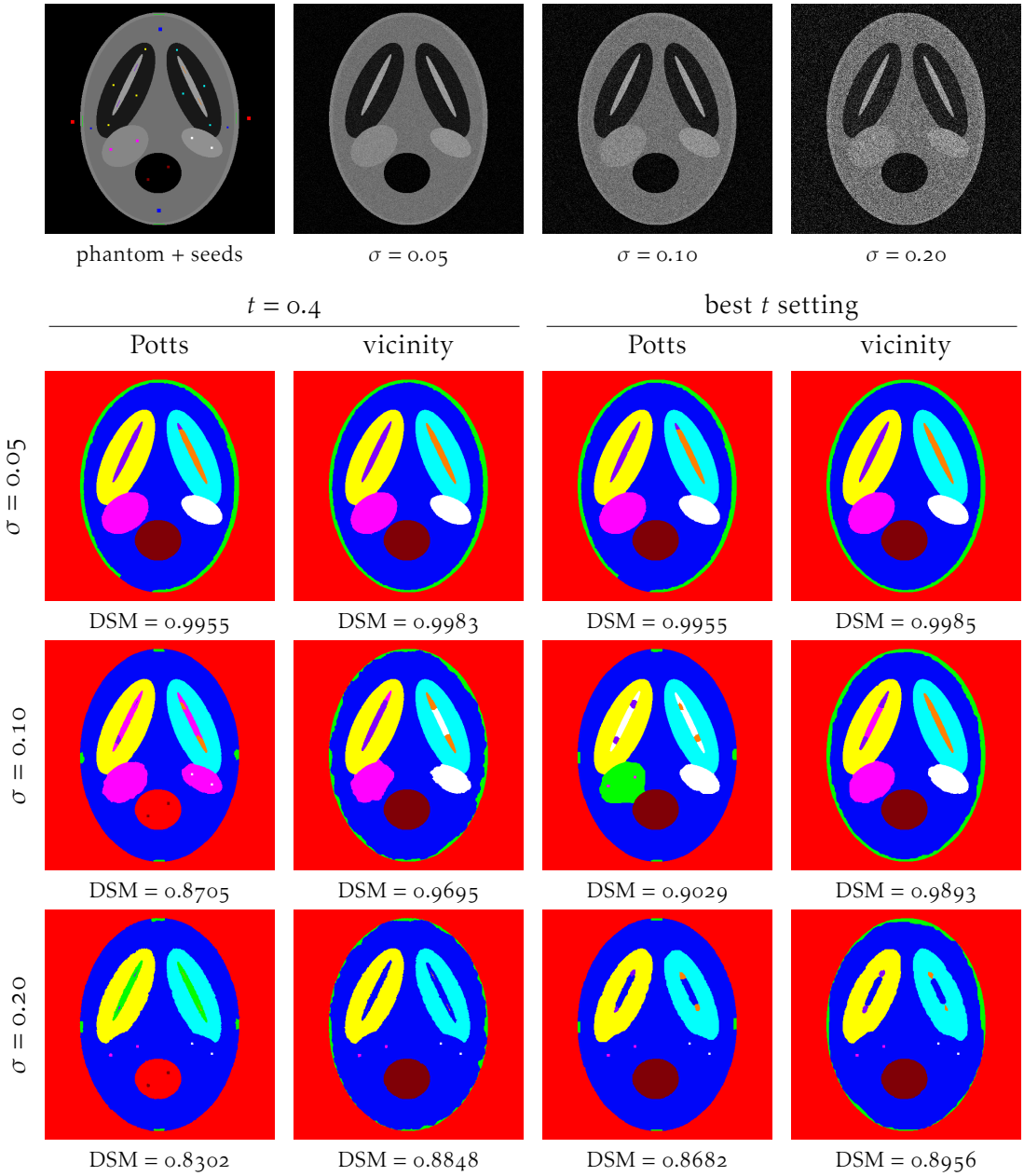


Figure 5.2: Qualitative inspection and comparison of Potts and vicinity prior pixel segmentations on the synthetic phantom dataset. (top row) The phantom along with "seeds" used in all evaluations on the dataset and three images generated from the phantom with additive zero-mean Gaussian noise of $\sigma = 0.05$, 0.10 and 0.20. Columns left and right give best Potts and vicinity prior segmentations at a fixed temperature and over all temperature settings on all images generated at a given noise level. Refer to color code in Figure 3.1b.

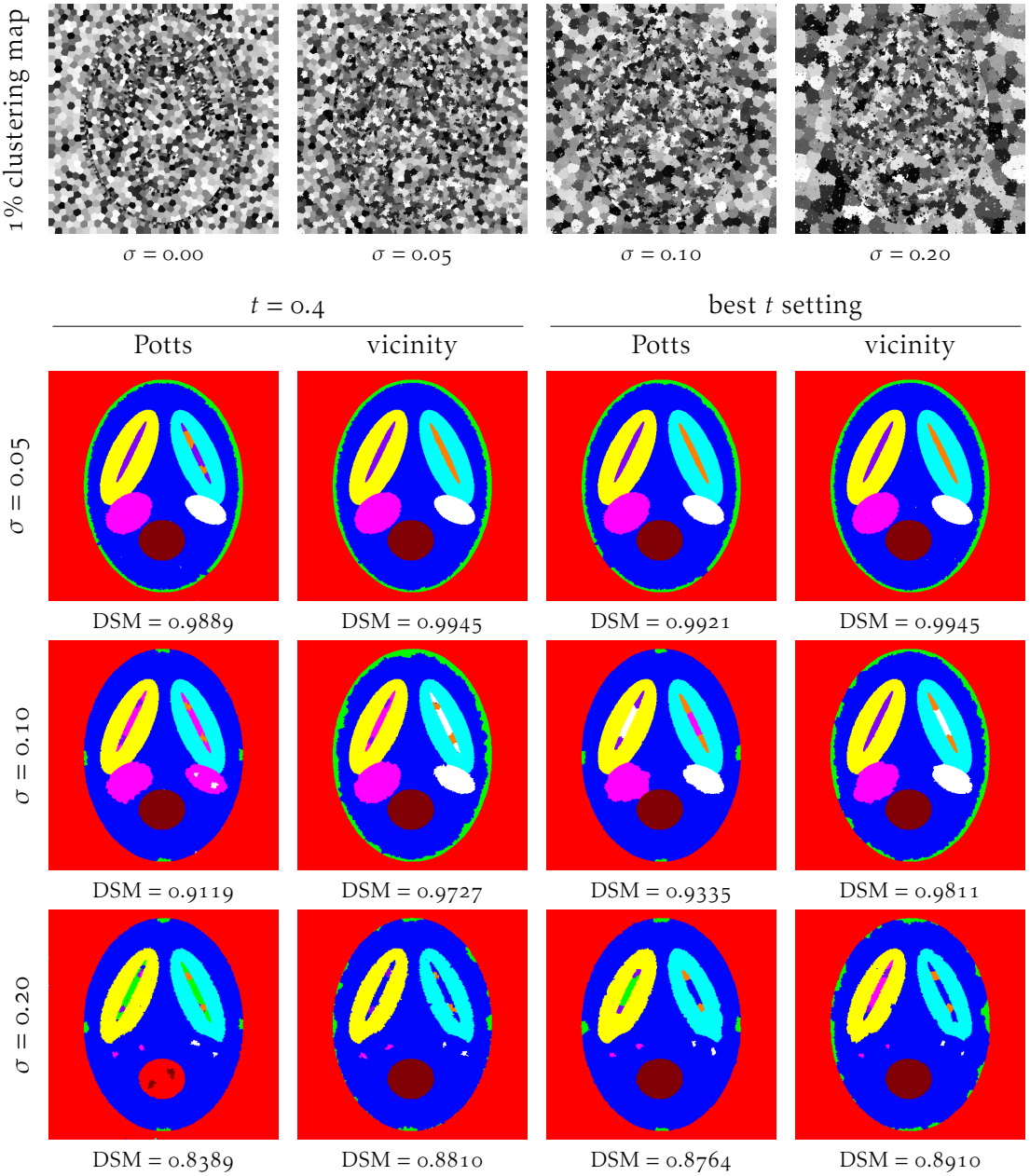


Figure 5.3: Qualitative inspection and comparison of Potts and vicinity prior 10%-clustered image segmentations on the synthetic phantom dataset. For illustration purposes, top row gives cluster maps of 1%-clustered images generated from the phantom with additive zero-mean Gaussian noise of $\sigma = 0.00$, 0.05, 0.10 and 0.20. Columns left and right below give best Potts and vicinity prior 10%-clustered image segmentations at a fixed temperature and over all temperature settings on all images generated at a given noise level. Refer to color code in Figure 3.1b.

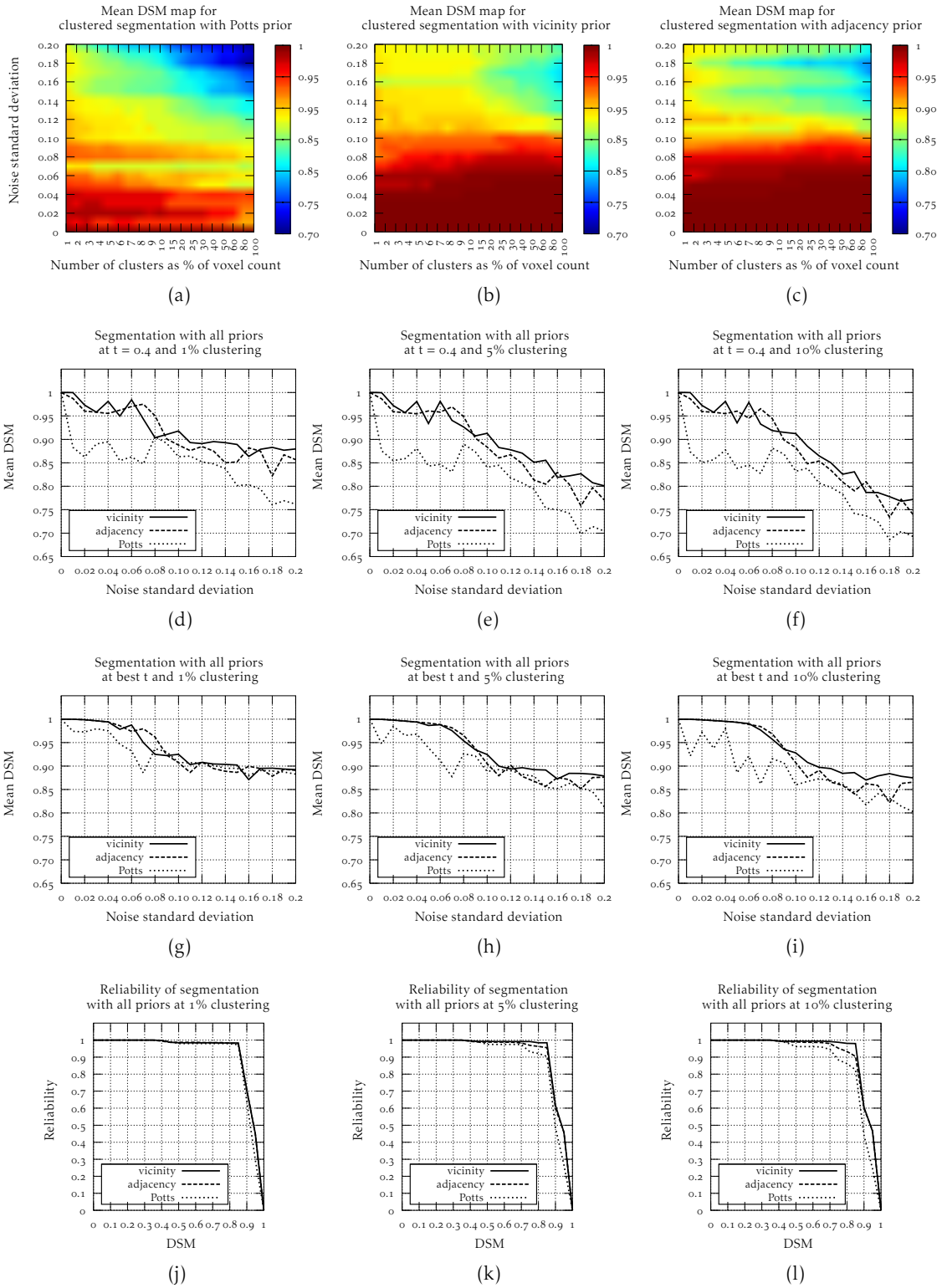


Figure 5.4: Quantitative evaluation and comparison of Potts, adjacency and vicinity prior-based clustered image segmentation on the synthetic phantom dataset.

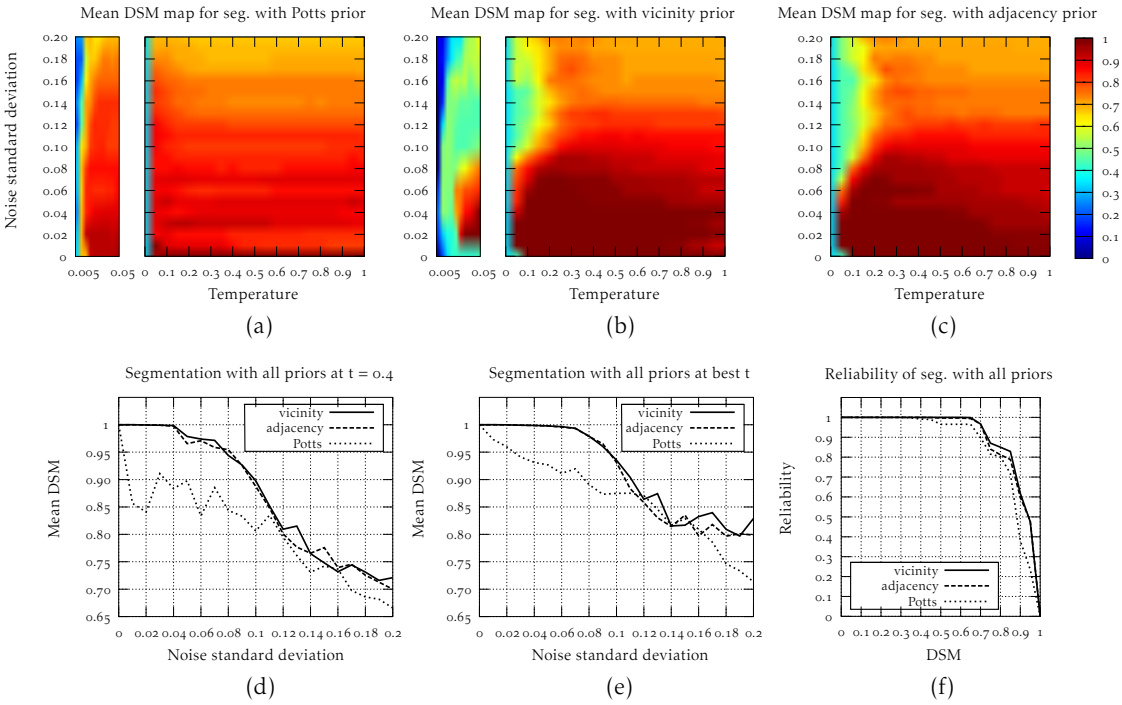


Figure 5.5: Quantitative evaluation and comparison of Potts, adjacency and vicinity prior-based pixel segmentation on the synthetic phantom dataset.

Figures 5.2 and 5.3 give a qualitative comparison of Potts and vicinity prior-based Graph Cut segmentation applied to pixels and 10%-clustered images at 3 noise levels. Comparisons are drawn between best segmentations obtained with either prior on all images generated at a given noise level for a fixed temperature $t = 0.4$ and over all temperature settings. The vicinity prior was defined according to the graph model of structure adjacency given in Figure 3.1b on page 40. Improvements and consistency in segmentation results brought by the vicinity prior, which are furthermore confirmed by DSM measures, are clear on both pixel and cluster segmentations. We can see that the pair of structures 1 and 8 having identical intensity levels and the pair 7 and 6 having close intensity levels, with the latter structure having a rather close level to structure 2, are always segmented correctly as distinct structures when the vicinity prior approach is able to retrieve them. The approach also shows better ability to retrieve thinner structures 9, 10 and the envelope 2 in particular.

In Figures 5.4a–5.4c and Figures 5.5a–5.5c, we give mean DSM maps for segmentation with Potts, adjacency and vicinity priors applied to clusters and pixels respectively. In the latter group, we calculate mean DSM values for segmentations corresponding to a particular temperature setting on all images generated for a given noise level. In the former group, such mean DSM values corresponding to a particular clustering resolution are calculated at temperature settings producing the best segmentation on all images generated for a given noise level. Compared to the Potts prior approach, im-

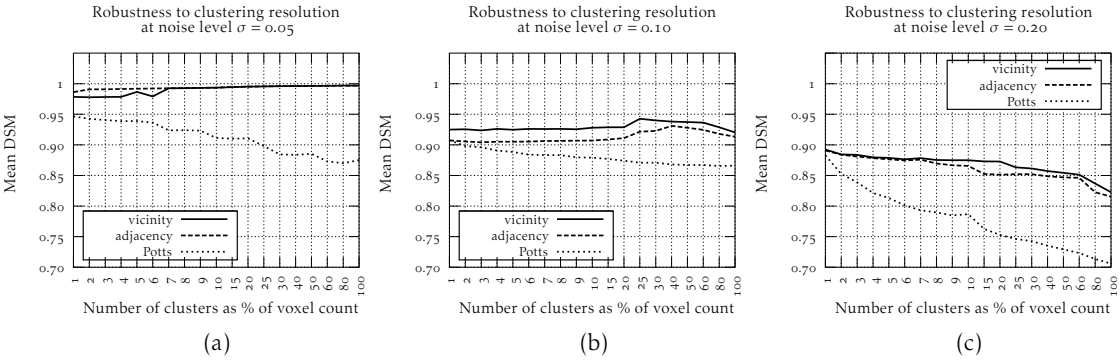


Figure 5.6: Quantitative evaluation and comparison of the robustness of Potts, adjacency and vicinity prior-based image segmentation to clustering resolution on the synthetic phantom dataset.

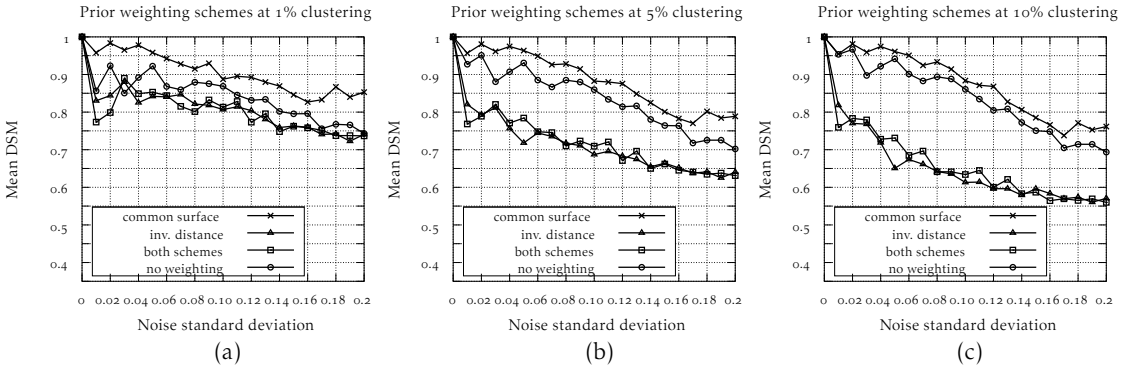


Figure 5.7: Quantitative evaluation and comparison of pairwise prior energy weighting schemes in vicinity prior segmentation for 3 clustering resolutions.

provements brought on mean DSM by the vicinity prior are remarkable, particularly for noise levels up $\sigma = 0.1$. Note that the adjacency prior segmentation displays overall performance patterns similar to its vicinity prior counterpart albeit being less robust to temperature settings and more sensitive to noise levels higher than $\sigma = 0.10$ as it can be observed in Figures 5.4c and 5.5c respectively.

Note that the low mean DSM values corresponding to low temperature settings in Figures 5.5a, 5.5b and 5.5c are due to high regularization by the prior since data terms do not contribute as much at low temperatures. In particular, the minimum energy configuration corresponding to the setting $t = 0$ is a constant labeling, since $V(a, a) = 0$ for any a. Recall that the vicinity prior incurs higher regularization than the Potts prior since its highest penalty in this case study is four times greater than its Potts counterpart. To observe the difference between the two priors on a finer scale, we give a magnified view of DSM maps on the low temperature range $[0.005, 0.05]$.

In order to understand the effect of additive Gaussian noise parameter σ alone, we give mean DSM comparisons of Potts, adjacency and vicinity prior segmentations on pixels and 3 clustering resolutions with respect to increasing noise levels at a fixed

temperature $t = 0.4$ (Figures 5.4d–5.4f, 5.5d) and at the temperature setting yielding the best segmentation for all images generated at a given noise level (Figures 5.4g–5.4i, 5.5e). As for reliability curves, they are given in Figure 5.5f for pixel segmentation, and Figures 5.4j–5.4l for segmentation corresponding to the 3 clustering resolutions. Quantitative measures in both groups of evaluations confirm that the vicinity prior segmentation approach outperforms the Potts prior segmentation, and the adjacency prior approach to a lesser extent.

We also evaluate the robustness of Potts, adjacency and vicinity prior segmentations to clustering resolution. Results of this evaluation for 3 noise levels are presented in Figures 5.6a–5.6c. We can readily see that with vicinity and adjacency priors, segmentation is quite robust to clustering resolution, even at a noise level as high as $\sigma = 0.20$, although it attains lower mean DSM values with the latter prior. This means that similar segmentation results can be obtained using lower clustering resolutions thus improving both runtime and memory footprint. For this evaluation, we calculate mean DSM measures for segmentations corresponding to the best temperature setting on all images generated at given noise level.

Lastly, in Figure 5.7 we present the results of evaluating different weighting schemes of pairwise prior terms in the segmentation energy (4.7) for 3 clustering resolutions. Let C_i and C_j be a pair of adjacent clusters sharing a common boundary (a surface in 3D), and denote their centroids by c_i and c_j respectively. We compare the following four weighting schemes. 1) Cluster common-boundary length weighting (4.9) where the pairwise prior is weighted by $|\partial C_i \cap \partial C_j|$. As shown by the Energy Invariance theorem 4.4.1, this weighting scheme along with the likelihood definition (4.8) maintain the clustering-resolution independence of segmentation energy. 2) Inverse centroid-distance weighting frequently encountered in MRF-based segmentation literature, for example in [BF06] and [CCB+05], where the pairwise term is weighted by $1/d(c_i, c_j)$. 3) Both weighting schemes applied together. 4) No weighting. As it can be seen from graphs in Figure 5.7, the common-boundary length weighting scheme is a clear winner independently of the clustering resolution thus confirming the theoretical result.

5.4 Simulated MRI: BrainWeb

In this section we summarize results of quantitative and qualitative validation of Graph Cut segmentation with Potts and vicinity priors on simulated MRI images. We first give quantitative evaluation results on the original BrainWeb simulated MRI dataset [CZK+98] and compare them to 3 tissue-class voxel segmentation results reported by [GTC11] who use an iterative variant of the Expansion Moves algorithm with the Potts prior model. Then we present a segmentation case study for a head image from the second generation BrainWeb simulated MRI dataset [AEC06].

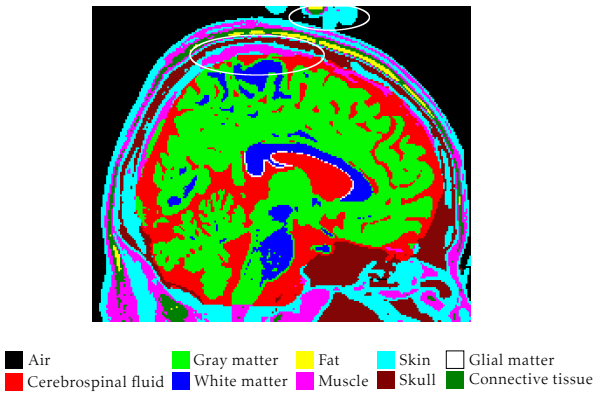


Figure 5.8: A sagittal cross-section of the head and brain phantom of the original BrainWeb simulated MRI dataset showing artifacts and anatomical inconsistencies.

5.4.1 Original BrainWeb simulated MRI dataset

As mentioned, we perform a comprehensive quantitative evaluation on the original BrainWeb simulated MRI dataset [CZK+98]. In the creation of this dataset, a semi-realistic anatomical head and brain phantom was used by an MRI simulator of T1-, T2- and PD-weighted modalities to generate images according to MR acquisition physics with 6 noise levels (0%, 1%, 3%, 5%, 7% and 9%) and 3 intensity inhomogeneity levels (0%, 20% and 40%). For evaluation purposes, we use 18 images from the T1-weighted modality covering the entire noise and inhomogeneity ranges. Image volumes are of size $217 \times 181 \times 217$ voxels with isotropic 1 mm voxel size. The phantom itself used to generate images comprises 10 structure and tissue classes.

It should be noted that the phantom exhibits certain anatomical inconsistencies and structural artifacts. As shown in Figure 5.8, some structures corresponding to tissue classes "muscle" and "skin" are located inside the skull within the volume of tissue class "cerebrospinal fluid" and next to "gray matter" tissue. We therefore define our graph-based anatomical model of structure adjacency, given in Figure 5.10a, as a compromise between phantom imprecisions and true anatomical properties. We note that we use 9 out of the 10 tissue classes the phantom features, as we merge the tissue class "glial matter" into the class "gray matter" without affecting anatomical consistency.

In Tables 5.1a, 5.1b and 5.1c, we give means and standard deviations for overall and individual-class DSM, sensitivity and specificity measures calculated on voxel and cluster segmentations corresponding to best temperature settings for all images in the dataset. We define the vicinity prior according to the graph model presented in Figure 5.10a. Reported clustered image segmentation results correspond to a clustering resolution of 10%. Vicinity prior performance measures indicating statistically significant improvements at levels $\alpha = 0.05$ and 0.01 are highlighted with blue and red respectively. It is easy to observe in these tables that Graph Cut segmentation attains better average performance levels with the vicinity prior on the original BrainWeb database. Moreover, it performs better on clusters, compared to its performance on image voxels.

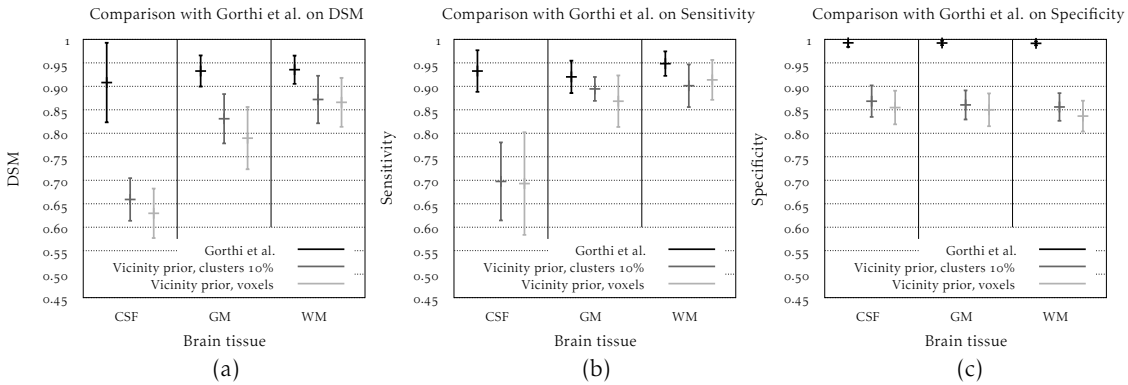


Figure 5.9: Quantitative comparison of brain tissue segmentation with [GTC11] on the original BrainWeb simulated MRI dataset.

We compare our results on the original BrainWeb simulated MRI dataset with those reported by Gorthi et al. in [GTC11] which is the only recent work we are aware of that evaluates the Expansion Moves algorithm for brain tissue classification using the Potts prior model. There are, however, some differences between their approach and ours. Firstly, they use Gaussian likelihood distributions the parameters of which are learned during an iterative application of the algorithm. Secondly, and more importantly, using the phantom as a volume mask they retain only brain structures i.e. cerebrospinal fluid, gray and white matters and discard other head tissues [Gor12]. Thus they formulate segmentation as a multilabeling problem with 3 labels plus 1 for the background. This not only reduces computational complexity, but also makes the segmentation task easier since other head structures, some of which have intensity levels similar to those of brain structures such as muscle and gray matter tissues, are eliminated beforehand. The aforementioned 3 brain structures, on the other hand, exhibit good contrast in images in general, and have reasonably distinct mean intensity values which further facilitate the segmentation task. In our evaluation, we use the same subset of the original BrainWeb dataset as [GTC11] corresponding to the 12 T1-weighted images generated with inhomogeneity levels 20% and 40% and all noise levels. In Figure 5.9, we compare means and standard deviations for DSM, sensitivity and specificity measures for the 3 brain structures from full-image vicinity prior segmentations with corresponding measures in [GTC11]. Unsurprisingly, results we report are not superior. Specialized approaches are expected to, and indeed should, produce better results. Therefore, the present comparison should be viewed as an indication of how close a generic approach can come to a specialized one on this problem instance. Nevertheless, a practical advantage of our approach is that no manual tissue removal is required to segment the brain volume.

	DSM			
	voxels		clusters 10%	
Structures	Potts	vicinity	Potts	vicinity
Air	0.98 ± 0.01	0.98 ± 0.01	0.98 ± 0.01	0.99 ± 0.00
Cerebrospinal fluid	0.25 ± 0.16	0.65 ± 0.06	0.29 ± 0.11	0.67 ± 0.05
Gray matter	0.79 ± 0.09	0.82 ± 0.08	0.80 ± 0.08	0.85 ± 0.06
White matter	0.86 ± 0.08	0.88 ± 0.07	0.87 ± 0.07	0.89 ± 0.06
Fat	0.89 ± 0.05	0.84 ± 0.08	0.87 ± 0.04	0.85 ± 0.06
Muscle	0.73 ± 0.06	0.74 ± 0.12	0.73 ± 0.07	0.79 ± 0.09
Skin	0.70 ± 0.07	0.75 ± 0.09	0.69 ± 0.07	0.75 ± 0.07
Skull	0.76 ± 0.07	0.79 ± 0.05	0.76 ± 0.05	0.80 ± 0.04
Connective tissue	0.71 ± 0.14	0.69 ± 0.21	0.73 ± 0.12	0.75 ± 0.15
Overall	0.81 ± 0.03	0.85 ± 0.04	0.82 ± 0.03	0.87 ± 0.04

(a)

	Sensitivity			
	voxels		clusters 10%	
Structures	Potts	vicinity	Potts	vicinity
Air	0.97 ± 0.02	0.97 ± 0.01	0.98 ± 0.01	0.98 ± 0.00
Cerebrospinal fluid	0.28 ± 0.29	0.72 ± 0.11	0.24 ± 0.21	0.72 ± 0.09
Gray matter	0.79 ± 0.13	0.90 ± 0.06	0.81 ± 0.12	0.91 ± 0.03
White matter	0.84 ± 0.10	0.93 ± 0.04	0.87 ± 0.08	0.91 ± 0.05
Fat	0.87 ± 0.05	0.78 ± 0.13	0.86 ± 0.05	0.81 ± 0.09
Muscle	0.85 ± 0.04	0.76 ± 0.18	0.84 ± 0.05	0.81 ± 0.13
Skin	0.85 ± 0.08	0.79 ± 0.16	0.82 ± 0.07	0.84 ± 0.03
Skull	0.90 ± 0.05	0.91 ± 0.05	0.87 ± 0.07	0.88 ± 0.07
Connective tissue	0.74 ± 0.13	0.63 ± 0.26	0.74 ± 0.12	0.71 ± 0.20
Overall	0.82 ± 0.03	0.85 ± 0.04	0.83 ± 0.03	0.87 ± 0.03

(b)

	Specificity			
	voxels		clusters 10%	
Structures	Potts	Vicinity	Potts	Vicinity
Air	0.73 ± 0.05	0.79 ± 0.08	0.74 ± 0.05	0.80 ± 0.07
Cerebrospinal fluid	0.86 ± 0.04	0.87 ± 0.04	0.87 ± 0.04	0.88 ± 0.04
Gray matter	0.84 ± 0.03	0.86 ± 0.04	0.84 ± 0.02	0.87 ± 0.03
White matter	0.82 ± 0.03	0.85 ± 0.04	0.83 ± 0.03	0.87 ± 0.03
Fat	0.82 ± 0.03	0.86 ± 0.04	0.83 ± 0.03	0.87 ± 0.03
Muscle	0.82 ± 0.03	0.87 ± 0.03	0.83 ± 0.03	0.88 ± 0.03
Skin	0.83 ± 0.04	0.87 ± 0.03	0.84 ± 0.03	0.89 ± 0.03
Skull	0.82 ± 0.03	0.85 ± 0.04	0.83 ± 0.03	0.87 ± 0.03
Connective tissue	0.83 ± 0.03	0.87 ± 0.03	0.84 ± 0.03	0.88 ± 0.03
Overall	0.79 ± 0.04	0.83 ± 0.05	0.80 ± 0.04	0.84 ± 0.05

(c)

Table 5.1: Means and standard deviations of overall and individual-class DSM, sensitivity and specificity measures for voxel and 10%-clustered image segmentations of the original BrainWeb dataset with Potts and vicinity priors. Significant improvements at $\alpha = 0.05$ and 0.01 are given in blue and red respectively.

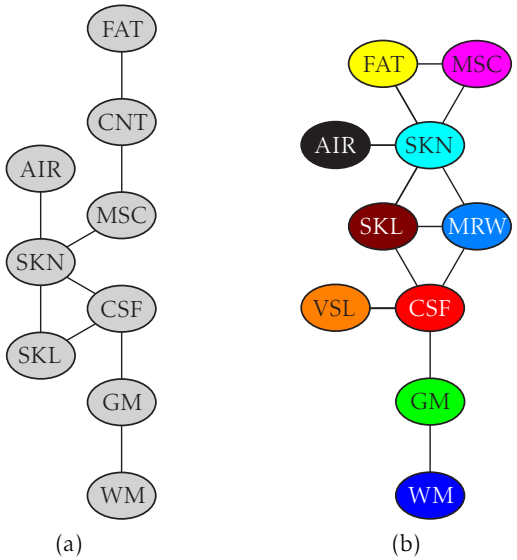


Figure 5.10: Graph-based anatomical models of structure adjacency relations for (a) the original and (b) the second generation BrainWeb simulated MR dataset phantoms. Structures corresponding to graph vertices; AIR: background, CNT: connective tissue, CSF: cerebrospinal fluid, FAT: fat, GM: gray matter, MRW: bone marrow, MSC: muscle, SKL: skull, SKN: skin, VSL: vessels and WM: white matter.

5.4.2 Second-generation BrainWeb simulated MR image

In this section, we present a segmentation case study of a single image from the second generation BrainWeb simulated MRI dataset [AEC06]. This dataset features 20 simulated T1-weighted head MR images generated by an MRI simulator at a fixed noise level of 3% according to subject-specific phantoms. These phantoms are superior to that of the original BrainWeb dataset in terms of anatomical precision and consistency. They comprise 12 tissue classes of which we use 10 merging tissue classes "dura mater" and "around fat" respectively into classes "cerebrospinal fluid" and "muscle" without affecting anatomical consistency. The image we choose for evaluation purposes is that of subject no. 4. Its volume is of size $217 \times 181 \times 217$ voxels with isotropic 1 mm voxel size. We apply Potts and vicinity prior-based Graph Cut segmentation to a 10% clustering and to the voxels of the image at a temperature setting of $t = 0.5$. The motivation behind keeping the temperature at a "factory default" value of 0.5 in the present as well as in other single-image case studies in the following sections comes from observing that in Figures 5.5a and 5.5b Graph Cut segmentation with both priors, and especially with vicinity prior, is reasonably robust to temperature settings in the range $[0.1, 0.7]$ for low to moderate noise levels $[0, 0.07]$. Seeds, illustrated in Figure 5.12b, are introduced by a user in the sagittal view of a graphical interface by marking about 300 pixels for every structure via brush strokes of size 5×5 pixels at most distributed evenly in the volume. The vicinity prior is defined according to the graph model given in Figure 5.10b.

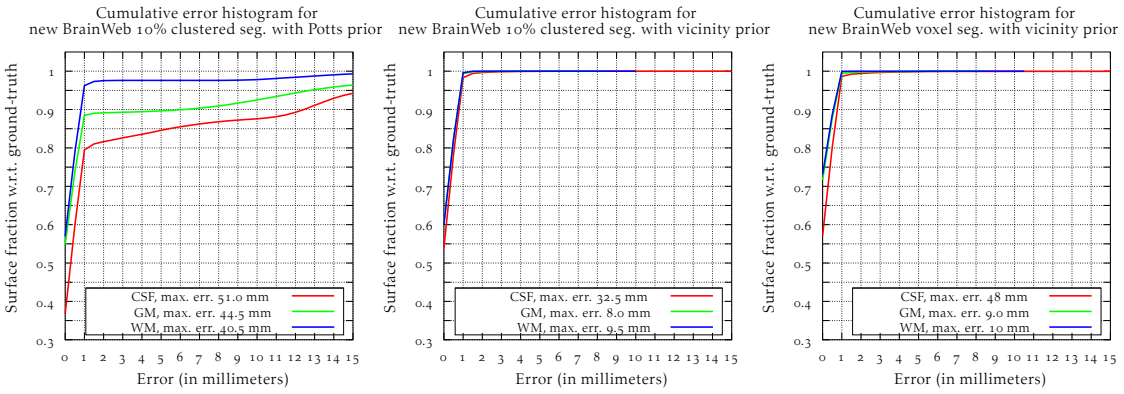


Figure 5.11: Surface error measurements for (a) Potts and (b) vicinity prior segmentation applied to the 10%-clustered second generation BrainWeb simulated MR image, and (c) vicinity prior segmentation applied to the voxels of the image.

Tables 5.2a and 5.2b give overall as well as individual-class DSM, sensitivity and specificity measures for cluster and voxel segmentations with either prior. In these tables, vicinity prior mean performance measures indicating statistically significant improvements at the level $\alpha = 0.01$ are highlighted. Qualitative validation and comparison of cluster and voxel segmentations is given in Figures 5.12 and 5.13 respectively on the 3 principal cross-sections of ground-truth, Potts prior and vicinity prior-segmented images. Disagreements between ground-truth and Potts prior segmentations are encircled. Figure 5.14 gives 3D views of simplified surface meshes generated from different brain and head tissue volumes of the vicinity prior-segmented image via the mesh simplification method outlined in [VC04]. Runtime and memory footprint figures are reported in Table 5.3.

Lastly, we give the surface error measurements of the aforementioned Potts and vicinity prior-based voxel and clustered image segmentations in Figure 5.11 for the 3 brain tissues. Cumulative error histograms presented in these graphs indicate the fraction of segmented surface lying within a given distance from the true surface. For example, we can see in Figure 5.14b that the vicinity prior-based segmentation applied to the 10% clustered image retrieves more than 95% of brain surfaces with an error of 1 mm, whereas with Potts prior, as shown in Figure 5.14a, the error for the same percentage is well above 15 mm. Moreover, we notice that surface error measurements of vicinity prior segmentation applied to image voxels, given in Figure 5.14c, are very close to the aforementioned measurements of clustered image segmentation with the same prior.

These evaluations confirm that the vicinity prior outperforms the Potts prior in Graph Cut segmentation on the simulated head MR image, and that with 10-fold reduction in image size, we are capable of producing a high-quality result which, for most practical purposes, is virtually identical to that obtained on the full-resolution image with two-fold improvements on runtime and more than three-fold savings of memory.

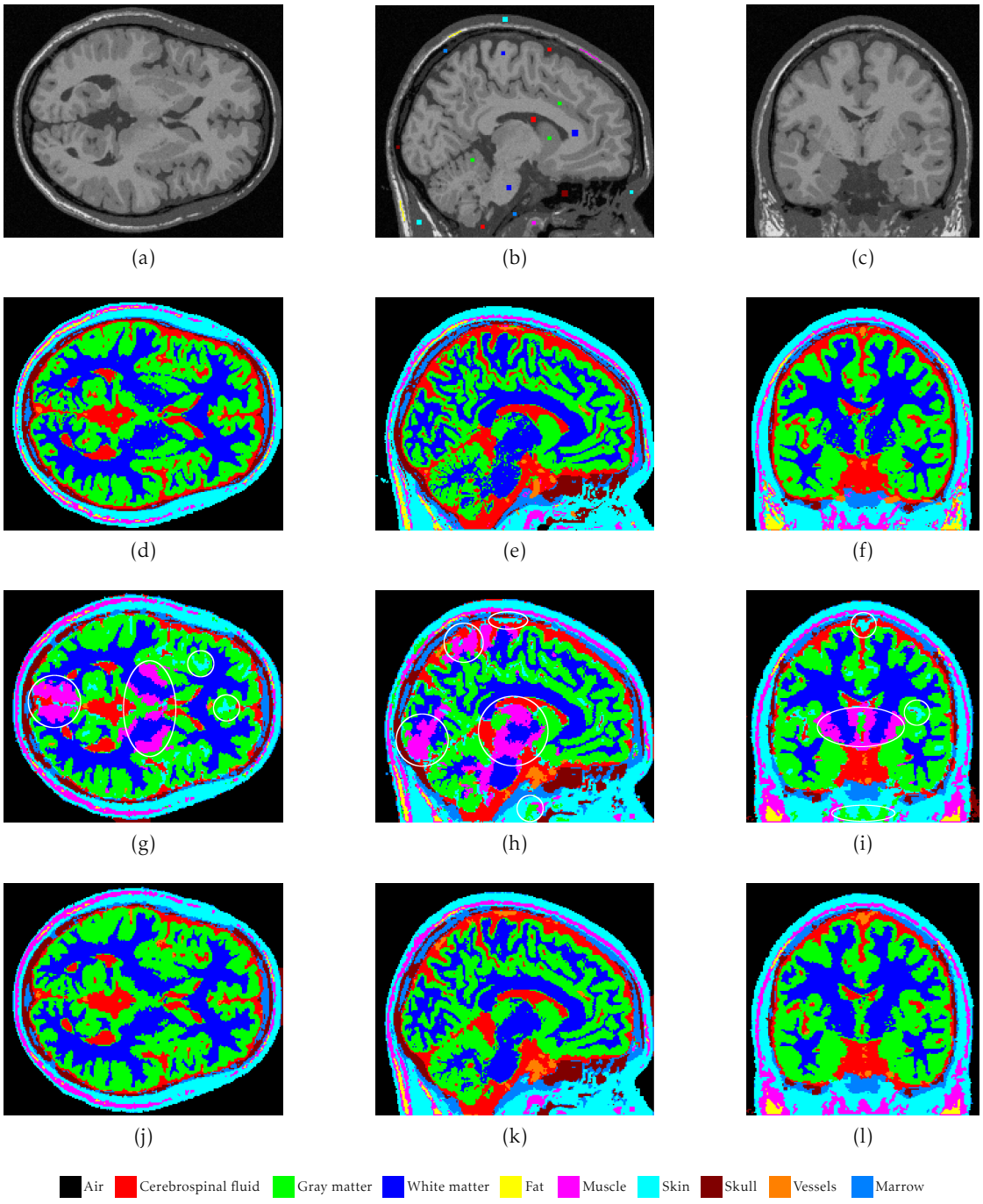
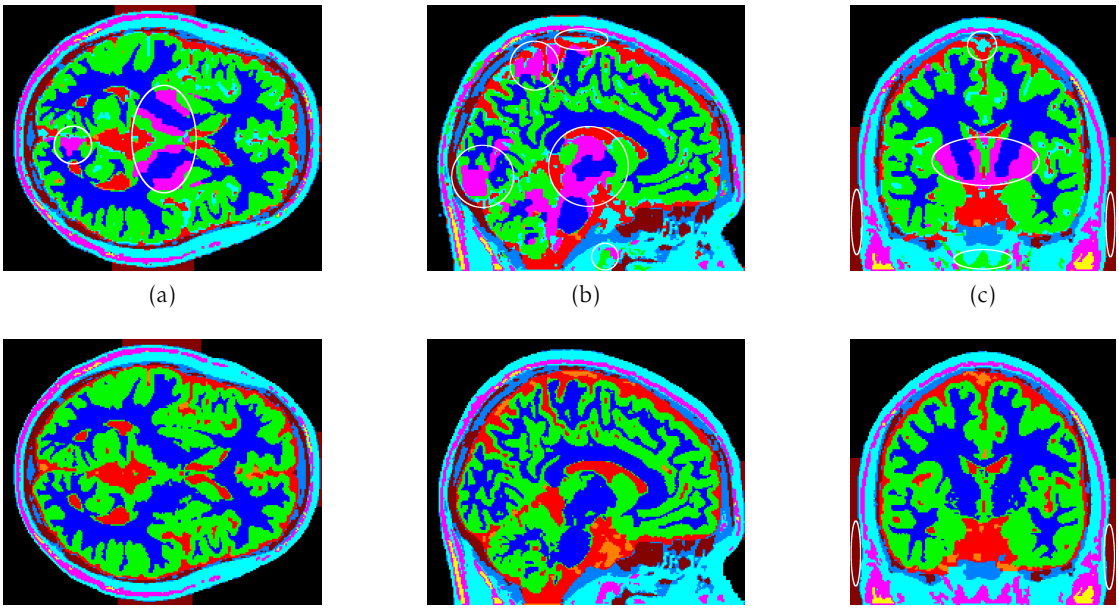


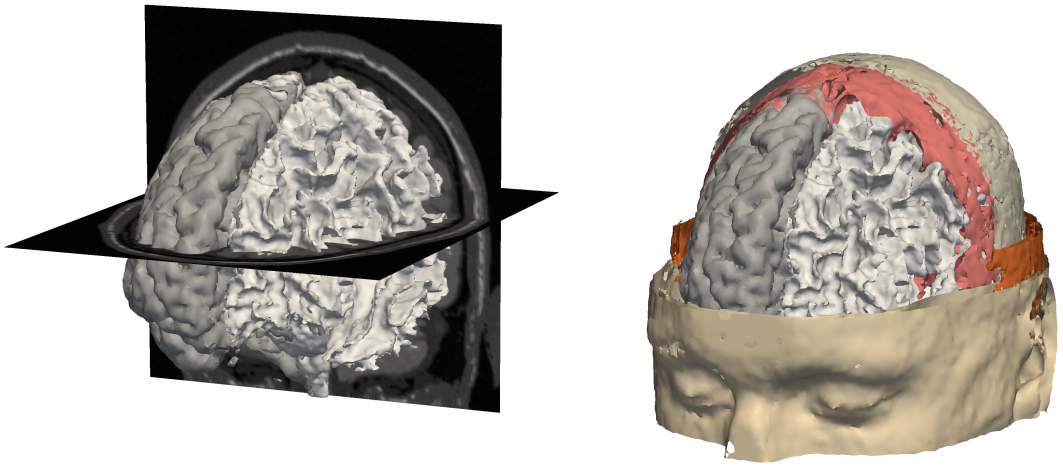
Figure 5.12: Qualitative comparison of Potts and vicinity prior segmentations of the 10%-clustered second generation BrainWeb simulated MR image. Axial (left column), sagittal (middle column) and coronal (right column) cross-sections correspond to (a)–(c) simulated MR, (d)–(f) ground-truth, (g)–(i) Potts prior and (j)–(l) vicinity prior segmented images. Ellipses on (g)–(i) indicate erroneous segmentations with Potts prior. Figure (b) shows seeds introduced by the user in the corresponding sagittal cross-section.



(a) (b) (c)

(d) (e) (f)

Figure 5.13: Qualitative comparison of Potts and vicinity prior segmentations applied to the voxels of the second generation BrainWeb simulated MR image. Refer to 5.12d–5.12f for ground-truth images. Axial (left column), sagittal (middle column) and coronal (right column) cross-sections correspond to (a)–(c) Potts prior and (d)–(f) vicinity prior segmented images. Ellipses indicate erroneous segmentations. Note the similarity between these results and those of segmentation applied to the 10%-clustered image given in 5.12.



(a) (b)

Figure 5.14: 3D views of surface meshes for brain tissues, skin, muscle and skull generated from the 10%-clustered second generation BrainWeb simulated MR image vicinity prior segmentation.

	DSM		Sensitivity		Specificity	
Structures	Potts	vicinity	Potts	vicinity	Potts	vicinity
Air	0.991	0.992	0.985	0.987	0.820	0.875
Cerebrospinal fluid	0.756	0.854	0.669	0.816	0.909	0.933
Gray matter	0.896	0.926	0.884	0.938	0.896	0.924
White matter	0.908	0.937	0.866	0.923	0.897	0.926
Fat	0.638	0.639	0.477	0.473	0.901	0.933
Muscle	0.595	0.740	0.782	0.864	0.899	0.928
Skin	0.845	0.906	0.850	0.886	0.901	0.932
Skull	0.806	0.817	0.790	0.792	0.899	0.932
Vessels	0.205	0.445	0.221	0.520	0.897	0.927
Marrow	0.704	0.750	0.874	0.881	0.895	0.927
mean	0.734	0.801	0.740	0.808	0.891	0.924
Overall	0.898	0.927	0.895	0.926	0.863	0.904

(a)

	DSM		Sensitivity		Specificity	
Structures	Potts	vicinity	Potts	vicinity	Potts	vicinity
Air	0.986	0.985	0.973	0.978	0.829	0.884
Cerebrospinal fluid	0.758	0.867	0.658	0.833	0.909	0.932
Gray matter	0.910	0.940	0.902	0.945	0.893	0.924
White matter	0.929	0.956	0.889	0.942	0.895	0.925
Fat	0.634	0.630	0.467	0.462	0.901	0.934
Muscle	0.614	0.758	0.788	0.909	0.899	0.927
Skin	0.841	0.911	0.850	0.866	0.901	0.936
Skull	0.772	0.778	0.805	0.809	0.898	0.932
Vessels	0.064	0.484	0.057	0.612	0.898	0.928
Marrow	0.707	0.766	0.908	0.917	0.894	0.927
mean	0.722	0.807	0.730	0.827	0.892	0.925
Overall	0.898	0.928	0.895	0.927	0.867	0.909

(b)

Table 5.2: Overall and individual-class performance measures for (a) 10%-clustered and (b) voxel segmentations of the new BrainWeb simulated MR image with Potts and vicinity priors. Significant improvements at $\alpha = 0.01$ are given in red.

Image + size (voxels)	Segmentation	Prior	Clust. (m)	Constr. (m)	Optim. (m)	Mem. (Mb)
BrainWeb subject no. 4 $217 \times 181 \times 217$	voxels	Potts	—	0.23	9.30	3127.67
		vicinity		0.25	12.46	3053.13
	clusters 10%	Potts	3.19	0.21	5.53	870.85
		vicinity		0.18	2.83	876.55

Table 5.3: Performance figures for Potts and vicinity prior segmentation applied to the voxels and a 10% clustering of the second generation BrainWeb simulated MR image presented in Figures 5.13 and 5.12 respectively. Last four columns indicate execution time in minutes for the clustering algorithm, graph construction, Expansion Moves optimization and the program memory footprint in megabytes. Figures are recorded on a desktop computer powered by a quad-core processor running at 2.84 GHz with 6 GB of RAM.

5.5 Real CT images

5.5.1 Thoracic-abdominal CT image

In this subsection, we present a segmentation case study of in vivo acquired thoracic-abdominal 3D CT scan realized during the arterial phase in inhaled position. The image comes from the 3D-IRCADb dataset [IRC12] and is accompanied by ground-truth estimates of 21 anatomical structures, which are basically manual segmentations performed by clinical experts. Of these 21 estimates, we use 16 in our evaluations for the following two reasons. 1) Some of these estimates do not correspond to any particular structure but to many, for example the "skin" tissue class (refer to Figures 5.16d–5.16f) which comprises skin, fat, muscle and visceral fat among other tissues. 2) Some estimates are partial, for example those of intestines for which ground-truth is available only for the colon part. We think that these 16 structures of different size, shape and topology are sufficient for evaluation purposes.

The image volume we use is of size 480×370×167 voxels with 0.961×0.961×1.8 mm voxel size. We segment all visible structures in this image, including those for which no ground-truth estimate is available, applying Graph Cut segmentation with Potts and vicinity priors to 5% and 10% clusterings of the image at a temperature setting of $t = 0.5$. Seeds, illustrated in Figure 5.16a, are introduced by a user in the axial view of a graphical interface by marking few dozens of pixels from any visible target structure in approximately every tenth volume slice. The vicinity prior is defined according to the graph model of 34 structures illustrated in Figure 5.15.

Tables 5.5a and 5.5b give overall as well as individual-class DSM, sensitivity and specificity measures for both clustered segmentations with either prior. In these tables, vicinity prior mean performance measures indicating statistically significant improvements at levels $\alpha = 0.05$ and 0.01 are highlighted with blue and red respectively. Runtime and memory footprint figures for segmentations on both clustering resolutions are reported in Table 5.4. Qualitative validation and comparison is given in Figure 5.16 on the 3 principal cross-sections of ground-truth, vicinity prior and Potts prior-segmented images for the clustering resolution of 5%. Disagreements between ground-truth and Potts prior segmentations are encircled. Notice in particular how right and left lungs are correctly segmented as distinct organs with vicinity prior. Note that, for the reasons mentioned above, ground-truth estimates can be relied on for qualitative inspection only for structures listed in Table 5.5. Figure 5.17 gives 3D views of simplified surface meshes generated via the mesh simplification method of [VC04] from different structure volumes of the vicinity prior-segmented 5%-clustered image.

Lastly, in Figure 5.18 we give the surface error measurements of the aforementioned Potts and vicinity prior-based 5% and 10% clustered image segmentations for 13 structures, excluding adrenal glands and the hyperplastic lesion since they are too small for their surface error measurements to be interesting. Cumulative error histograms of structures for each segmentation are broken into 3 graphs to improve readability. Presented graphs indicate the fraction of segmented surface lying within a given distance from the true surface. We can see that for reasonably low error levels of 0–1 cm,

vicinity prior segmentation retrieves a considerably larger fraction of structure surfaces than its Potts prior counterpart suffering much lower maximum error. Also take note of the similarity between error measurements of vicinity prior segmentations applied to the 5%-clustered and the 10%-clustered image. Nevertheless, vicinity prior segmentation produces some noticeable outliers, namely for the volumes of left lung, heart and spleen. In fact, the CT scanner table in the image is erroneously segmented as part of the body. Such outliers constitute a small percentage of the labeled volume, and can be corrected via "corrective" seeds, as mentioned in Section 2.2.1, or by a separate modeling of the scanner table with its own set of labels and pairwise relations with other structures in the image. Note that surface error measures alone do not constitute an indicator of segmentation quality. Consider, for example, the case of erroneous segmentation of lungs with the Potts prior in Figure 5.16h, where both lungs are segmented as the left lung. Corresponding error graphs in Figure 5.18a indicate lower surface error for the right lung whereas its DSM entry in Table 5.5a indicates little agreement with the corresponding ground-truth estimate. The higher surface error for the left lung in Figure 5.18a is due to most of the right lung volume being segmented as left lung, and consequently its surface having larger per-point minimum distance (5.8) with respect to the left lung ground-truth surface.

Like many Graph Cut approaches, our generic method is prone to shrinking bias affecting thin elongated structures. Take note of fragmentary abdominal and hepatic arteries and portal veins in Figure 5.17b. This is due to the sum of pairwise costs along the boundary of a vessel being higher than the cost of a shortcut across it when likelihood terms do not incur a strong preference for the vessel's class. A number of solutions have been proposed for this problem in Graph Cut segmentation. [KB05] segment vessels using surface functionals combining image-gradient flux and image-based Riemannian area, [VKR08] improve a result with user-supplied connectivity constraints indicating object regions which need to be connected, [PWL11] propose a Hessian-based "vesselness" likelihood for segmenting hepatic arteries and portal veins in CT.

We conclude this section recalling that quantitative and qualitative evaluation confirm the significant improvements brought by the vicinity prior to this instance of multiobject segmentation even when it is applied to a coarsely clustered image, and that doubling the clustering resolution does not improve the quality but comes at the cost of doubling the runtime and increasing memory footprint by one and a half times.

Image + size (voxels)	Segmentation	Prior	Clust. (m)	Constr. (m)	Optim. (m)	Mem. (Mb)
3D-IRCADb CT 480 × 370 × 167	clusters 5%	Potts	19.01	1.65	26.45	2379.13
		vicinity		1.77	33.39	2357.96
	clusters 10%	Potts	12.70	2.08	67.56	3883.37
		vicinity		2.07	106.68	3731.72

Table 5.4: Performance figures for Potts and vicinity prior segmentation applied to the 5% and 10%-clustered 3D-IRCADb thoracic-abdominal CT image. Last four columns indicate execution time in minutes for the clustering algorithm, graph construction, Expansion Moves optimization and the program memory footprint in megabytes. Figures are recorded on a desktop computer powered by a quad-core processor running at 2.84 GHz with 6 GB of RAM.

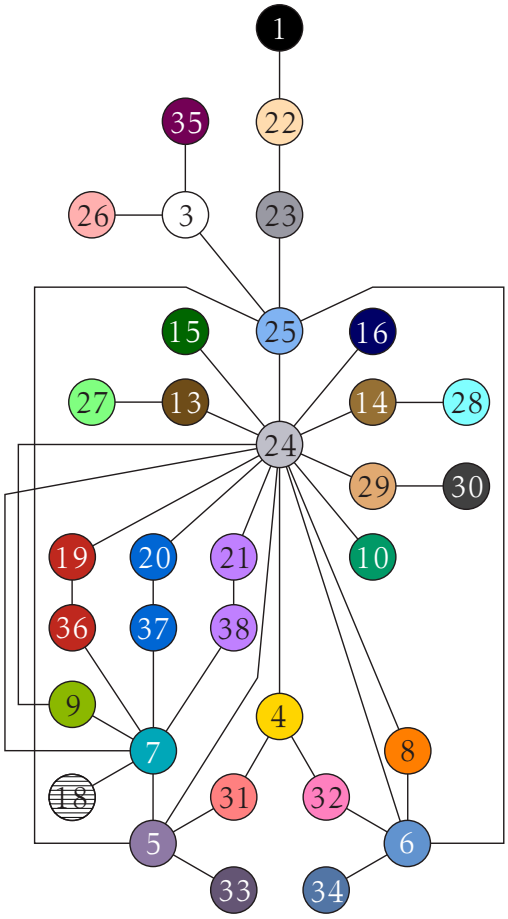


Figure 5.15: Graph-based anatomical model of structure adjacency relations for the 3D-IRCADb thoracic-abdominal CT image. Structures corresponding to graph vertices: 1. air 3. bone 4. heart 5. lung (R) 6. lung (L) 7. liver 8. spleen 9. gallbladder 10. pancreas 13. kidney (R) 14. kidney (L) 15. adrenal gland (R) 16. adrenal gland (L) 18. hyperplasia 19. aorta 20. vena cava 21. portal vein 22. skin 23. fat 24. visceral fat 25. muscle 26. marrow 27. renal pyramids (R) 28. renal pyramids (L) 29. digestive tract tissue 30. digestive tract cavity 31. pulmonary arteries (R) 32. pulmonary arteries (L) 33. bronchi (R) 34. bronchi (L) 35. intervertebral discs 36. hepatic arteries 37. hepatic vena cava 38. hepatic portal veins.

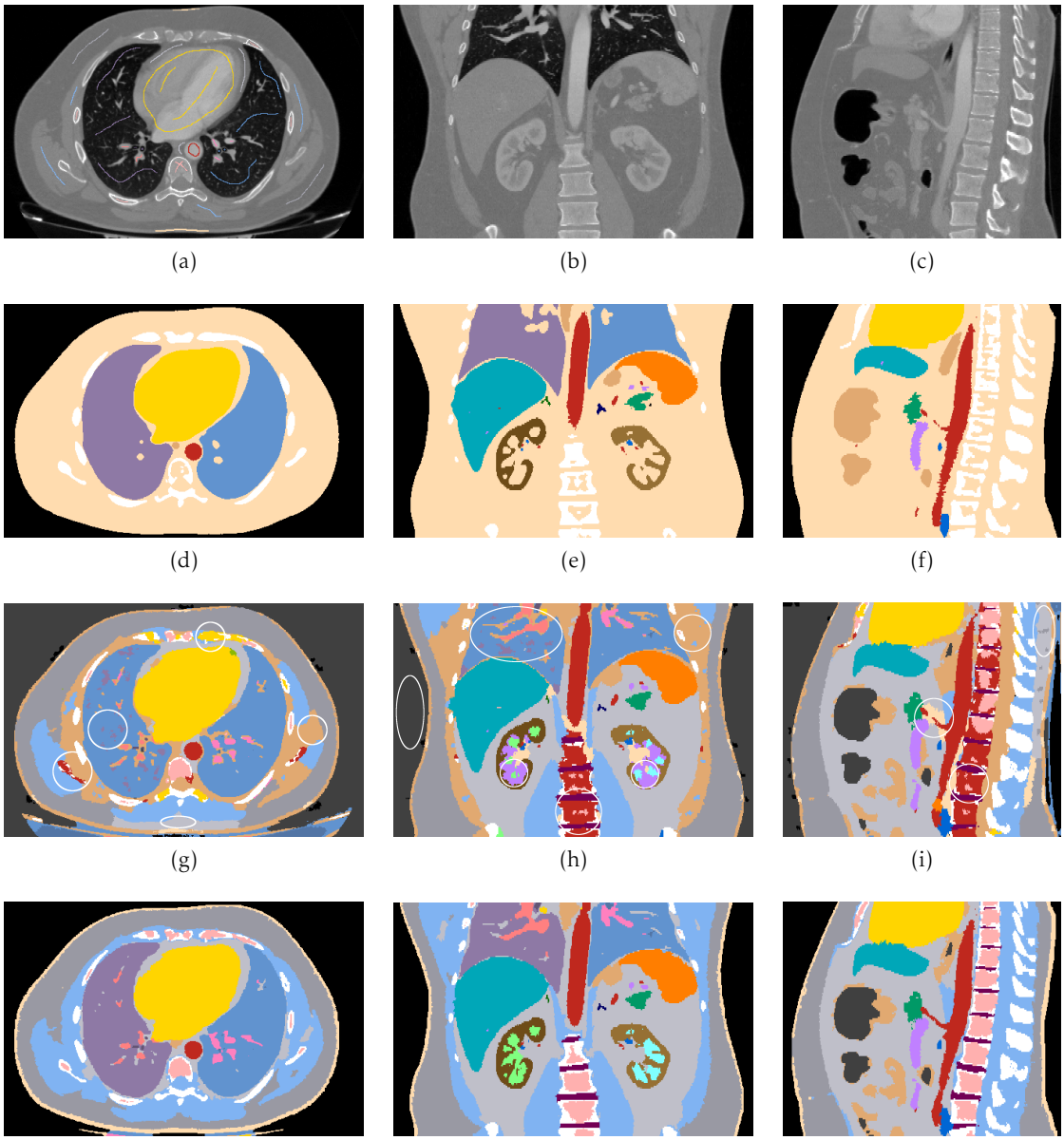
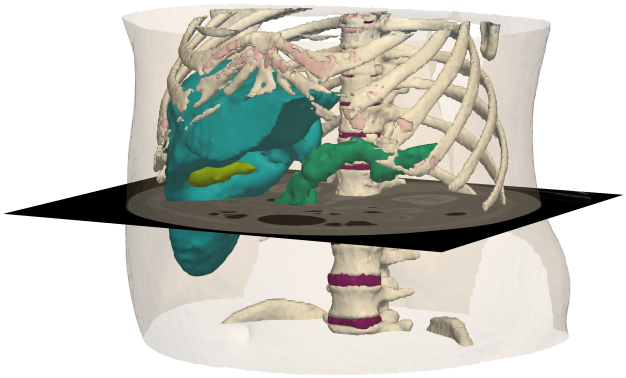
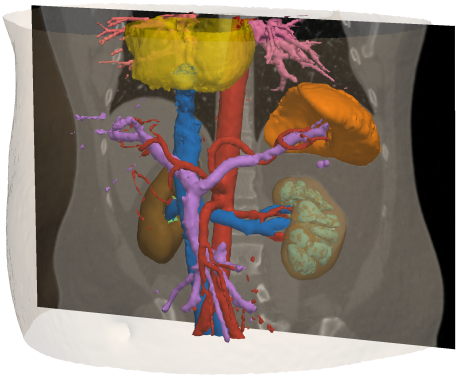


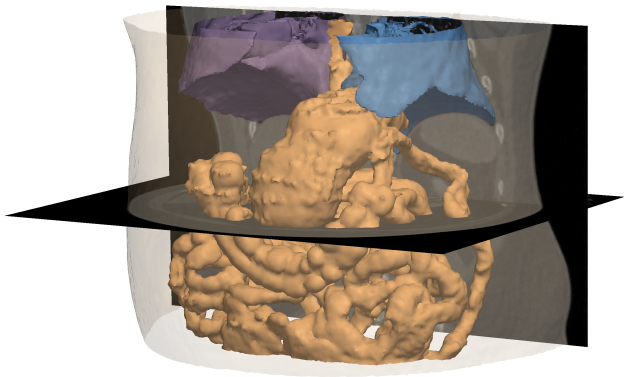
Figure 5.16: Qualitative comparison of Potts and vicinity prior segmentations of the 5%-clustered 3D-IRCADb thoracic-abdominal CT image. Axial (left column), coronal (middle column) and sagittal (right column) cross-sections correspond to (a)–(c) CT, (d)–(f) ground-truth, (g)–(i) Potts prior and (j)–(l) vicinity prior segmented images. Refer to the color code in Figure 5.15 or the color legend in Figure 5.17. Ellipses on (g)–(i) indicate erroneous segmentations with Potts prior. Notice how right and left lungs were correctly segmented as distinct organs with vicinity prior. Figure (a) shows seeds introduced by the user in the corresponding axial cross-section.



(a)



(b)



(c)

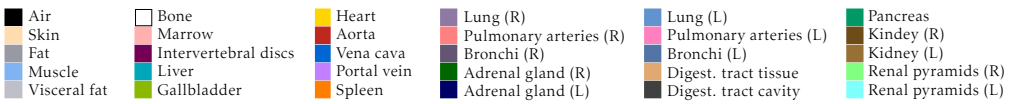


Figure 5.17: 3D views of surface meshes for selected structures generated from the 5%-clustered 3D-IRCADb thoracic-abdominal CT image vicinity prior segmentation.

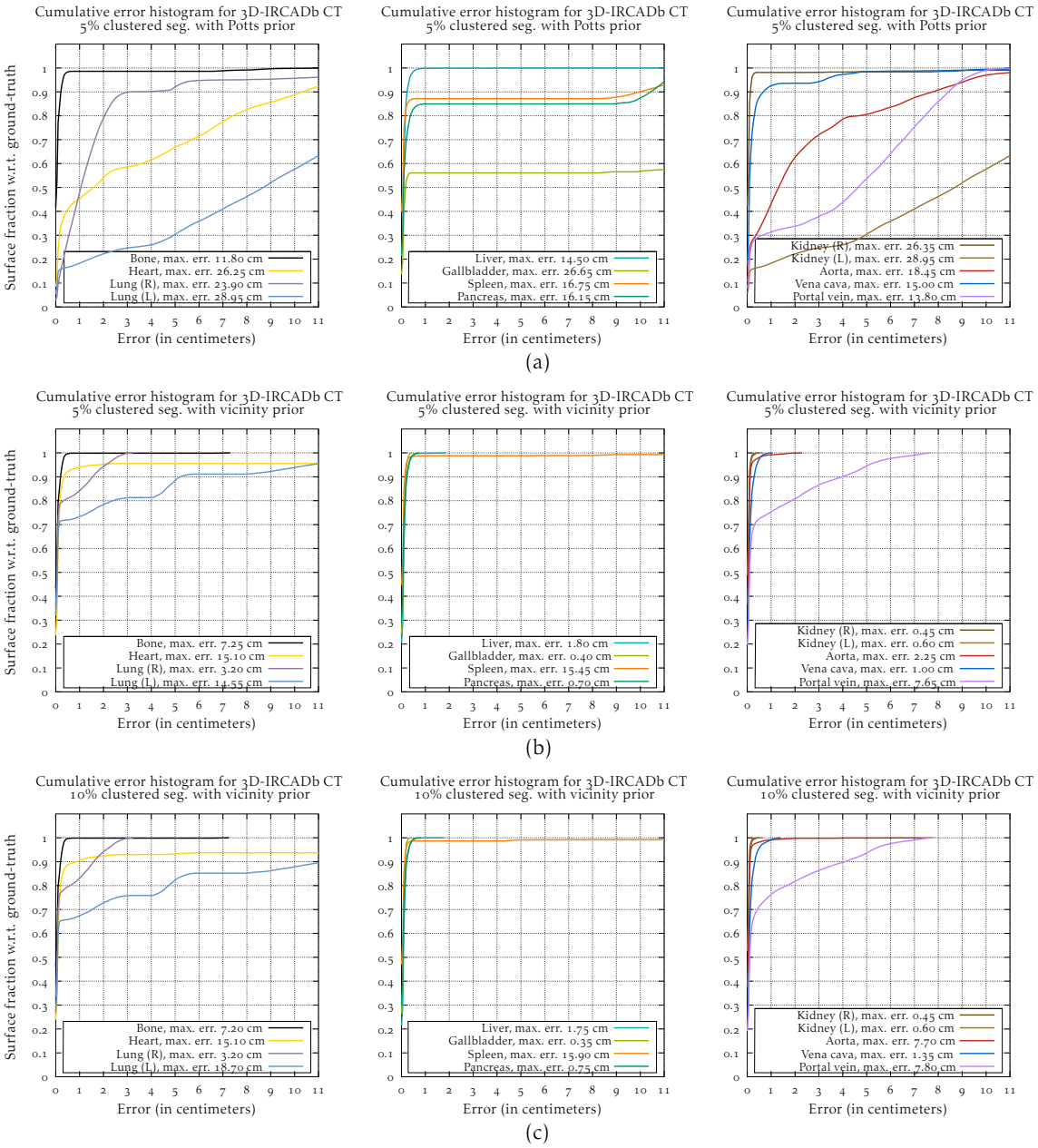


Figure 5.18: Surface error measurements for (a) Potts and (b) vicinity prior segmentation applied to the 5%-clustered 3D-IRCADb thoracic-abdominal CT image, and (c) vicinity prior segmentation applied to the 10%-clustered image. For each segmentation, structures are distributed over 3 graphs to improve readability.

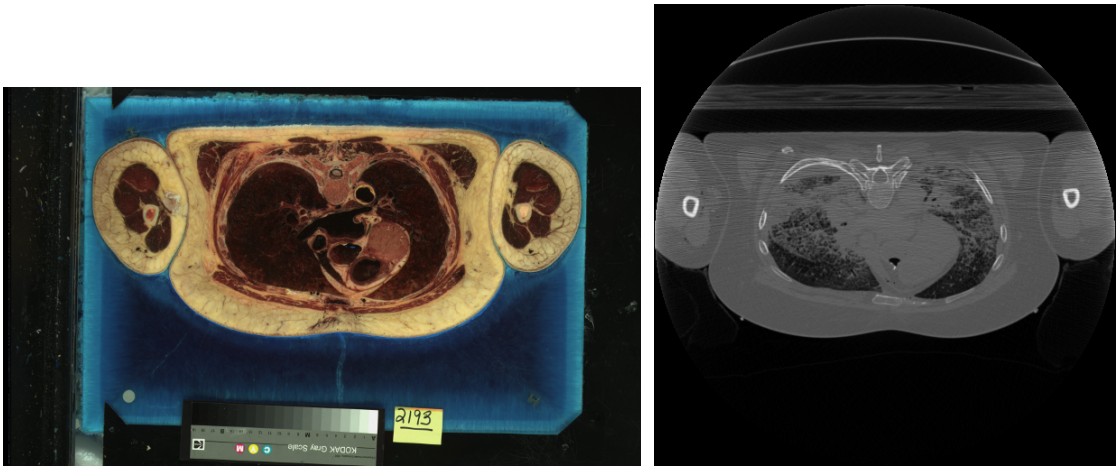
	DSM		Sensitivity		Specificity	
Structures	Potts	vicinity	Potts	vicinity	Potts	vicinity
Bone	0.735	0.857	0.581	0.750	0.768	0.964
Heart	0.918	0.988	0.930	0.978	0.725	0.932
Lung (R)	0.096	0.976	0.050	0.953	0.886	0.932
Lung (L)	0.646	0.980	0.930	0.961	0.709	0.931
Liver	0.990	0.989	0.993	0.990	0.635	0.912
Spleen	0.965	0.982	0.958	0.965	0.732	0.934
Gallbladder	0.902	0.923	0.931	0.932	0.743	0.936
Pancreas	0.895	0.959	0.860	0.940	0.742	0.936
Kidney (R)	0.907	0.967	0.831	0.938	0.742	0.936
Kidney (L)	0.880	0.968	0.786	0.939	0.742	0.935
Adrenal gland (R)	0.765	0.743	0.626	0.595	0.743	0.936
Adrenal gland (L)	0.706	0.708	0.546	0.546	0.744	0.936
Hyperplasia	0.645	0.646	0.486	0.486	0.744	0.937
Aorta	0.513	0.950	0.869	0.908	0.741	0.936
Vena Cava	0.894	0.923	0.862	0.894	0.742	0.937
Portal vein	0.537	0.801	0.700	0.708	0.744	0.938
mean	0.750	0.898	0.746	0.750	0.743	0.936
Overall	0.721	0.962	0.743	0.936	0.730	0.931

(a)

	DSM		Sensitivity		Specificity	
Structures	Potts	vicinity	Potts	vicinity	Potts	vicinity
Bone	0.726	0.859	0.571	0.753	0.755	0.963
Heart	0.891	0.988	0.891	0.978	0.715	0.931
Lung (R)	0.051	0.974	0.026	0.949	0.876	0.932
Lung (L)	0.637	0.977	0.921	0.955	0.695	0.931
Liver	0.991	0.989	0.994	0.992	0.616	0.910
Spleen	0.964	0.984	0.960	0.970	0.718	0.933
Gallbladder	0.941	0.942	0.961	0.961	0.730	0.935
Pancreas	0.904	0.960	0.853	0.937	0.729	0.935
Kidney (R)	0.870	0.971	0.772	0.946	0.729	0.935
Kidney (L)	0.787	0.972	0.722	0.946	0.730	0.935
Adrenal gland (R)	0.733	0.746	0.583	0.597	0.730	0.935
Adrenal gland (L)	0.634	0.642	0.473	0.473	0.731	0.936
Hyperplasia	0.674	0.677	0.516	0.516	0.731	0.936
Aorta	0.480	0.937	0.845	0.884	0.728	0.936
Vena Cava	0.891	0.918	0.858	0.887	0.728	0.936
Portal vein	0.463	0.794	0.712	0.709	0.730	0.938
mean	0.727	0.896	0.729	0.841	0.729	0.935
Overall	0.706	0.962	0.730	0.935	0.716	0.930

(b)

Table 5.5: Overall and individual-class performance measures for (a) 5%-clustered and (b) 10%-clustered segmentations of 3D-IRCADb thoracic-abdominal CT image with Potts and vicinity priors. Significant improvements at $\alpha = 0.05$ and 0.01 are given in blue and red respectively.



(a) (b)

Figure 5.19: Images from the Visible Human female dataset. (a) a cryosectional photo at the heart level and (b) the corresponding CT slice.

5.5.2 Visible Human female CT

In this section, we present a case study of complete skeleton segmentation in a high-resolution CT image reconstructed from the Visible Human female dataset. Due to good contrast, CT is frequently the imaging modality of choice for bone structures.

The Visible Human dataset is the result of an initiative by the U.S. National Library of Medicine (NLM) to create a detailed digital image library of the normal adult male and female anatomy [SASW96]. The dataset comprises radiographs, axial and coronal MR and axial CT images and high-resolution cryosectional photographs all acquired on donated bodies of two deceased male and female individuals, considered representative of an evaluated population. All images in both male and female datasets are provided in the convenient PNG format as 24-bit RGB, for cryosections, and as 16-bit grayscale, for radiological images. A representative view is given in Figure 5.19.

Since its creation, the Visible Human dataset has generated significant interest in medical, image processing and computer vision research communities, in particular for the creation of anthropomorphic phantoms for applications in health physics and nuclear medicine, such as whole-body and individual-organ radiation dosimetry for radiation protection, radiotherapy and medical imaging. Computational procedures of such studies usually involve Monte Carlo simulations to track radiation interactions and energy deposition in the body. The accuracy of such simulations largely depend on the precision and the realism of the anatomical representation, especially for internal organs. Among digital phantoms created from the Visible Human male dataset, we mention the NURBS-based cardiac-torso (NCAT) phantom [Seg01] and voxel-based VIP-Man [XCB00] and Voxel-Man [PHP+01] phantoms. Me mention that all of these phantoms were created from cryosectional photographs mostly by manual contouring and volume editing complemented by color-space classification and thresholding.

(a) (b) (c)

Figure 5.20: (a) An axial cross-section of femoral head and hip bones from the Visible Human male dataset, bones segmented (b) with and (c) without contrast function weighted Potts prior.

We choose to segment only the complete skeleton from Visible Human female CT dataset scanned from the fresh cadaver for two reasons. 1) We have not attempted to segment any other structures because of the lack of ground-truth estimates to validate any results on soft tissue and internal organs, as the aforementioned phantoms were created from the Visible Human male dataset. The skeleton, on the other hand, especially a 3D surface representation of it, is relatively easy to validate by qualitative inspection. 2) Slices from the Visible Human male CT dataset have variable thickness and are missing 4 slices from just superior to the knees to superior to the ankles as well as an one from the abdominal area. The Visible Human female CT dataset, which comes in 1784 slices, is complete and has uniform 1 mm thickness throughout. Pixel size is variable though, ranging from 0.488281 mm in the head and neck area to 0.9375 mm in the torso. We reconstruct a $512 \times 512 \times 1784$ 3D volume from 2D slices by rescaling and padding images and setting voxel size to $0.9375 \times 0.9375 \times 1$ mm. Given computer memory resources at our disposal, segmentation applied to the voxels of this image is an impossible task, therefore, we would also like to demonstrate the possibility of accomplishing it by clustering the image first.

Naturally, we formulate skeleton segmentation as a binary Bayesian labeling problem and obtain the best segmentation with respect to user-supplied evidence and constraints by minimizing the energy (4.7). The set of labels is $L = \{0, 1\}$, where "1" represents the sought object, that is, all bone structures, and "0" the background, which corresponds to everything else in the image. For pairwise energy terms, the Potts prior suffices to ensure the spatial consistency of this instance of binary labeling. We weight pairwise terms, in addition to the cluster common boundary length mentioned in Section 4.4.2, by a smooth contrast function the value of which is inversely proportional to the mean intensity difference between neighbouring clusters. It is defined as:

$$f(I_i - I_j) = e^{-\frac{1}{2}\left(\frac{I_i - I_j}{\sigma}\right)^2} \tag{5.9}$$

where I_i and I_j are the intensities of clusters $i, j \in \mathcal{S}$ calculated from mean intensities of their voxels respectively and σ can be estimated as the sample standard deviation from the image so that the higher it is, the more the segmentation result is spatially

regularized and its boundaries are smooth. Its effect is demonstrated in Figure 5.20. The data terms of the segmentation energy (4.7) follow the definition in (4.8) on the set of labels $L = \{0, 1\}$. Thus, the segmentation is cast as a binary optimization problem which can be solved exactly, as shown in Section 2.4.1, by a single application of a maximum flow algorithm to find the minimum cut of a directed weighted graph representing the energy (4.7). In this case, the graph corresponds to the CVT dual graph on image clusters $\tilde{\mathcal{G}} = \langle \mathcal{S} \cup \{s, t\}, \tilde{\mathcal{E}} \rangle$ with the addition of two terminal nodes. We follow the graph construction method presented in Section 2.4.1. For completeness, the edge weights of this construction are summarized in Table 5.6. To obtain the minimum cut of the graph thus defined, we use the maximum flow algorithm proposed by [BK04], which we discussed in Section 2.3.1. Even though its authors do not give an asymptotic bound on the running time, the algorithm has been empirically shown to outperform standard Push-relabel algorithms, generally regarded as the fastest by the combinatorial optimization community, on vision tasks formulated as labeling problems.

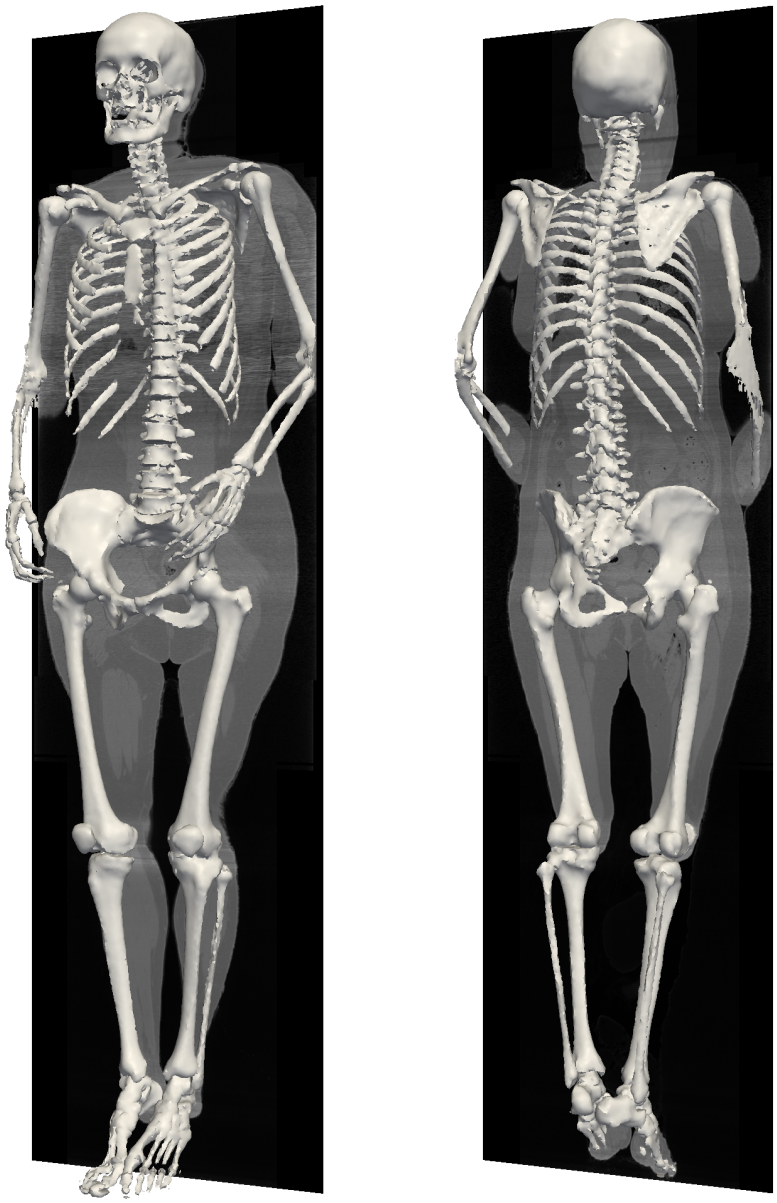
Two views of surface meshes generated via the mesh simplification method of [VC04] from a skeleton segmentation of the 10%-clustered reconstructed 3D CT volume are given in Figure 5.21. Runtime and memory footprint measures are given in Table 5.7. Due to insufficient memory resources, we are unable to run our segmentation program on image voxels to draw comparisons with the clustered variant.

edge	weight	
(i, j)	$\lvert\partial C_i \cap \partial C_j\rvert e^{-\frac{1}{2}\left(\frac{I_i - I_j}{\sigma}\right)^2}$	$\{i, j\} \in \tilde{\mathcal{E}} \vee \{j, i\} \in \tilde{\mathcal{E}}$
(s, i)	0	$\exists h^k \in C_i, k = \text{“1”}$
	∞	$\exists h^k \in C_i, k = \text{“0”}$
	$-t \ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i = \text{“1”})$	$\nexists h^k \in C_i, \forall k \in L$
(i, t)	0	$\exists h^k \in C_i, k = \text{“0”}$
	∞	$\exists h^k \in C_i, k = \text{“1”}$
	$-t \ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i = \text{“0”})$	$\nexists h^k \in C_i, \forall k \in L$

Table 5.6: Definitions of maximum flow graph edge weights for Visible Human female CT skeleton segmentation.

Image + size (voxels)	Segmentation	Clust. (m)	Constr. (m)	Optim. (m)	Mem. (Mb)
VH CT female	voxels	—	N/A	N/A	N/A
$512 \times 512 \times 1784$	clusters 10%	157.93	9.03	21.34	50277.43

Table 5.7: Performance figures for Potts prior segmentation applied to 10%-clustered reconstructed CT image of the Visible Human female dataset presented in Figure 5.21. Last four columns indicate execution time in minutes for the clustering algorithm, graph construction, Maximum flow optimization and the program memory footprint in megabytes. Figures are recorded on a dedicated cluster computer powered by a 64-core processor running at 2.127 GHz with 128 GB of RAM. Due to insufficient memory resources we are unable to apply segmentation to the voxels of the image.



(a) (b)

Figure 5.21: Lateral ventral and dorsal 3D views of surface mesh generated from Potts prior skeleton segmentation of a 10%-clustered reconstructed CT image from the Visible Human female dataset.

5.6 Runtime evaluations

We conclude this chapter with evaluations of image clustering, graph construction and Expansion Moves numerical optimization runtimes giving also total runtimes and global program memory footprint figures in terms of the maximum resident size of the process. The clustering step would obviously be less interesting should the average time to cluster an image and segment it subsequently be longer than applying segmentation to the image's voxels directly, regardless of the higher memory footprint. In Figures 5.22, 5.23 and 5.24 respectively, we give evaluation results for three images of increasing size; a 512×512 pixel 2D image from the synthetic phantom dataset (Section 5.3), the $217 \times 181 \times 217$ voxel simulated MR image from the second generation BrainWeb dataset (Section 5.4.2) and the $480 \times 370 \times 167$ voxel thoracic-abdominal CT image from the 3D-IRCADb dataset (Section 5.5.1). For Expansion Moves optimization step, the temperature hyperparameter was set to $t = 0.5$ for all executions.

We can see from the bar graphs in Figures 5.22, 5.23 and 5.24 that the clustering runtime is inversely proportional to the clustering resolution, as previously encountered in Section 4.3.2. To see how this follows from the asymptotic bound on the clustering algorithm runtime we gave in Section 4.3.1, recall that the connectivity of the dual graph of the CVT is inversely proportional to the clustering resolution, as shown in Figure 2.2, and that higher cluster densities yield shorter mean cluster common-boundary length. Graph Cut data structure construction and optimization runtime and program memory footprint figures are, unsurprisingly, directly proportional to the clustering resolution. We note that graph construction on relatively high clustering resolutions (30%–) takes as long as or longer than graph construction for voxel segmentation because of the longer calculation time involved in evaluating segmentation energy data terms for image clusters from its constituent voxels (4.8).

Total runtime bar graphs in Figures 5.22, 5.23 and 5.24 confirm that the clustering step does indeed accelerate the entire segmentation process compared to an application on the image's voxels directly. Thus, for practical clustering resolutions in the range [5%–10%], two to ten-fold improvements in total runtime can be observed, with better speedups on higher-resolution images naturally. On such images, up to ten-fold memory savings can be achieved.

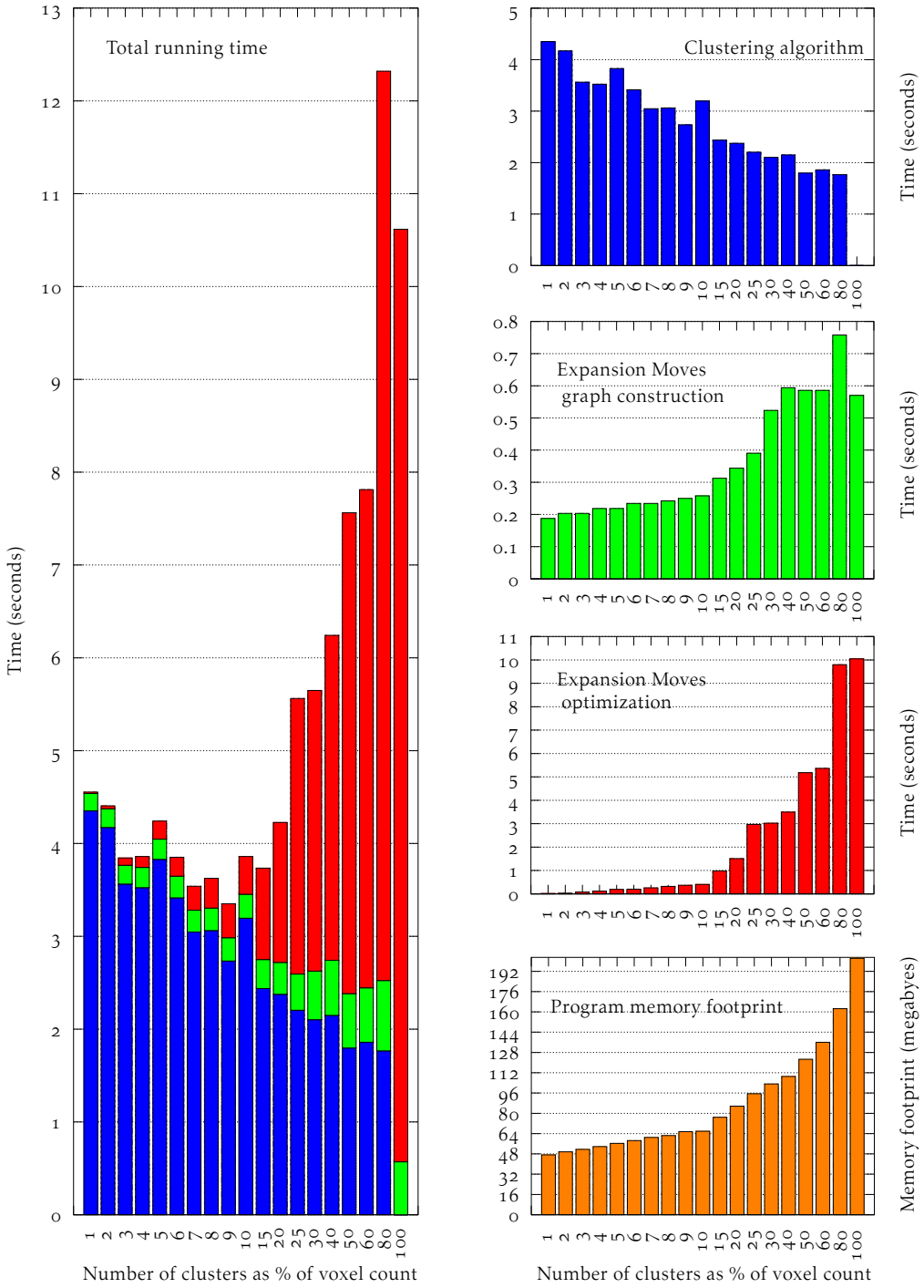


Figure 5.22: Performance evaluation for segmentation program total and individual step runtimes measured on a 512×512 pixel image from the synthetic phantom dataset generated at $\sigma = 0.10$. Figures are recorded on a desktop computer powered by a quad-core processor running at 2.84 GHz with 6 GB of RAM.

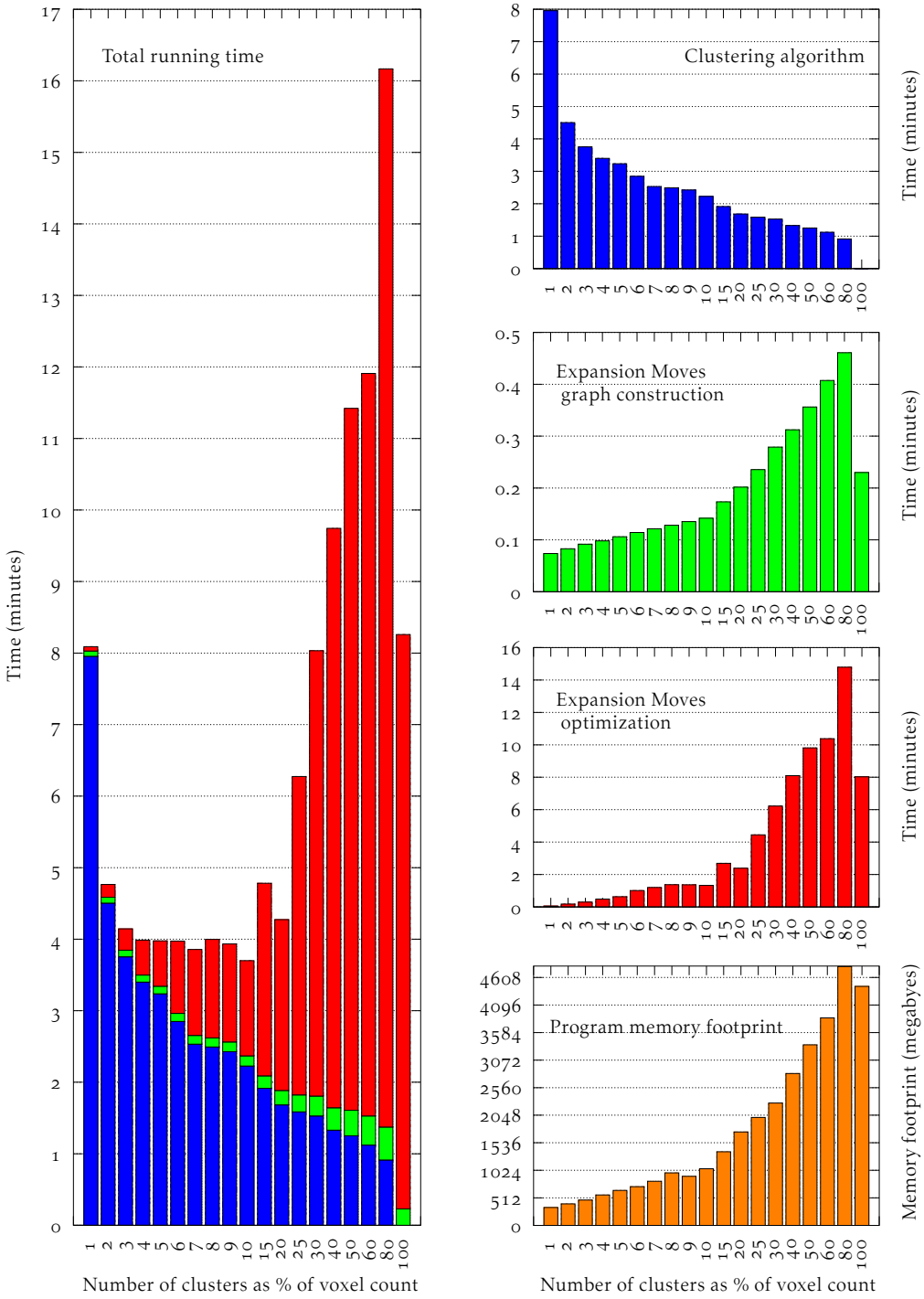


Figure 5.23: Performance evaluation for segmentation program total and individual step runtimes measured on the 217×181×217 voxel second generation BrainWeb simulated MR image. Figures are recorded on a dedicated cluster computer powered by a 24-core processor running at 2 GHz with 47 GB of RAM.

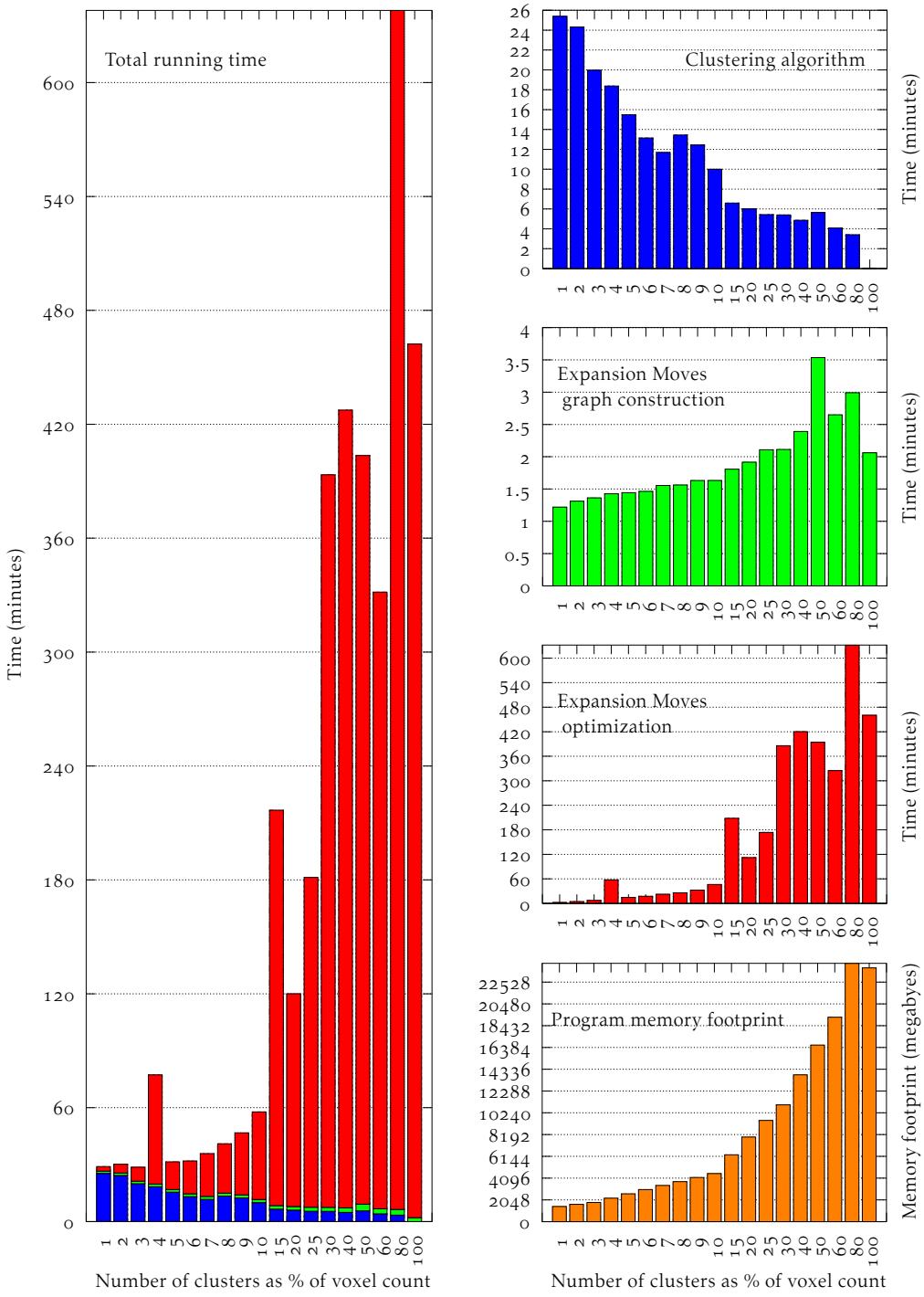


Figure 5.24: Performance evaluation for segmentation program total and individual step runtimes measured on the 480 × 370 × 167 voxel 3D-IRCADb thoracic-abdominal CT image. Figures are recorded on a dedicated cluster computer powered by a 16-core processor running at 2.659 GHz with 47 GB of RAM.

5.7 Discussion and conclusions

Qualitative and quantitative evaluation on synthetic, simulated and real medical images in this chapter confirmed the advantages of our vicinity prior model over the standard Potts model for Graph Cut multiobject segmentation counting up to a few dozen of objects. Improvements brought by our model were remarkable particularly in the correct segmentation of distinct objects having identical intensities, the accurate placement of structure boundaries and the robustness of segmentation with respect to clustering resolution. These results are also important from a theoretical perspective. Recall that the Expansion Moves algorithm's bound on local minima is equal to twice the ratio of the largest nonzero pairwise potential to the smallest (2.13). Even though the Potts model yields the best bound, equal to 2, it was outperformed by the vicinity prior in all evaluations, which has a worse bound in all cases. For example, the vicinity prior defined after the graph model in Figure 5.15 gives a bound of 12.

All graph models of adjacency that were used to derive vicinity priors were defined manually on per-dataset basis according to visible anatomy in the image and the chosen level of detail. As suggested in Section 3.5, such graph models could also be learned from training data or several model variants. Increasing the detail of a model simply amounts to representing new structures, in particular those on finer scales, by introducing new labels and associated graph vertices, and properly defining their neighbourhood with respect to other vertices. In this way, it would have been possible, for example, to represent right and left ventricles in brain MRI segmentation case studies.

Evaluations of runtime and memory footprint confirmed that the clustering step is a worthwhile undertaking, particularly for higher-resolution 3D images. Compared to the voxel-based approach, it accelerates the total segmentation process and reduces memory overhead remarkably. The robustness of vicinity prior segmentation with respect to clustering resolution means that in general there are no advantages of applying segmentation on image voxels or fine-grained partitions, especially that in this case the labeling space has a higher dimension and potentially many more local minima that are hard to escape. In practice, the clustering resolution can be set according to a predefined budget based on the storage and processing capacities of computing platforms.

Chapter 6

Perspectives of future work

6.1 Mission restatement

In this work, we have developed a general Graph Cut-based semiautomatic multiobject image segmentation method principally for use in routine medical applications ranging from moderately difficult tasks involving few objects in 2D images to fairly complex near whole-body 3D image segmentation of few dozens of objects for the creation of subject-specific models. We have shown that the flexible formulation of the method allows its straightforward adaption to a given application context, which does not necessarily have to be medical. In particular, spatial relationships between objects in a given problem instance can be accounted for by a simple redefinition of the general graph-based vicinity prior model we propose, and the segmentation algorithm can be tailored to the runtime requirements of the application and the online storage capacities of the computing platform by an efficient and a controllable clustering of the input image, which can be carried out in offline mode prior to presenting the image to the user for initializing the segmentation algorithm.

In order to save the reader the tedium of reiterating contributions and closely related conclusions already addressed in respective sections of chapters where they were presented and discussed, the current chapter shall point at venues of future research discussing ideas which have not been touched upon. Given the method described in this volume and our experimental setup, future research directions encompass near and long-term objectives.

6.2 Near-term objectives

Such objectives include alternative energy definitions and algorithm implementations which do not require addressing the theoretical properties of the segmentation method.

Combination of energy models. Due to the flexibility of the energy-based theoretical framework of Graph Cut segmentation, different likelihood and prior energy models can be combined. In particular, the vicinity prior can replace the Potts model in some segmentation methods we discussed while presenting related work, such as [VM08,

GTC11] which, respectively, use a shape similarity measure for prior and Gaussian intensity distributions for likelihood-based data energies. In our work, we have used intensity histograms in such data energies since our focus was on the prior formulation. Nevertheless, it is interesting to investigate the influence of more expressive intensity distribution models, such as Gaussian mixtures, even though they would require an appropriate learning support.

Algorithm optimization. The clustered-image Graph Cut segmentation process can be further accelerated by using a parallel implementation of a maximum flow algorithm, typically of the Push-relabel variant [BS05], which would take advantage of symmetric multiprocessor architectures of modern computers. The parallelization of the clustering algorithm is more involved however. We have experimented with a rudimentary parallelization of the algorithm on image subvolumes, but the lack of congruence between clusters at the boundaries of adjacent subvolumes created blocky partition artifacts which were too degrading of the subsequent segmentation quality for the resulting speedup to be worthwhile.

Interactive segmentation. All results reported in this work relied on an implementation that executes in batch mode; user-provided seeds are collected, passed to the segmentation program which performs clustering and Graph Cut optimization writing the segmentation result onto a disk to be read and displayed by a visualization front-end. If the segmentation result does not satisfy the user, he supplies corrective seeds and the program is rerun. This basic mechanism is sufficient for developing and evaluating methods, but not for an interactive segmentation application integrating the user gesture and the segmentation algorithm. Maximum flow algorithms of the Augmenting-paths variant allow a minimum cut to be recomputed efficiently after modifying relatively few edge capacities [KT05]. This property can be taken advantage of in designing an interactive application which allows efficient editing of a current segmentation with the addition or the removal of seeds for correction and refinement purposes. Naturally, parallel maximum flow algorithms would help attain interactive execution rates in such applications.

6.3 Long-term objectives

Such objectives require further research and investigations which entail redesigning algorithms, extending prior models and revisiting the method's theoretical foundations.

Learning prior models. As mentioned previously, adjacency graphs used to derive the vicinity prior were defined manually, which is acceptable for the purposes of method evaluation. In practical applications however, automatic definition of graph models is interesting. For example, such models can be readily generated from existing anatomical phantoms by structure common-surface analysis. In general, automatic definition relies on a learning support of adjacency patterns from presegmented training data, possibly also capturing model variabilities and relation uncertainty, especially with partial data. While graph models of continuous edge weight can represent "degrees" of adjacency, care should be taken for the prior ultimately defined from learned relations

to satisfy submodularity in order for Graph Cut optimization to apply. Priors defined as shortest-path constraints, such as the vicinity prior, do respect this condition, however there remains to be seen whether the formulation is meaningful for variable or different types of relations.

Introducing other relations. It is no doubt interesting to define graph models representing object relationships other than adjacency, such as distance and spatial directionality relationships of the type "object a is on the left of object b". Distance relationships in particular would require a more precise description of the locations and the surfaces of sought objects, therefore we think that prior formulations taking account of object distance would require a learning support as well. Spatial directionality relations rely on orientation information and would call for directed graph structures. The shortest-path formulation on a directed graph however produces asymmetric pairwise constraints which would require revisiting some of the fundamental properties of the underlying theory of Bayesian labeling, in particular, to express the prior distribution according to a directed Markov model [Lau01].

Are vicinity prior-based energies tractable? It is known that the shortest-path length between two vertices defines a natural distance metric on graphs bringing notions of convex functions and convex sets to graphs [BC08]. Therefore, it is alluring to see whether tools of convex analysis could be applied to energies defined on shortest-path constraints for designing optimizers having better runtime, capable of finding better solutions than the Expansion Moves algorithm.

Appendix A

Présentation des travaux

A.1 Résumé

Nous définissons des contraintes sous forme de plus court chemin dans le graphe d'adjacence des structures présentes dans les images. Nous les introduisons dans un cadre commun de partition Voronoï barycentrique d'images et de segmentation semi-automatique multi-objet par coupure de graphes. Le modèle d'a priori de proximité ainsi défini est un modèle constant par morceaux qui introduit plusieurs niveaux de pénalisation et représente la configuration spatiale possible des structures en segmentation multi-objet. Des évaluations qualitatives et quantitatives sur des images médicales synthétiques, simulées et réelles confirment que, contrairement à une approche de segmentation utilisant le modèle de Potts, notre méthode permet la segmentation correcte des structures distinctes ayant des profils d'intensité identiques et améliore la précision du positionnement des frontières de la segmentation tout en assurant sa robustesse par rapport à la résolution de partition. Quant à notre approche de partition d'image qui vise à simplifier l'image avant de la segmenter, elle établit un bon équilibre entre les objectifs concurrents de compacité des régions et de respect des frontières des objets, permettant en outre de contrôler le compromis. Par comparaison avec l'approche de segmentation appliquée directement sur les voxels d'image, l'étape de partition améliore à la fois le temps d'exécution global et l'empreinte mémoire du processus de segmentation jusqu'à un ordre de grandeur sans compromettre la qualité du résultat en pratique.

A.2 Introduction

Traditionnellement adaptées à un seul organe ou une pathologie particulière, les méthodes de traitement d'images récentes ont rendu possible l'analyse et la segmentation simultanées de plusieurs structures anatomiques pour un diagnostic et une planification préopératoire plus complets [SBZ+09, FAB12, OYH+12, LPPS12]. Par ailleurs, les technologies de traitement et de visualisation à distance, pour les grands volumes de données, ont mûri et ouvert la porte à la navigation sémantique de l'anatomie du corps entier [BCK+11]. Ainsi, la création des modèles anatomiques et physiologiques «patient-spécifiques» devient crédible.

La pratique clinique, en particulier la tomodensitométrie (TDM) corps-entier, et la recherche médicale génèrent aujourd'hui un grand nombre d'images 3D de haute résolution, spécifiques aux patients, rendant difficile les tâches d'accès à ces données, de transfert, d'analyse et de visualisation, particulièrement dans les environnements informatiques distribués qui témoignent

d'une utilisation des terminaux mobiles pour la visualisation interactive des données anatomiques. Les images TDM et résonance magnétique (IRM) ont des caractéristiques intrinsèques qui rendent l'analyse automatique particulièrement difficile. En théorie, elles sont considérés comme des cartes d'intensité constantes par morceaux sur un nombre de classes de tissus. Toutefois, le contraste entre les intensités de ces classes, qui facilite la segmentation, dépend des conditions d'acquisition qui sont rarement idéales. Dans la pratique, la propriété de constance par morceaux est considérablement dégradée par différentes sources de bruit et l'effet de volume partiel qui créent des bords diffus rendant difficile l'identification des vraies frontières de structure. Les images RM peuvent, en outre, subir une distorsion spatiale d'intensité des tissus en raison de l'inhomogénéité du champ magnétique. Quels que soient la modalité d'imagerie et les artefacts liés, de nombreuses structures anatomiquement et fonctionnellement distinctes manifestent des niveaux d'intensité identiques et se fondent dans les tissus voisins. Ces problèmes, couplés avec la variabilité des tissus à travers les populations de patients et les grands volumes d'images 3D, rendent irréalisable la conception d'algorithmes de segmentation automatiques d'usage générique. Alors que certaines tâches élémentaires, dans des conditions bien définies, sont déléguées aux ordinateurs pour un traitement automatique, les experts médicaux ont fréquemment recours à des méthodes de segmentation essentiellement manuelles dans la construction des modèles 3D «patient-spécifiques» qui demandent des investissements importants de temps et d'effort [PHP+01, Seg01].

Les méthodes de segmentation semi-automatiques sont habituellement invoquées afin d'alléger le fardeau de la délimitation manuelle et comme une alternative aux méthodes automatiques lorsque celles-ci ne sont pas applicables, par exemple en raison de la singularité du cas étudié ou le manque de bases de données d'apprentissage. En outre, les méthodes automatiques qui s'appuient souvent sur les caractéristiques photométriques et anatomiques apprises à partir des données segmentées, tout en étant efficaces dans de nombreux cas, sont liées aux caractéristiques de la base d'apprentissage et à ces variabilités. Elles ne peuvent pas facilement répondre aux situations non rencontrées au préalable telles que les cas pathologiques et des images acquises dans des conditions différentes.

De nombreuses méthodes reposant sur différents paradigmes d'interaction ont été conçues pour permettre à un utilisateur d'initialiser, de piloter un processus de segmentation et d'affiner son résultat avec une interaction nécessitant un effort raisonnable [MB98, FU00, BJ01, GFL04, VK05]. L'intervention de l'utilisateur fournit aussi des indices, pour obtenir des informations a priori, qu'un algorithme de segmentation peut ensuite exploiter, telles que les statistiques d'intensité, d'emplacement et de disposition spatiale des objets ciblés. En particulier, lorsque la segmentation est formulée comme un problème d'étiquetage des voxels d'image où chaque objet ciblé est représenté par une étiquette, l'interaction de l'utilisateur peut être effectuée par l'attribution de graines à l'intérieur des structures ciblées. Cela fournit des indices à l'algorithme sur ce que l'utilisateur voudrait segmenter et peut être utilisé, par exemple, pour recueillir des statistiques sur l'intensité des objets et leurs positions relatives, et pour contraindre l'espace des solutions de l'algorithme en interdisant l'attribution d'autres étiquettes à des voxels graines.

A.2.1 Segmentation par coupure de graphes et sa complexité

Problème fondamental en traitement d'images et vision par ordinateur, la segmentation d'images est l'épine dorsale de nombreuses applications, en particulier en imagerie médicale. Les méthodes de coupure de graphes, qui sont couramment appliquées aux problèmes de segmentation mono-objet [BF06], s'appuient sur un mécanisme d'optimisation binaire par maximisation de flot qui est capable de trouver la solution optimale [GPS89]. À quelques exceptions

[Isho3], les généralisations multi-objets de cette approche n'ont pas, en règle générale, de telles propriétés d'optimalité en raison de la difficulté NP du problème d'optimisation sous-jacent, mais certains algorithmes [BVZ01] sont néanmoins capables de trouver une solution approchée à une distance connue de l'optimum global. Les approches de segmentation multi-objets simultanées, d'autre part, sont supérieures à leurs homologues séquentielles car elles ne soulèvent pas de questions ni sur la meilleure séquence de segmentation à suivre ni sur la façon d'éviter la propagation des erreurs des segmentations individuelles [FAB12].

En imagérie, les algorithmes d'optimisation par coupure de graphe utilisent des structures de données basées sur la trame rectangulaire de l'image. Dans les applications impliquant des images médicales 3D de haute résolution, le nombre de sommets du graphe est égal au nombre de voxels d'image et le nombre d'arêtes au moins cinq fois supérieur. Pour des images 3D de plusieurs disaines de millions de voxels, la taille de la structure de données dépasse la capacité de stockage de l'ordinateur, l'optimisation numérique devient alors irréalisable.

Les approches qui s'adressent à la complexité des méthodes par coupure de graphe sur des données de haute résolution peuvent être classés en deux catégories: celles qui s'appuient sur la parallélisation des algorithmes sous-jacents et celles qui reposent sur la simplification d'images. Des versions parallèles de certains algorithmes de flot maximum ont été conçues pour des architectures multi-processeurs [DB08] et les unités de traitement graphique [VN08]. Bien que ces méthodes atteignent de bonnes pérformances, elles ne réduisent pas l'empreinte mémoire car elles opèrent sur l'image entière. Les approches de simplification d'images s'adressent à ce problème directement en observant que la représentation pixellique est souvent redondante car les objets sont en général constitués de plusieurs pixels similaires qui peuvent être regroupés. Dans [LSGX05], la segmentation par coupure de graphes est appliquée à une image à basse résolution et la solution est propagée à la pleine résolution où la segmentation est appliquée dans une bande étroite autour de la frontière d'objet. Afin d'accélérer l'interaction avec l'utilisateur, [LSTS04] sur-segmentent l'image via l'algorithme de ligne de partage des eaux. La segmentation par coupure de graphes est alors formulée selon le graphe des régions de la partition. A partir d'une partition d'image en grille, [CA10] regroupent les pixels d'image par un algorithme k-means itératif intégrant des critères de similarité de couleur et de compacité de région. Le graphe de partition est ensuite segmenté produisant une segmentation grossière de l'image. Les approches de segmentation récentes qui reposent sur les champs conditionnels aléatoires telles que [FVS09, LLB+11] appliquent une partition d'image non seulement afin de réduire l'empreinte mémoire, mais aussi pour recueillir des descripteurs d'image à partir des régions et leur voisins.

A.2.2 Informations a priori en segmentation par coupure de graphes

Les méthodes de segmentation les plus performantes intègrent souvent de l'information préalable spécifique au problème ou à l'image. Les méthodes de segmentation qui s'adressent aux images médicales devraient idéalement être en mesure d'intégrer un large éventail d'a priori anatomiques et physiologiques pour assurer la cohérence du résultat par rapport aux propriétés anatomiques et les variabilités de celles-ci. Du point de vue de l'optimisation par coupure de graphe, l'information préalable peut être considérée comme un ensemble de contraintes définies par un modèle de référence.

De nombreuses formes d'a priori ont été introduites dans le cadre de segmentation par coupure de graphes dans différentes applications. Nous classifions ces approches en trois catégories en fonction de la nature de l'a priori utilisé.

a) A priori de position, où des cartes de probabilité spatiale indiquant les positions des structures, également appelées atlas probabilistes, sont utilisées pour guider la segmentation

[STAG06, LPPS12]. Dans les cas les plus simples, celles-ci sont définies comme des cartes de distance [FZ05]. Ces cartes probabilistes doivent être apprises à l'avance et la plupart doivent être recalées sur l'image avant qu'elle ne soit segmenté. Le recalage ne peut pas facilement gérer les changements topologiques, dûs par exemple aux pathologies, et nécessite souvent un calcul intensif.

b) A priori de forme, où des contraintes de forme explicites ou implicites sont utilisées pour la mise en correspondance de l'objet segmenté avec des formes prédéfinies ou apprises. Par exemple, [VM08] utilisent une mesure de similarité invariante aux transformations affines en segmentation multi-objet séquentielle par coupure de graphes. Dans [MRT07], un modèle statistique de forme et ses variabilités est appris à partir d'un ensemble de formes dans une base d'apprentissage par l'intermédiaire de l'analyse des composantes principales. Les approches de segmentation par a priori de forme impliquent une initialisation de la position d'objet et des mécanismes d'alignement avec l'image, et souvent nécessitent un apprentissage des modèles.

c) A priori de configuration spatiale, dont l'utilisation figure le plus souvent dans le contexte de segmentation multi-objet, où des contraintes (généralement par paires) définissent les relations géométriques et spatiales entre les objets. Cet usage est apparu dans la segmentation des scènes où les relations spatiales sont prédéfinies à partir d'un faible nombre d'objets [LVS10], et dans la détection d'objets à partir de ses composantes où ces relations sont apprises des données annotées [WS06, LLB+11]. [DB09] proposent une méthode de segmentation pour plusieurs objets ayant des positions relatives d'inclusion ou d'exclusion, en introduisant aussi des contraintes sur les distances préférées entre les bords des régions d'objet. Cependant certaines configurations pratiques ne peuvent pas être représentées par cette méthode, telle que l'inclusion d'une paire adjacente d'objets dans un troisième.

Il existe un nombre d'approches d'introduction d'a priori, étroitement liées à notre travail, qui sont apparues en dehors de la segmentation par coupure de graphes. [CCB+05] utilisent un modèle d'a priori constant par morceaux à trois niveaux de pénalisation pour la classification des tissus cérébraux dans les images IRM. Le modèle favorise la classification identique des paires de pixels adjacents, permettant les adjacences qui sont compatibles avec le modèle et pénalisant celles qui ne le sont pas. Nous utilisons un modèle similaire dans le cadre de segmentation multi-objet par coupure de graphes [JKD+12]. Dans [SBZ+09], les relations géométriques sont définies par un réseau anatomique sur des repères corporels facilement détectables. Elles sont utilisés dans la phase de détection d'organes thoraciques et abdominaux en TDM corps entier dans un cadre de segmentation multi-objet séquentielle. [OYH+12] abordent le même problème avec une méthode automatique en deux étapes. Les organes relativement stables sont d'abord identifiés et segmentés, puis les relations entre les organes sont utilisées pour procéder à la segmentation des organes restants en recalant des atlas statistiques et des modèles de forme appris au préalable sur l'image. Les relations entre les organes sont définis selon une analyse de corrélation canonique et sont représentées par un graphe orienté. Intuitivement, ces relations représentent le degré d'influence de la surface d'un organe sur un autre. Un modèle de scène générique sous forme de graphe représentant l'information spatiale explicite est également utilisé par [FAB12] pour la segmentation de structures sous-corticales dans les images IRM. Des représentations floues de la distance, l'adjacence et les relations d'orientation spatiale sont utilisées pour tenir compte des imprécisions du modèle et ses variabilités. Cependant, l'algorithme de segmentation est séquentiel et nécessite des stratégies de détection d'erreur et d'adaptation d'ordre de segmentation.

A.2.3 Contribution

En nous appuyant sur les algorithmes d'optimisation par coupure de graphes, nous développons une méthode générique pour la segmentation multi-objet semi-automatique efficace d'images médicales 3D de haute résolution. La méthode peut facilement être adaptée à un problème particulier par une simple reformulation du modèle d'a priori que nous proposons. L'algorithme sous-jacent peut aussi être adapté aux contraintes de temps d'exécution et à la capacité de stockage par une simplification adaptative de l'image à segmenter.

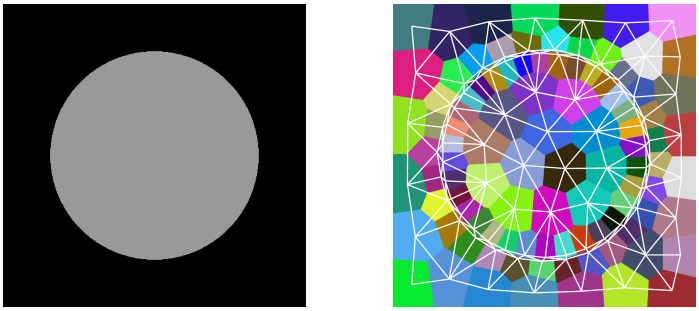
Notre contribution principale est un a priori de structure, appelé «modèle de proximité» qui introduit plusieurs niveaux de pénalisation et représente la configuration spatiale des objets en segmentation multi-objet, permettent, en outre, la minimisation des énergies résultantes à l'aide des algorithmes d'optimisation par coupure de graphes. Il est formulé à partir du plus court chemin entre deux objets dans un graphe décrivant les relations d'adjacence entre les objets.

Nous introduisons le modèle de proximité avec les contraintes fournies par l'utilisateur dans un cadre commun de partition Voronoï barycentrique adaptative d'images et de segmentation semi-automatique multi-objet par coupure de graphes. La méthode de partition d'images que nous utilisons pour les simplifier avant la segmentation établit un bon équilibre entre les objectifs concurrents de compacité des régions et de respect des frontières des objets, permettant, en outre, de contrôler le compromis, et contribuant à la mise en place fine des frontières de la segmentation ultérieure. Nous formulons la segmentation par coupure de graphe selon le graphe des régions Voronoï tel que l'énergie de segmentation est indépendante de la résolution de partition. Ceci constitue notre deuxième contribution. Par comparaison avec l'approche de segmentation appliquée directement sur les voxels d'image, l'étape de partition améliore à la fois le temps d'exécution global et l'empreinte mémoire du processus de segmentation jusqu'à un ordre de grandeur sans compromettre la qualité de segmentation en pratique.

Des évaluations qualitatives et quantitatives, sur des images médicales synthétiques, simulées et réelles de tailles et de contenus différents, confirment les avantages de notre modèle d'a priori de proximité sur celui de Potts et notre modèle d'adjacence élémentaire en segmentation multi-objet par coupure de graphe comptant jusqu'à quelques dizaines d'objets. Les améliorations apportées par notre modèle de proximité sont significatives, en particulier dans la segmentation correcte des objets distincts ayant des intensités similaires, dans le positionnement précis des frontières des structures ainsi que dans la robustesse de segmentation par rapport à la résolution de partition.

A.2.4 Organisation du chapitre

La Section A.3 suivante décrit la partition de Voronoi adaptatif et la compare avec deux algorithmes étroitement liés [LSK+09, VBM10]. L'approche de segmentation statistique est décrite ensuite dans la Section A.4. Dans la Section A.4.1, nous présentons notre modèle d'a priori de proximité suivi par la définition du modèle d'attache aux données dans la Section A.4.2. Nous décrivons brièvement la méthode d'optimisation numérique que nous utilisons dans la Section A.4.3 et établissons la condition nécessaire pour son application. Les résultats des évaluations qualitatives et quantitatives sur des images médicales synthétiques, simulées et réelles sont donnés dans la Section A.5. Dans la Section A.5.1 nous présentons les résultats de l'évaluation sur des images générées à partir de notre fantôme synthétique (Fig. A.4a). Ensuite, dans la Section A.5.2.1 nous donnons les résultats des évaluations quantitatives sur la base de données d'images IRM simulées BrainWeb [CZK+98] et les comparons avec les résultats de classification de tissus cérébraux rapportés dans [GTC11]. Enfin, dans la Section A.5.2.2, nous présentons



(a) (b)

Figure A.1: Une partition de Voronoï barycentrique adaptative d'une image de cercle et son graphe dual.

une étude de cas de segmentation de 34 structures thoraciques et abdominaux dans une image TDM et concluons ce chapitre dans la Section A.6 avec une brève discussion décrivant les futures directions de recherche. ,

A.3 Partition d'images par tesselation de Voronoï adaptative

Plusieurs algorithmes de partition d'image ont été développés par les communautés d'analyse d'image et de vision par ordinateur pour la classification et la segmentation d'images. Nous nous intéressons particulièrement à une classe d'algorithmes qui produisent une sur-segmentation dense de régions compactes, souvent appelées super-pixels, de taille et de forme relativement uniforme, qui s'adaptent aux bords d'intensité dans l'image. Deux exemples récents sont l'algorithme TurboPixels [LSK+09] et l'algorithme de partition en super-pixels par coupure de graphes [VBM10]. Les partitions d'images produites par ces algorithmes établissent un meilleur équilibre entre les objectifs de compacité des régions et de respect des frontières des objets que d'autres méthodes de partition connues qui produisent des régions de forme et de taille très variables, comme la méthode de partition par lignes de partage des eaux [VS91] et la méthode de décalage moyen [CM02]. Une grande région de forme irrégulière est plus susceptible de s'étendre sur plus d'un objet, en particulier dans le cas de contraste insuffisant. D'autre part, une forme régulière compacte est moins susceptible de franchir les frontières d'objet, sauf si elles sont trop ondulées. Cependant, la compacité se fait au détriment du respect des frontières d'objets, il est donc important de pouvoir contrôler le compromis.

Avant de présenter notre approche de partition d'image, nous définissons une image en niveaux de gris $\mathcal{I}$ comme une fonction $\mathcal{I}\colon V \mapsto I$ d'un ensemble de voxels $V = \{v \,|\, v = (x, y, z)\}$ vers les niveaux d'intensité (de gris) $I \subset \mathbb{R}$. Afin de simplifier la notation, nous écrivons $v \in \mathcal{I}$, et dénotons le niveau d'intensité d'un voxel par I_v.

Étant donné une image en niveaux de gris $\mathcal{I}$ et n sites dans l'espace image $c_i \in \mathcal{I}$ tel que $1 \leq i \leq n$, une tesselation de Voronoï partitionne l'image $\mathcal{I}$ en n régions disjointes C_i associées avec les sites c_i:

$$C_i = \{v \in \mathcal{I} \mid d(v, c_i) < d(v, c_j);\ 1 \leq j \leq n,\ j \neq i\}. \tag{A.1}$$

Dans une partition de Voronoï barycentrique, chaque site c_i correspond au centre de gravité de la région associée, et la partition minimise l'énergie suivante:

$$E(v;c_i) = \sum_{i=1}^{n} \left(\sum_{v \in C_i} \rho(v) \|v - c_i\|^2 \right) \tag{A.2}$$

où $\rho(v)$ est une fonction de densité définie plus bas.

Nous étendons la propriété de compacité géométrique exprimé par (A.2) en introduisant une norme dans l'espace d'intensité:

$$F(v;c_i) = \sum_{i=1}^{n} \left(\sum_{v \in C_i} \rho(v) \left(\|v - c_i\|^2 + \alpha \|I_v - I_i\|^2 \right) \right) \tag{A.3}$$

où α est un scalaire positif et I_i est le niveau d'intensité de la région C_i définie comme l'intensité moyenne de ses voxels. Intuitivement, la minimization de (A.3) correspond désormais à la maximisation de la compacité des régions en termes de géométrie et d'intensité.

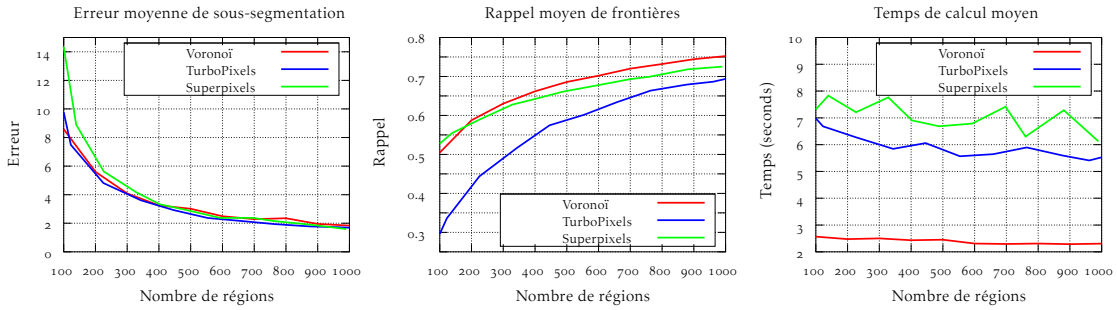
Dans (A.3), nous définissons $\rho(v)$ comme une fonction linéaire de la norme du gradient d'intensité au voxel v, $\rho(v) = m|\nabla I_v| + b$. Ceci favorise la formation des régions de petite taille à proximité des contours, alignées avec ceux-ci, ce qui permet le positionnement fin des frontières de la segmentation ultérieure. Dans tous les résultats présentés dans ce chapitre, nous choisissons $\alpha = m = b = 1$. Nous référons le lecteur à la Fig. A.1b pour une illustration de la partition Voronoï adaptative en 2D.

Afin de minimiser (A.3), nous utilisons une version de l'algorithme proposé par [DVS+09] qui produit une partition de Voronoï approchée avec une méthode efficace impliquant des comparaisons locales sur les voxels situés aux frontières des régions Voronoï. Afin de mesurer les performances de cet algorithme en termes de temps de calcul et de qualité des partitions produites, nous le comparons avec les algorithmes TurboPixels et super-pixels par coupure de graphes, qui implantent des critères similaires de compacité de régions, sur la base de données de segmentation Berkeley [MFTM01]. Cette base consiste en 300 photos génériques de taille 321×481 accompagnées des segmentations manuelles. Dans l'analyse quantitative, nous évaluons le temps d'exécution, le rappel de frontières et l'erreur de sous-segmentation sur 80 images en niveaux de gris tirées de la base en choisissant, pour chaque image, une segmentation manuelle pour nous en servir comme vérité terrain telle qu'elle ne comporte pas de petits objets de taille de quelques pixels. Tels objets produisent des points extrêmes et perturbent les mesures d'erreur moyenne de sous-segmentation.

Le rappel de frontière est défini comme la fraction des pixels de frontière de vérité terrain qui sont à une faible distance d'au moins un pixel de frontière de partition. Nous utilisons une distance d'un pixel car nous cherchons une grande précision dans le positionnement des frontières de segmentation dans les applications médicales. Étant donné une partition de Voronoï adaptative d'une image $\mathcal{C} = \{C_1, \ldots, C_n\}$ et une segmentation vérité terrain de la même image $\mathcal{T} = \{g_1, \ldots, g_k\}$, l'erreur de sous-segmentation pour l'objet g_j est définie par:

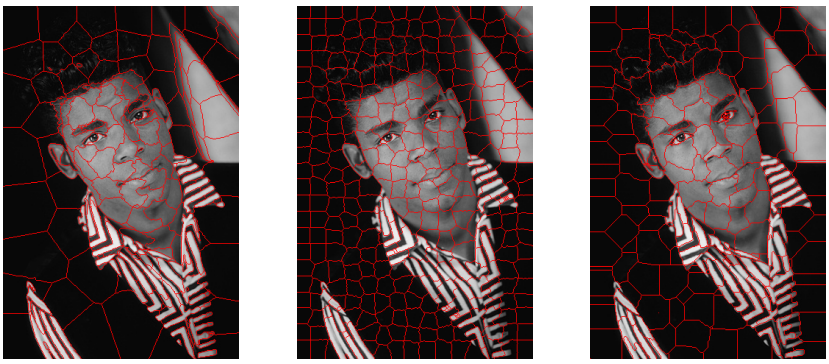
$$\underset{j}{\text{seg err}}(\mathcal{C}, \mathcal{T}) = \frac{\left(\sum_{i:C_i \cap g_j \neq \emptyset} |C_i| \right) - |g_j|}{|g_j|} \tag{A.4}$$

Pour une partition d'image, nous calculons la moyenne de (A.4) pour tous les objets de la vérité terrain correspondante et donnons l'erreur moyenne de sous-segmentation.



(a) (b) (c)

Figure A.2: Évaluation quantitative des méthodes de partition TurboPixels [LSK+09], super-pixels par coupure de graphes [VBM10] et par tesselation Voronoï adaptative sur 80 images en niveaux de gris tirées de la base de données de segmentation Berkeley.



(a) (b) (c)

Figure A.3: Comparison qualitative des méthodes de partition TurboPixels [LSK+09], super-pixels par coupure de graphes [VBM10] et par tesselation Voronoï adaptative avec 300 clusters sur une image tirée de la base de données de segmentation Berkeley. Le rappel de frontières et l'erreur de sous-segmentation sont respectivement (a) Voronoï: 0.59, 1.53(b) TurboPixels: 0.52, 2.01 (c) coupure de graphes: 0.59, 2.76.

Les graphiques dans les Fig. A.2a, A.2b et A.2c présentent les résultats des évaluations quantitatives des algorithmes de partition Voronoï, TurboPixels et super-pixels par coupure de graphe avec contraintes d'homogénéité de régions. Étant donné une résolution de partition, nous rapportons les mesures moyennes de performance calculées sur toutes les images. On peut constater dans ces graphiques que les performances de la méthode de partition Voronoï adaptative en termes de temps d'exécution sont meilleures que celles des deux autres méthodes tout en produisant une partition de qualité comparable à celle produite par l'algorithme ayant la meilleure performances parmi les deux autres algorithmes. Notons que ces expériences ont été menées sur une plate-forme informatique pilotée par un processeur cadencé de 2.84 GHz en utilisant les implantations logicielles fournies par les auteurs respectifs des trois algorithmes. Une comparaison qualitative des partitions produites par les trois algorithmes sur une image de la base Berkeley est donnée dans la Fig. A.3a, A.3b et A.3c.

Définissons formellement le graphe dual de la partition Voronoï. Ce graphe est illustré dans la Fig. A.1b. Soit ∂C_i la frontière 2D (la surface en 3D) d'une région C_i. Étant donnée une partition $\mathcal{C}$ où les régions sont indexées par $\mathcal{S}$, soit $\mathcal{G} = \langle \mathcal{S}, \mathcal{E} \rangle$ un graphe non-orienté dont les sommets sont les barycentres des régions et les arêtes $\mathcal{E} = \left\{ \{i,j\} \mid i,j \in \mathcal{S},\ |\partial C_i \cap \partial C_j| \neq 0 \right\}$ joignent les régions ayant une frontière (surface) commune de longueur (aire) non-nulle. Le voisinage d'un noeud de graphe $i \in \mathcal{S}$ est défini par $\mathcal{N}_i = \left\{ j \mid j \in \mathcal{S}, \exists \{i,j\} \in \mathcal{E} \right\}$.

A.4 Segmentation multi-objet comme étiquetage bayésien

Nous allons formuler la segmentation comme un problème d'étiquetage, qui consiste à affecter un ensemble de m étiquettes $L = \{l_1, \ldots, l_m\}$ représentant les objets ciblés à un ensemble de n variables correspondant aux régions d'une partition de Voronoï adaptative de l'image $\mathcal{C} = \{C_1, \ldots, C_n\}$ indexées par $\mathcal{S}$. On suppose, en outre, que chaque variable $i \in \mathcal{S}$ est associée avec le noeud correspondant dans le graphe dual de la partition $\mathcal{G}$ défini dans la Section A.3.

On appelle une affectation à toutes les variables une configuration, dénotée par $\ell \in \mathcal{L}$, où $\mathcal{L} = L^n$ est l'espace de toutes les configurations possibles. L'affectation à une seule variable $i \in \mathcal{S}$ est dénotée par ℓ_i. L'espace des solutions acceptables, qui correspond aux segmentations possibles, peut être identique à $\mathcal{L}$, ou, si les contraintes de l'utilisateur sont prises en compte, à un sous-ensemble de $\mathcal{L}$. Soit $X = \{X_1, \ldots, X_n\}$ un ensemble de variables aléatoires sur $\mathcal{S}$ qui prennent leurs valeurs dans L. Une configuration ℓ peut aussi être exprimée en termes probabilistes comme l'événement $X_1 = \ell_1, \ldots, X_n = \ell_n$, dénoté par $X = \ell$.

La "meilleure" segmentation est définie comme la configuration correspondant au maxima de la distribution de probabilité a posteriori des configurations $\Pr(X = \ell \mid O)$ étant donné une observation déduite de l'image O. Selon la loi de Bayes, la probabilité a posteriori peut être exprimée en termes d'une distribution de vraisemblance $\Pr(O \mid X = \ell)$ et une distribution a priori des configurations $\Pr(X = \ell)$. Si cette dernière est décrite par un champ aléatoire de Markov du premier ordre par rapport au graphe $\mathcal{G}$, et que des distributions de vraisemblance peuvent être estimées pour chaque étiquette $l \in L$ à partir des indices fournis par l'utilisateur, alors le maximum de la probabilité a posteriori peut également être calculé en minimisant la fonctionnelle d'énergie suivante [Li09]:

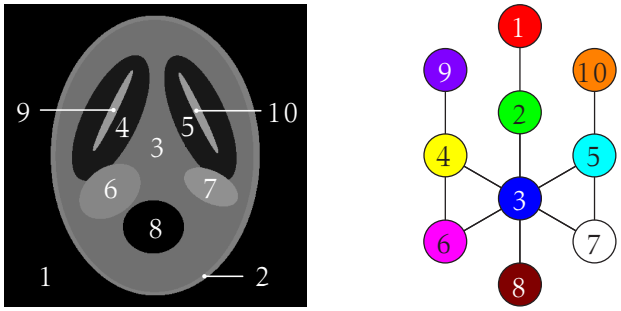
$$E(\ell) = t \sum_{i \in \mathcal{S}} D_i(\ell_i) + \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{N}_i} V_{i,j}(\ell_i, \ell_j) \tag{A.5}$$

où t est un hyper-paramètre appelé la température et $\mathcal{N}_i$ correspond au voisinage de la variable $i \in \mathcal{S}$. Les termes binaires de (A.5) encodent de l'information a priori sur les interactions entre les étiquettes affectées aux paires de variables voisines et assurent la cohérence spatiale de l'étiquetage par rapport à un modèle de référence. Les termes unaires, appelés également les termes d'attache aux données, sont essentiellement des fonctions de logarithme négatif de la vraisemblance déduite de l'image et mesurent le coût d'affectation d'une étiquette à la variable correspondante.

A.4.1 Modèle d'a priori de proximité

Dans le problème de segmentation d'images, l'a priori binaire dans l'énergie (A.5) doit favoriser un étiquetage constant par morceaux. De plus, l'étiquetage doit respecter les discontinuités en affectant des étiquettes différentes aux régions d'image de part et d'autre d'un contour.

Un modèle d'a priori classique vérifiant les propriétés ci-dessus est celui de Ising en étiquetage binaire où $|L| = 2$ [GG84]. Il est défini par $V_{i,j}(\ell_i, \ell_j) = 1 - \delta(\ell_i, \ell_j)$, où δ est la fonction



(a) (b)

Figure A.4: (a) Notre fantôme synthétique et (b) le modèle de graphe représentant les relations d'adjacence entre ses structures.

de Kronecker. La généralisation multi-valuée de ce modèle pour $|L| > 2$ est appelée «modèle de Potts», un modèle d'a priori qui a été largement utilisé en segmentation d'images et continue d'être appliqué dans les approches de segmentation récentes s'appuyant sur les champs conditionnels aléatoires [FVS09, SWRC09]. On peut constater que ce modèle n'introduit qu'un seul niveau de pénalisation correspondant à l'affectation d'étiquettes différentes à des variables voisines et favorise un étiquetage identique de ces variables. Ce mécanisme de pénalisation élémentaire provient de l'étiquetage binaire où une seule discontinuité existe entre les régions d'image étiquetées par chacune de deux étiquettes. Cette hypothèse est fausse en segmentation multi-objet.

Nous étendons la définition du modèle de Potts afin de tenir compte des multiples discontinuités en segmentation multi-objet à travers plusieurs niveaux de pénalisation dans le modèle d'a priori. Afin de motiver notre approche, nous allons illustrer un cas simple sur une image de synthèse. Considérons notre fantôme dans la Fig. A.4a dont la structure est décrite plus bas dans la Section A.5.1. Notons comment la structure "9" est imbriquée dans la structure "4" qui, à son tour, est adjacente à la structure "6". Étant donné cette configuration spatiale, il est parfaitement logique de favoriser l'attribution de la paire d'étiquettes $\{9,4\}$ aux variables voisines dans une segmentation et pénaliser l'attribution de la paire $\{9,6\}$. Donc, $\forall i,j \in \mathcal{S}$, $V_{i,j}(9,6) > V_{i,j}(9,4)$. En outre, comme la structure "8" est encore plus loin, on peut écrire $V_{i,j}(9,8) > V_{i,j}(9,6) > V_{i,j}(9,4)$. Par comparaison, le modèle d'a priori d'adjacence utilisé dans notre publication [JKD+12] et dans [CCB+05] pénalise les paires non adjacentes $\{9,6\}$ et $\{9,8\}$ de manière identique impliquant que l'attribution de ses paires d'étiquettes à des paires de variables voisines violerait la cohérence spatiale de l'étiquetage de la même manière.

Rappelons que l'ensemble d'étiquettes L représente les objets à segmenter. Soit $\mathcal{R}$ l'ensemble des relations binaires symétriques sur les paires d'étiquettes distinctes, $\mathcal{R} = \{\mathrm{r} \mid l_a \mathrm{r}\, l_b, l_a, l_b \in L, a \neq b\}$, représentant des relations telles que l'adjacence, la distance, l'aire de la surface commune, etc. Supposons que chaque relation $\mathrm{r} \in \mathcal{R}$ possède une fonction caractéristique $\hat{\mathrm{r}}$ qui donne une mesure quantitative de la relation r; $\hat{\mathrm{r}}\colon L \times L \setminus \{(l_a, l_a)\} \mapsto \mathbb{R}^+$. $\mathcal{R}$ peut également être représenté par un graphe pondéré non orienté sur l'ensemble d'étiquettes $\mathcal{A} = \langle L, W \rangle$ avec l'ensemble d'arrêtes $W = \left\{\{l_a, l_b\} \mid \exists \mathrm{r} \in \mathcal{R}, l_a \mathrm{r}\, l_b, a \neq b\right\}$ où le poids d'arrête est défini par $w(\{l_a, l_b\}) = \hat{\mathrm{r}}(l_a, l_b)$ tel que $w(\{l_a, l_b\}) = \infty$, $\nexists \mathrm{r} \in \mathcal{R}, l_a \mathrm{r}\, l_b$.

Étant donné le graphe $\mathcal{A}$, nous définissons le terme binaire dans (A.5) comme:

$$V_{i,j}\left(\ell_i, \ell_j\right) = \omega\left(l_a, l_b\right), \quad \ell_i = l_a, \ell_j = l_b \tag{A.6}$$

où $\omega\big(l_a, l_b\big)$ est la longueur du plus court chemin de l_a à l_b dans $\mathcal{A}$.

À présent, la définition que nous donnons aux relations $\mathrm{r} \in \mathcal{R}$ est celle d'adjacence des structures. Une structure est adjacente à une autre si elles partagent une frontière en 2D (une surface, en 3D). Pour deux telles structures a et b, la définition évidente de la fonction caractéristique de la relation d'adjacence est $\hat{\mathrm{r}} = 1$ quand $\exists \mathrm{r} \in \mathcal{R}, l_a \mathrm{r}\, l_b$. Cela donne un graphe $\mathcal{A}$ connecté non orienté avec des arrêtes de poids unitaire entre les paires de sommets correspondant aux structures adjacentes. Les plus court chemins entre un sommet et tout autre sommet, qui peuvent facilement être calculés avec un parcours en largeur, définissent la *proximité* de la structure associée par rapport à toute autre structure. Celle-ci est introduite dans l'énergie de la segmentation (A.5) par les termes binaires (A.6). Clairement, le graphe $\mathcal{A}$ constitue un modèle d'a priori de la configuration spatiale explicite des objets. Pour illustrer, le modèle de graphe pour les relations d'adjacence entre les structures d'image dans la Fig. A.4a est donné dans la Fig. A.4b.

Notons que le graphe $\mathcal{A}$ ne se limite pas aux arrêtes de poids unitaire. Ses arrêtes peuvent représenter toute quantité additive le long d'un chemin que nous voudrions minimiser.

Nous pondérons (A.6) avec la longueur (la superficie, en 3D) de la frontière (la surface) commune entre une paire de régions adjacentes, dénotée par $|\partial C_i \cap \partial C_j|$, afin que la somme des termes binaires dans (A.5) pour toute paire d'étiquettes $l_a, l_b \in L$ correspond à la longueur (la superficie) de la frontière (la surface) commune entre la paire d'objets correspondante, multiplié par $V(l_a, l_b)$. Cela, avec notre définition de l'attache aux données (A.7), assure que les énergies des configurations identiques sur des partitions de résolutions différentes sont égales. Nous ne donnons pas la preuve formelle ici, mais établissons empiriquement les avantages de ce mécanisme de pondération dans la Section A.5.1 par une comparaison quantitative avec la méthode de pondération par la distance inverse, fréquemment rencontrée dans les approches de segmentation d'images par les champs aléatoires de Markov, par exemple dans [BF06] et [CCB+05].

A.4.2 Attache aux données basé sur la vraisemblance

La probabilité de vraisemblance d'une variable $i \in \mathcal{S}$ étant donné l'étiquette ℓ_i est définie par $\Pr(O_i = I_i \mid X_i = \ell_i)$, où l'observation O correspond aux niveaux d'intensité des voxels d'image spatialement indépendants. Étant donné les graines fournies par l'utilisateur,[1] nous estimons la distribution de probabilité conditionnelle d'intensité $\Pr(I \mid L = l)$ pour chaque étiquette $l \in L$ à partir des histogrammes d'intensité normalisés et lissés par une gaussienne.

Afin de faciliter la compréhension des différents cas, nous définissons le terme $D_i(\ell_i)$ de l'énergie (A.5) comme suit:

$$D_i(\ell_i) = \begin{cases} 0, & \exists s_k \in C_i, k = \ell_i \quad \text{(A.7a)} \\ \infty, & \exists s_k \in C_i, k \neq \ell_i \quad \text{(A.7b)} \\ -\ln \prod_{v \in C_i} \Pr(I_v \mid \ell_i), & \nexists s_k \in C_i, \forall k \in L \quad \text{(A.7c)} \end{cases}$$

où s_k représente un voxel graine de type $k \in L$. Notons que (A.7) est une fonction continue où (A.7a) et (A.7b) correspondent au logarithme négatif de vraisemblance respectivement à la probabilité maximale et nulle. Nous représentons le coût infini de (A.7b) par une grande constante, supérieure à la valeur de (A.7c) correspondant à la plus faible probabilité de plusieurs

[1] Rappelons nous que les graines sont des affectations d'étiquette aux voxels à l'intérieur des structures ciblées.

ordres de grandeur (10^9–10^{12}). Par conséquent, pour les réglages pratiques de température $t > 0$, (A.7a) et (A.7b) reviennent à imposer des contraintes fortes pour les configurations impliquées par les graines. Par comparaison, [BF06] ne multiplient pas (A.7b) par t dans (A.5) et définissent une constante représentant son coût tel que les termes binaires n'imposent pas une étiquette sur la variable correspondante par hasard. Dans notre formulation, quand t s'approche de zero, la valeur de (A.7b) s'approche de l'ordre de grandeur des termes binaires jusqu'à ce que la somme des termes d'attache aux données dans (A.5) s'annule quand $t = 0$; un réglage qui n'est guère intéressant en pratique.

Pour un éventuel raffinement d'une segmentation initiale, nous permettons à l'utilisateur d'introduire des graines correctives afin de contraindre le choix d'étiquette pour les régions d'image qui n'ont pas été segmentées correctement. Nous ne prenons pas ces graines supplémentaires en compte dans l'estimation de vraisemblance afin de ne pas perturber les modèles d'intensité initiaux.

A.4.3 Optimisation multi-valuée par coupure de graphes

Après avoir défini le modèle d'a priori et l'attache aux données, nous tournons maintenant notre attention vers l'optimisation numérique de la fonction d'énergie de segmentation (A.5). Nous ne donnons pas les détails de l'algorithme d'optimisation ici, mais établissons la condition nécessaire et suffisante de son application.

[BVZ01] ont démontré que la minimisation multi-valuée de (A.5) est un problème NP difficile même avec le modèle simple de Potts. Néanmoins, en étiquetage binaire où $|L| = 2$, le minimum global de l'énergie (A.5) utilisant le modèle de Potts peut être calculé en temps polynomial à l'aide d'un algorithme de flot maximum [GPS89]. Dans le cas général, et pour notre problème de segmentation multi-objet où $|L| > 2$, des algorithmes d'approximation efficaces existent. Nous utilisons l'algorithme de coupure de graphes par mouvements d'expansion, dit expansion-α, proposé par [BVZ01]. Il surpasse les autres algorithmes d'optimisation multi-valuée connus en termes de vitesse d'exécution et de qualité des solutions obtenues [SZS+08].

L'algorithme expansion-α produit un étiquetage correspondant à une solution approchée de (A.5) à une distance connue du minimum global, proportionnelle au rapport $2 \times \max V_{i,j}(l_a, l_b) / \min V_{i,j}(l_a, l_b)$, $\forall i, j \in \mathcal{S}, l_a, l_b \in L, a \neq b$. Par exemple, ce rapport est de deux pour le modèle de Potts. [KZ04] ont démontré que l'algorithme expansion-α peut minimiser (A.5) si et seulement si ses termes binaires sont sous-modulaires, c'est à dire, si la condition suivante est vérifiée $\forall i, j \in \mathcal{S}, \forall l_a, l_b, l_c \in L$:

$$V_{i,j}(l_a, l_a) + V_{i,j}(l_b, l_c) \leq V_{i,j}(l_a, l_c) + V_{i,j}(l_b, l_a). \tag{A.8}$$

Notre modèle d'a priori de proximité (A.6) est sous-modulaire par définition. On peut démontrer cette propriété facilement en observant que les longueurs des plus court chemins entre tout triplet de sommets dans le graphe $\mathcal{A}$ vérifient l'inégalité triangulaire. Par comparaison, notre modèle d'a priori d'adjacence [JKD+12] est limité aux définitions où la pénalisation d'une non-adjacence est au plus deux fois supérieure à celle d'une adjacence pour que le modèle vérifie la propriété de sous-modularité. Autrement dit, il est impossible de pénaliser les structures non-adjacentes avec une grande constante sans pénaliser les structures adjacentes avec une autre constante dont la valeur est au moins la moitié de la première.

A.5 Évaluations et applications

Dans cette partie, nous présentons les résultats des évaluations quantitatives et qualitatives de notre approche de segmentation par coupure de graphes utilisant le modèle d'a priori de proximité. Dans toutes les évaluations, nous comparons notre approche avec la méthode de segmentation par coupure de graphe utilisant le modèle de Potts, et dans certaines évaluations, avec notre précédent modèle d'a priori d'adjacence [JKD+12]. Notons ici que notre implantation logicielle utilise la librairie d'optimisation multi-valuée développée par les auteurs de [BVZ01, KZ04, BK04].

Dans toutes les évaluations, nous mesurons la qualité d'une segmentation par rapport à la vérité terrain avec la métrique de similarité de Dice (DSM), qui mesure le chevauchement entre la segmentation et la référence. Soit S_l et T_l les ensembles de voxels étiquetés avec $l \in L$ dans l'image segmentée et dans l'image de vérité terrain respectivement. Dénotons la DSM pour une étiquette $l \in L$ par $\mathrm{dsm}_l(\mathcal{I}, \mathcal{T})$. Nous définissons une métrique DSM moyenne globale à partir des mesures pondérées de DSM pour toutes les étiquettes:

$$\underset{L}{\mathrm{dsm}}(\mathcal{I}, \mathcal{T}) = \frac{\sum_{l \in L} \mathrm{dsm}_l(\mathcal{I}, \mathcal{T}) |T_l|}{|\mathcal{T}|}. \tag{A.9}$$

Afin de comprendre avec quelle fiabilité l'algorithme de segmentation utilisant chacun des trois modèles d'a priori peut atteindre une valeur de DSM globale d, nous évaluons une fonction de fiabilité de DSM dans l'intervalle $[0,1]$ sur une ensemble d'images segmentées $\mathbb{I}$. Dénotons le sous-ensemble d'images qui atteignent une valeur de DSM globale supérieure à d par $\mathbb{I}_d = \{\mathcal{I} \in \mathbb{I} \mid \mathrm{dsm}_L(\mathcal{I}, \mathcal{T}) > d\}$, et soit D une variable aléatoire dans $[0,1]$. La fiabilité à une valeur de DSM d est définie par:

$$\mathrm{rel}(d) = \Pr(D > d) = \frac{|\mathbb{I}_d|}{|\mathbb{I}|}. \tag{A.10}$$

Afin d'évaluer la précision du positionnement des frontières de segmentation dans le cas d'étude médical présenté dans la Section A.5.2.2, nous utilisons une métrique d'erreur de surface qui mesure la distance Euclidienne de chaque point de la surface segmentée à la surface de référence correspondante. Soient $M^l_{\mathcal{S}}$ et $M^l_{\mathcal{T}}$ respectivement les maillages surfaciques représentant les isosurfaces des volumes étiquetés par $l \in L$ dans l'image segmentée $\mathcal{I}$ et dans l'image de référence $\mathcal{T}$. Nous obtenons ces maillages en appliquant l'algorithme «marching cubes» aux volumes étiquetés [LC87]. L'erreur au point $x \in M^l_{\mathcal{S}}$ est définie par:

$$\mathrm{err}(x) = \inf\{d(x,y) \mid y \in M^l_{\mathcal{T}}\}. \tag{A.11}$$

A.5.1 Images de synthèse

Nous avons généré un jeu de données d'images de synthèse à partir du fantôme que nous avons créé en nous inspirant du fantôme Shepp-Logan [SL74]. Nous nous servons de ce jeu d'images dans les évaluations qualitatives et quantitatives de l'algorithme de segmentation utilisant les trois modèles d'a priori: celui de Potts et nos modèles d'adjacence et de proximité. En raison du petit nombre de structures et de leur configuration spatiale triviale, le fantôme Shepp-Logan n'est pas bien adapté pour de telles évaluations. La configuration spatiale des 10 structures de notre fantôme (Fig. A.4a) imite celle de certains organes et structures thoraciques et abdominales dans une section coronale d'une image TDM. En effet, les niveaux d'intensité des

(a) (b) (c) (d) (e)

Figure A.5: (a) Image générée du fantôme dans la Fig. A.4a avec du bruit gaussien additif d'écart type $\sigma = 0.09$ accompagnée des graines utilisées dans toutes les évaluations sur le jeu d'images de synthèse (b) Segmentation de pixels d'image avec le modèle de Potts (DSM = 0.937) et (c) avec le modèle de proximité (DSM = 0.994) (d) Segmentation de partition d'image de résolution 10% avec le modèle de Potts (DSM = 0.938) et (e) avec le modèle de proximité (DSM = 0.986). Ces résultats correspondent aux segmentations générées pour le meilleur réglage de température. Voir la légende de couleur dans la Fig. A.4b

structures du fantôme ont été obtenus à partir des valeurs d'intensité moyennes des structures correspondantes dans une image TDM.

Nous avons appliqué la segmentation avec les trois modèles d'a priori aux pixels[2] ainsi qu'aux partitions Voronoï adaptatives des images de taille 512×512 générées à partir de notre fantôme en introduisant du bruit gaussien additif de moyenne nulle et d'écart-type variable $\sigma \in [0, 0.2]$, les valeurs d'intensité étant dans la fourchette $[0, 1]$. Nous avons généré 40 images pour chaque niveau de bruit. Le paramètre de température t de l'énergie (A.5) varie entre $[0, 1]$. Nous donnons la résolution de la partition d'image comme un pourcentage du nombre de pixels d'image. L'ensemble de graines utilisé par l'estimation de la vraisemblance et les contraintes fortes est donné dans la Fig. A.5a. Le nombre graines pour chaque étiquette est compris entre $[32, 160]$ et proportionnel à la taille de la structure correspondante. Sinon, les contraintes de support spatiale similaire pour toutes les étiquettes entraîneraient un biais inégale de facilité de segmentation pour les structures les plus petites, comme "9" et "10".

La Fig. A.5 donne une comparaison quantitative des meilleurs résultats de segmentation par coupure de graphes avec le modèle de Potts ou le modèle de proximité sur les pixels ainsi qu'avec une partition d'image de résolution 10%. L'image est générée à partir de notre fantôme avec un niveau de bruit de $\sigma = 0.09$, lequel peut être considéré comme un niveau relativement élevé. Le modèle de proximité est défini à partir du graphe des relations d'adjacence dans la Fig. A.4b. Les améliorations entraînées par le modèle de proximité sont évidentes dans les deux segmentations, en particulier pour les structures fines "2" et "9".

Dans le reste de cette section ainsi que dans les Fig. A.5, A.6 et A.7, toute mention du terme DSM correspond aux mesures DSM globales pour toutes les structures (A.9). Les mesures moyennes de DSM globale sont calculées sur les segmentations correspondant au meilleur réglage de température pour les 40 images générées à un niveau de bruit donné.

Dans la Fig. A.6a, nous présentons les cartes de DSM moyenne pour la segmentation utilisant le modèle de Potts ou le modèle de proximité appliquée aux pixels d'image. Nous calculons les valeurs moyennes de DSM pour les segmentations correspondant à un réglage partic-

[2] Nous avons mené nos expériences au niveau des pixels (voxels) d'image afin de comparer ses résultats avec ceux de la segmentation appliquée aux partitions d'image. Le cadre théorique décrit dans la Section A.4 s'applique en faisant correspondre les variables $i \in \mathcal{S}$ aux pixels (voxels) d'image $v \in \mathcal{I}$ et en définissant un voisinage de 4 (6) $\mathcal{N}_i$ pour chaque variable (respectivement en 3D).

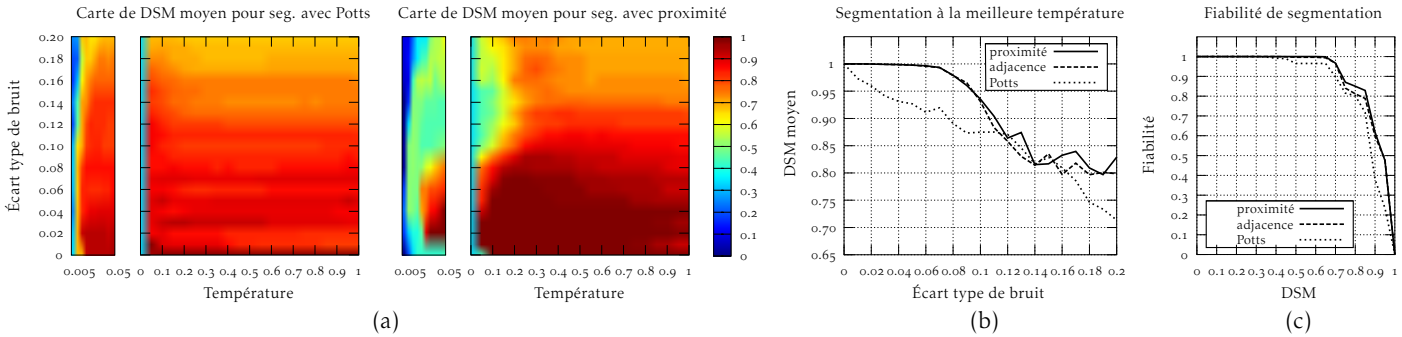


Figure A.6: Évaluation et comparaison quantitatives pour la segmentation par coupure de graphes utilisant les modèles d'a priori de Potts, d'adjacence et de proximité appliquée aux voxels des images de synthèse.

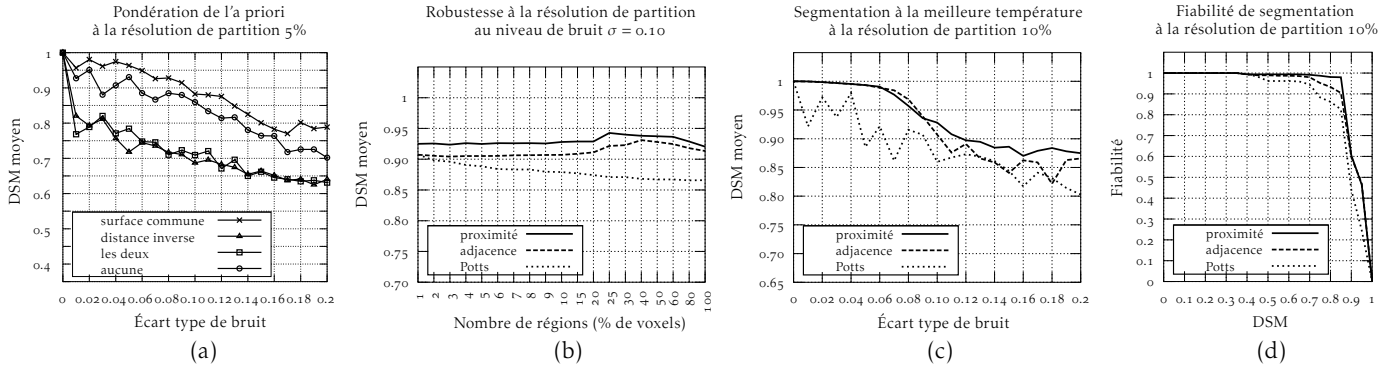


Figure A.7: Évaluation et comparaison quantitatives pour la segmentation par coupure de graphes utilisant les modèles d'a priori de Potts, d'adjacence et de proximité appliquée aux partitions des images de synthèse.

ulier de température sur toutes les images générées avec un niveau de bruit donné. Les améliorations entraînées par le modèle de proximité sont remarquables, en particulier pour les niveaux de bruit jusqu'à $\sigma = 0.1$. Notons que les faibles valeurs moyennes de DSM correspondant aux petits réglages de température sont dues à une forte régularisation par l'a priori car l'influence de l'attache aux données est faible à ces réglages-là. Rappelons aussi que le modèle de proximité entraîne une régularisation supérieure à celle du modèle de Potts car son niveau de pénalisation le plus élevé dans ce cas est 4 fois supérieur à celui de Potts. Afin d'observer la différence entre les deux modèles à une échelle plus fine, nous donnons une vue agrandie des cartes de DSM sur la fourchette de température $[0.005, 0.05]$. Nous ne donnons pas les cartes de DSM pour la segmentation appliquée aux partitions d'image car elles ont des performances similaires.

Les Fig. A.6b et A.7c comparent les performances moyennes de DSM par rapport aux niveaux de bruit croissants pour les meilleures segmentations utilisant les modèles d'a priori de Potts, d'adjacence et de proximité appliquées aux voxels et aux partitions d'image. Les Fig. A.6c et A.7d donnent les graphiques de fiabilité correspondants à ces segmentations. Nous présentons ici uniquement les résultats des segmentations appliquées aux partitions de résolution 10% car les évaluations avec d'autres résolutions ont des comportements similaires.

Nous évaluons aussi la robustesse de la segmentation par rapport à la résolution de partition pour les trois modèles d'a priori. Nous présentons les résultats correspondant à un seul niveau de bruit dans la Fig. A.7b. On peut constater que la segmentation utilisant le modèle de proximité est assez robuste à la résolution de partition. Cela implique que des segmentations de qualité similaire peuvent être obtenues en utilisant des résolutions inférieures de partition, ce qui permet d'améliorer le temps d'exécution et l'empreinte mémoire du processus de segmentation.

Globalement, dans les évaluations de segmentation par rapport aux niveaux de bruit et de résolution de partition, le modèle d'a priori de proximité est largement supérieur à celui de Potts et à celui d'adjacence dans une moindre mesure.

En dernier, nous présentons les résultats de l'évaluation quantitative des mécanismes de pondération d'a priori par rapport aux résolutions de partition variables. Les résultats correspondant à la résolution 5% sont présentés dans la Fig. A.7a. Soient C_i et C_j une paire de régions se partageant une frontière (une surface, en 3D) commune et soient c_i et c_j leur barycentres respectivement. Nous comparons les 4 mécanismes suivants: 1) pondération par l'aire de la frontière commune, où les termes binaires (A.6) sont pondérés par $|\partial C_i \cap \partial C_j|$, 2) pondération par la distance inverse entre les barycentres, où les termes binaires sont pondérés par $1/d(c_i, c_j)$, 3) les deux mécanismes de pondération précédents appliqués ensemble, et 4) aucune pondération. La Fig. A.7a montre que le premier mécanisme est nettement meilleur.

A.5.2 Images médicales simulées et réelles

A.5.2.1 Base de données d'images IRM simulées BrainWeb

Nous avons évalué notre approche de segmentation par coupure de graphes utilisant l'a priori de proximité sur la base de données d'images IRM simulées BrainWeb [CZK+98]. Dans la création de cette base, un fantôme de tête et de cerveau semi-réaliste a été utilisé dans un simulateur d'IRM des modalités T1, T2 et PD afin de générer des images selon la physique de l'acquisition d'images RM avec 6 niveaux de bruit (0%, 1%, 3%, 5%, 7% and 9%) et 3 niveaux d'inhomogénéité de champs magnétique (0%, 20% and 40%). Dans notre évaluation, nous avons utilisé 18 images issues de la modalité T1 couvrant l'ensemble de niveaux de bruit et d'inhomogénéité. Les volumes d'images sont de taille $217 \times 181 \times 217$ avec des voxels isotropes de taille 1mm. Le fantôme lui même consiste en 10 structures dont nous utilisons 9 dans l'évaluation.[3] Ces structures sont énumérées dans le Tableau A.1.

Notons ici que le fantôme de la base BrainWeb manifeste comporte incohérences anatomiques. Par exemple, des structures correspondant à la classe de tissu "muscle" et "peau" apparaissent dans le crâne à l'intérieur du volume de la classe "liquide cérébrospinal" à côté des structures de la classe "matière grise". Par conséquent, nous avons défini notre modèle de graphe des relations d'adjacence comme un compromis entre les imprécisions du fantôme et les vraies propriétés anatomiques.

Dans le Tableau A.1, nous donnons les moyennes et les écart-types des mesures de DSM globale et celles pour les structures individuelles. Ses mesures sont calculées sur les meilleures segmentations des voxels et des partitions d'image de résolution 10% pour toutes les images dans la base BrainWeb. Le modèle d'a priori de proximité est défini selon le modèle de graphe des relations d'adjacence dans la Fig. A.9a. On peut observer dans le Tableau A.1 que la segmentation par coupure de graphes peut atteindre de meilleurs niveaux moyens de performance avec l'a priori de proximité sur la base BrainWeb. En outre, elle est légèrement plus performante sur

[3] Nous avons fusionné la classe de tissu "matière gliale" dans la classe "matière grise" sans affecter la cohérence anatomique

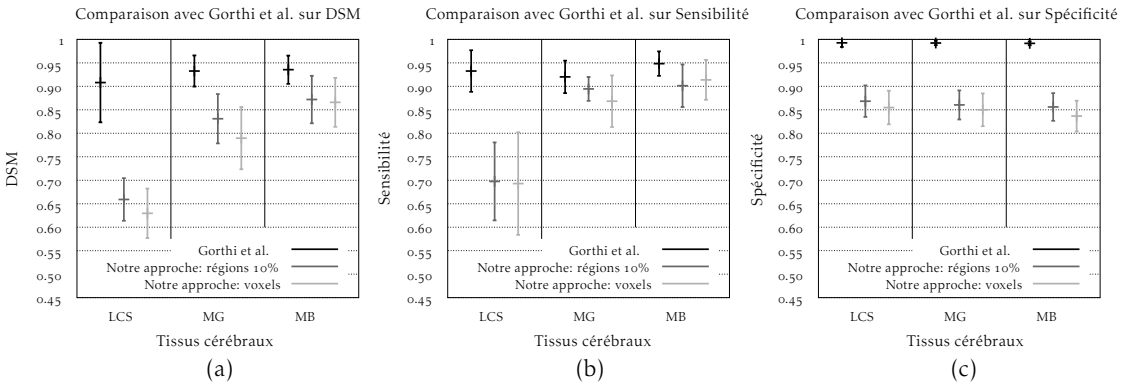


Figure A.8: Comparaison quantitative de notre approche de segmentation avec celle de [GTC11] sur la base d'images IRM simulées BrainWeb.

les partitions d'image que sur les voxels. Les chiffres correspondant aux améliorations significatives au niveau $\alpha = 0.05$ sont donnés en bleu. Les mesures moyennes de temps d'exécution et d'empreinte mémoire pour la segmentation appliquée aux voxels et aux partitions d'image de résolution 10% sont données dans le Tableau A.3.

Nous avons comparé nos résultats sur la base BrainWeb avec ceux de la classification des tissus cérébraux avec un modèle de Potts rapportés par [GTC11]. Il existe d'importantes différences entre l'approche de [GTC11] et la notre. Premièrement, elle utilise des fonctions de vraisemblance gaussiennes dont les paramètres sont appris. Deuxièmement, elle ignore toute structure autre que les matières grise, blanche et le liquide cérébrospinal, qui, en outre, ont un bon contraste dans les images [Gor12]. Ainsi, la complexité de calcul est réduite et le problème de segmentation est rendu moins difficile. Dans notre évaluation, nous avons utilisé le même sous-ensemble d'images de la base BrainWeb que [GTC11] correspondent aux 12 images en T1 générées avec les niveaux d'inhomogénéité 20% et 40% et tous les niveaux de bruit. Dans la Fig. A.8, nous comparons les moyennes et les écart-types des mesures de DSM, de sensibilité et de spécificité[4] pour les trois structures cérébrales issues de nos segmentations image-entière avec les mesures correspondantes rapportées dans [GTC11]. Sans surprise, nos résultats ne sont pas supérieurs. Les méthodes spécialisées doivent en effet produire des meilleurs résultats que les méthodes génériques comme la notre. Donc, la présente comparaison est seulement une indication du degré de similarité des résultats obtenus par une méthode générique avec ceux obtenus par une méthode spécialisée sur ce problème de segmentation. Néanmoins, notre méthode a l'avantage pratique de permettre une segmentation de cerveau sans suppression manuelle préalable des autres tissus.

A.5.2.2 Image TDM thoracique-abdominale

Les derniers résultats que nous présentons dans ce chapitre sont ceux d'un cas d'étude de segmentation sur une image TDM thoracique-abdominale 3D acquise in vivo à la position inspirée. L'image est issue de la base de données 3D-IRCADb [IRC12]. Elle est accompagnée d'une segmentation manuelle de 21 structures réalisée par des experts dont nous utilisons 16 dans nos évaluations. L'image est de taille $480 \times 370 \times 167$ avec des voxels anisotropes de taille

[4]La sensibilité et la spécificité sont définies respectivement comme le taux de vrai-positif et de vrai-négatif pour les classes d'étiquettes individuelles.

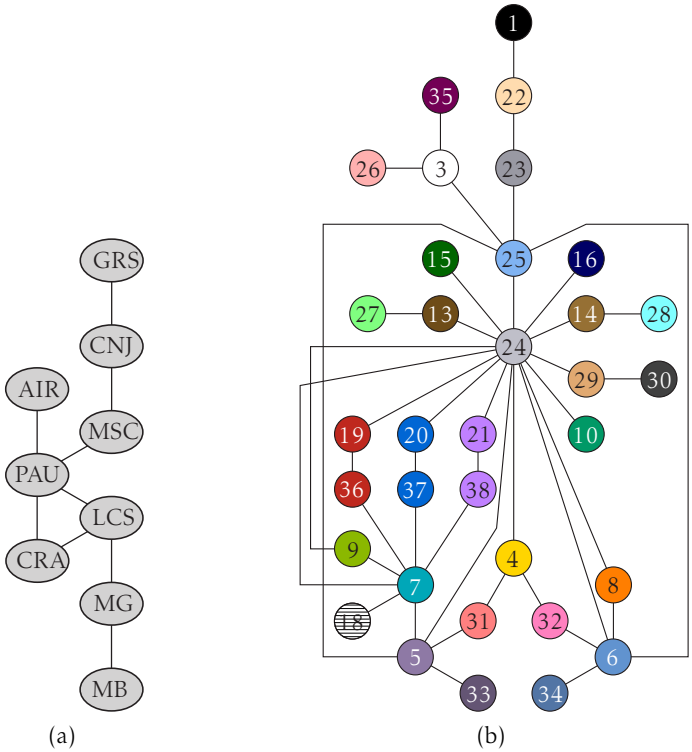
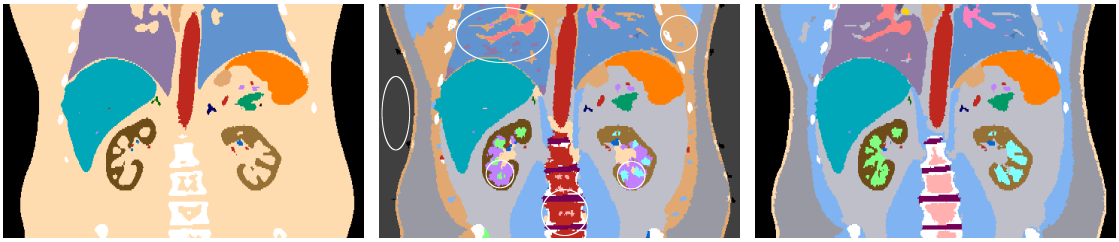


Figure A.9: Modèles de graphe anatomiques représentant les relations d'adjacence pour (a) Le fantôme de la base d'images IRM simulées BrainWeb: AIR. fond, PAU: peau, CRA; crâne, GRS: graisse, CNJ: tissu conjonctif, MSC: muscle, LCS: liquide cérébrospinal, MG: matière grise, MB: matière blanche (b) L'image TDM thoracique-abdominale 3D-IRCADb: 1. fond, 3. os, 4. coeur, 5. poumon (D), 6. poumon (G), 7. foie, 8. rate, 9. vésicule biliaire, 10. pancréas, 13. rein (D), 14. rein (G), 15. glande surrénale (D), 16. glande surrénale (G), 18. hyperplasie, 19. aorte, 20. veine cave, 21. veine porte, 22. peau, 23. graisse, 24. graisse viscérale, 25. muscle, 26. moelle d'os, 27. pyramides rénales (D), 28. pyramides rénales (G), 29. tube digestif tissu, 30. tube digestif cavité, 31. artères pulmonaires (D), 32. artères pulmonaires (G), 33. bronches (D), 34. bronches (G), 35. disques intervertébrals, 36. artères hépatiques, 37. veine cave hépatique, 38. veines portes hépatiques.

$0.961 \times 0.961 \times 1.8$ mm. Nous avons segmenté toutes les structures visibles dans cette image, y compris celles pour lesquelles aucune estimation de vérité terrain n'est disponible. Le modèle d'a priori de proximité est défini selon le graphe de 34 structures donné dans la Fig. A.9b. Les graines sont saisies par un utilisateur à l'aide d'une interface graphique en marquant, dans chaque dixième coupe axiale du volume, quelques dizaines de pixels appartenant à chaque structure ciblée. Dans le Tableau A.2, nous donnons les mesures de DSM globales et celles pour les structures individuelles correspondant aux segmentations par coupure de graphe avec le modèle de proximité ou celui de Potts appliquées à une partition d'image de résolution 5% avec une température $t = 0.5$. Le test statistique montre que les améliorations entraînées par le modèle de proximité sur la DSM moyenne sont significatives au niveau $\alpha = 0.05$.

Une comparaison qualitative est donnée dans la Fig. A.10 sur des coupes coronales de l'image de vérité terrain et les segmentations avec les deux modèles d'a priori. Notons que la vérité terrain n'est valable que pour les structures énumérées dans le Tableau A.2.



(a) (b) (c)

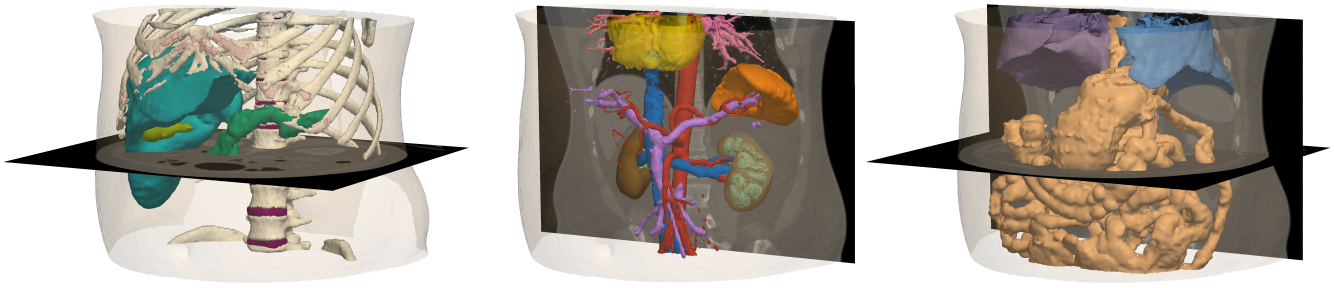
Figure A.10: Comparaison qualitative des segmentations par coupure de graphe, utilisant les modèles d'a priori de proximité et celui de Potts, appliquées à une partition de résolution 5% de l'image TDM thoracique-abdominale 3D-IRCADb. Les coupes coronales correspondent à (a) la vérité terrain (c) la segmentation utilisant le modèle de proximité et (b) celle utilisant le modèle de Potts. Voir la légende de couleur dans la Fig. A.9b. Les ellipses dans l'image (b) indiquent les mauvaises segmentations du poumon droit (en haut, à gauche), du muscle (en haut, à droite), du fond (au centre, à gauche), des pyramides rénales droite et gauche (au centre) et la moelle d'os (en bas). Notons, entre autres améliorations, que les poumons droit et gauche sont correctement segmentés comme des structures distinctes par la segmentation utilisant l'a priori de proximité.

Dans la Fig. A.11, nous donnons 3 vues 3D des maillages surfaciques simplifiés générés à partir des volumes étiquetés d'une sélection de structures. Ces maillages ont été générés avec la méthode de simplification décrite dans [VC04]. Les mesures de temps d'exécution et d'empreinte mémoire pour la segmentation appliquée aux voxels et aux partitions d'image de résolution 5% sont données dans le Tableau A.3.

Nous concluons cette section avec les mesures d'erreur de surface pour les segmentations utilisant le modèle de proximité ou celui de Potts appliquées aux partitions de résolution 5% de l'image TDM. Dans la Fig. A.12, nous traçons les histogrammes d'erreurs cumulées (A.11) pour 6 structures représentatives des deux segmentations. Les graphiques dans la Fig. A.12 indiquent la fraction de surface segmentée se trouvant à une distance donnée de la surface de vérité terrain. On peut constater que, pour les niveaux d'erreur relativement faibles 0–1 cm, la segmentation avec le modèle d'a priori de proximité récupère une fraction de surface beaucoup plus grande que celle utilisant le modèle de Potts, tout en ayant une erreur maximale beaucoup plus faible. Cependant, elle produit des points extrêmes visibles, notamment pour le coeur et le poumon droit. Ces points extrêmes constituent en général un faible pourcentage de la structure correspondante et peuvent être corrigés avec des graines correctives comme mentionné dans la Section A.4.2.

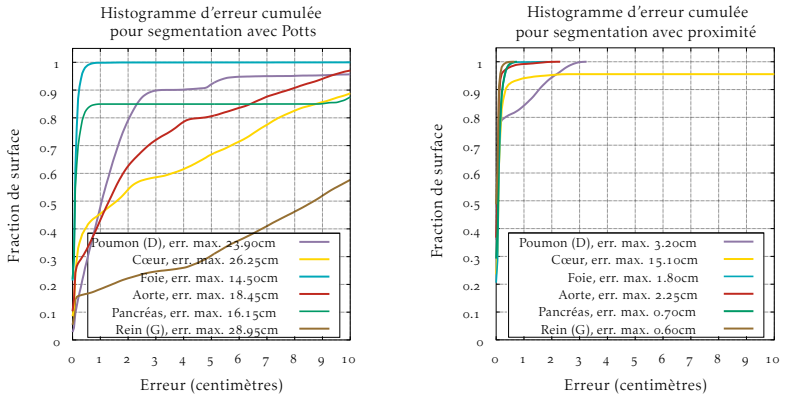
A.6 Discussion et travaux futurs

Contrairement au modèle de Potts, le modèle d'a priori de proximité que nous proposons est un modèle constant par morceaux, imposant plusieurs niveaux de pénalisation selon la configuration spatiale explicite des objets dans les problèmes de segmentation multi-objet. Par rapport aux approches de segmentation s'appuyant sur l'a priori de forme, les modèles d'a priori structurels représentant la configuration spatiale sont robustes aux déformations de forme, car les positions relatives des objets restent largement stables. En outre, ces modèles sont relativement faciles à définir à un niveau de détail arbitraire à travers de simples spécifications des relations binaires entre les objets, et peuvent être générés à partir des modèles anatomiques



(a) (b) (c)

Figure A.11: Maillages surfaciques générés à partir des volumes étiquetés d'une sélection de structures dans l'image TDM 3D-IRCADb segmentée par la méthode de coupure de graphes avec l'a priori de proximité.



(a) (b)

Figure A.12: Mesures d'erreur de surface pour la segmentation par coupure de graphe avec (a) le modèle d'a priori de Potts et (b) le modèle de proximité. Les segmentations sont appliqué à une partition de résolution de 5% de l'image TDM 3D-IRCADb.

existants pour les applications médicales, comme nous l'avons vu dans la Section A.5.

Notre formulation d'a priori à partir de la longueur du plus court chemin est un modèle sous-modulaire par définition permettant d'optimiser l'énergie de segmentation (A.5) de manière efficace et optimale avec l'algorithme expansion-α. Les modèles d'a priori binaire appris à partir des bases d'apprentissage ne vérifient souvent pas la propriété de sous-modularité. Par conséquent, certains appliquent une troncature des termes non sous-modulaires afin d'optimiser l'énergie avec les algorithmes d'optimisation par coupure de graphes [WS06], et d'autres font appel à des algorithmes d'optimisation d'énergies arbitraires [LLB+11].

La validation qualitative et l'évaluation quantitative sur des images synthétiques, simulées et réelles dans la Section A.5 ont confirmé les avantages de notre modèle d'a priori de proximité par rapport au modèle standard de Potts dans la segmentation multi-objet par coupure de graphe. En particulier, il permet la segmentation correcte de structures distinctes ayant des niveaux d'intensité similaires, le positionnement précis des frontières des structures segmentées ainsi qu'une bonne robustesse par rapport à la résolution de partitions d'image.

Cependant, comme d'autres approches de segmentation s'appuyant sur les algorithmes de coupure de graphes, notre méthode de segmentation et sujette à un biais de rétrécissement. Ceci se manifeste particulièrement dans la segmentation des structures fines allongées, comme les vaisseaux, où la somme des termes binaires le long de la frontière d'un vaisseau est supérieure au coût d'un raccourci au travers cette structure fine, en particulier quand les termes d'attache aux donnée n'imposent pas de préférence suffisamment forte pour l'étiquette du vaisseau. On peut observer, par exemple, la segmentation fragmentaire des artères abdominales et hépatiques et les veines portes dans la Fig. A.11b. Un nombre de solutions ont été proposées afin de résoudre ce problème. Par exemple, [VKR08] améliorent une segmentation initiale par coupure de graphes mono-objet avec des contraintes de connectivité introduites par l'utilisateur à l'aide de marquages sur les régions d'image qui doivent être connectées à l'objet principal. Dans le problème de la segmentation des artères et des veines portes hépatiques dans les images TDM, [PWL11] utilisent un modèle d'attache aux données Hessien s'appuyant sur une vraisemblance mesurant la similarité de la structure correspondante à un vaisseau. Cette méthode pourrait se prêter à d'autres structures similaires, comme les bronches ou les artères pulmonaires.

Nous étudions actuellement des extensions du modèle d'a priori graphique afin de permettre des relations spatiales autre que l'adjacence, notamment les relations d'orientation spatiale. Comme la formulation s'appuyant sur la longueur du plus court chemin sur un graphe orienté produit une définition d'a priori binaire asymétrique, de telles extensions nécessiteront une révision des propriétés fondamentales de la théorie de l'étiquetage bayésien sous-jacente.

Une autre direction intéressante des travaux futurs est l'introduction de l'incertitude dans le modèle d'a priori graphique. Comme mentionné dans la Section A.4.1, les modèles de graphe dont nous nous servons dans la définition du modèle d'a priori de proximité ne se limitent pas aux arrêtes de poids unitaire. Des poids continus peuvent être utilisés, par exemple, pour représenter le degré d'adjacence, lequel peut être appris d'une base d'apprentissage ou de plusieurs variantes du modèle pour ainsi représenter ses variabilités.

	voxels		**partitions** 10%	
Structures	Potts	proximité	Potts	proximité
Fond	0.98 ± 0.01	0.98 ± 0.01	0.98 ± 0.01	0.99 ± 0.00
Liquide cérébrospinal	0.25 ± 0.16	0.65 ± 0.06	0.29 ± 0.11	0.67 ± 0.05
Matière grise	0.79 ± 0.09	0.82 ± 0.08	0.80 ± 0.08	0.85 ± 0.06
Matière blanche	0.86 ± 0.08	0.88 ± 0.07	0.87 ± 0.07	0.88 ± 0.06
Graisse	0.89 ± 0.05	0.84 ± 0.08	0.87 ± 0.04	0.85 ± 0.06
Muscle	0.73 ± 0.06	0.74 ± 0.12	0.73 ± 0.07	0.79 ± 0.09
Peau	0.70 ± 0.07	0.75 ± 0.09	0.69 ± 0.07	0.75 ± 0.07
Crâne	0.76 ± 0.07	0.79 ± 0.05	0.76 ± 0.05	0.80 ± 0.04
Tissu conjonctif	0.71 ± 0.14	0.69 ± 0.21	0.73 ± 0.12	0.75 ± 0.15
Global	0.81 ± 0.03	0.85 ± 0.04	0.82 ± 0.03	0.87 ± 0.04

Tableau A.1: Les moyens et les écart-types des mesures DSM globales et celles pour les structures individuelles calculées sur des segmentations utilisant les modèles d'a priori de proximité et de Potts appliquées aux voxels et aux partitions de résolution 10% des images IRM T1 simulées issues de la base BrainWeb. Les chiffres en bleu correspondent aux améliorations significatives au niveau $\alpha = 0.05$.

Structures	Potts	proximité	Structures	Potts	proximité
Os	0.735	0.857	Rein (D)	0.907	0.967
Coeur	0.918	0.988	Rein (G)	0.880	0.968
Poumon (D)	0.096	0.976	Glande surrénale (D)	0.765	0.743
Poumon (G)	0.646	0.980	Glande surrénale (G)	0.706	0.707
Foie	0.990	0.989	Hyperplasie	0.645	0.646
Rate	0.965	0.982	Aorte	0.512	0.950
Vésicule	0.902	0.923	Veine cave	0.894	0.923
Pancréas	0.895	0.959	Veine porte	0.537	0.800
Global	0.721	0.962	*moyen*	0.750	0.897

Tableau A.2: Mesures de DSM globale et pour les structures individuelles calculées sur deux segmentations utilisant les modèles d'a priori de proximité et de Potts appliquées à une partition de résolution 5% de l'image TDM 3D-IRCADb. Les chiffres en bleu correspondent aux améliorations significatives.

Images	Segmentation	Part.	Constr.	Optim.	Mém.
IRM BrainWeb	voxels	—	0.25	12.46	3053.13
	partitions 10%	3.19	0.18	2.83	376.55
TDM 3D-IRCADb	voxels	—	2.10	460.23	23864.38
	partitions 5%	15.48	1.44	14.57	2624.69

Tableau A.3: Évaluation du temps de calcul moyen pour la segmentation utilisant le modèle d'a priori de proximité appliquée aux voxels et aux partitions de deux ensembles d'images; les images IRM BrainWeb et l'image TDM 3D-IRCADb, pour lesquels les mesures ont été effectuées respectivement sur deux plateformes informatiques dotées des processeurs cadencés de 2.84 GHz et 2.66 GHz et de 6 Go et 47 Go de mémoire vive. Les 4 dernières colonnes indiquent respectivement le temps pris par la partition d'images, la construction de graphe et l'optimisation par coupure de graphes en min. et l'empreinte mémoire en Mo.

Bibliography

[AO89] Ravindra K. Ahuja and James B. Orlin. A fast and simple algorithm for the maximum flow problem. *Operations Research*, 37(5):748–759, 1989.

[AEC06] Berengere Aubert-Broche, Alan C. Evans, and Louis Collins. A new improved version of the realistic digital brain phantom. *NeuroImage*, 32(1):138–145, August 2006.

[BS05] David A. Bader and Vipin Sachdeva. A cache-aware parallel implementation of the push-relabel network flow algorithm and experimental evaluation of the gap relabeling heuristic. In *18th ISCA International Conference on Parallel and Distributed Computing Systems*, pages 41–48, 2005.

[BC08] Hans-Jurgen Bandelt and Victor Chepoi. Metric graph theory and geometry: a survey. *Contemporary Mathematics*, 453:49–86, 2008.

[Bes74] Julian Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):192–236, January 1974.

[Bes86] Julian Besag. On the statistical analysis of dirty pictures (with discussion). *Journal of the Royal Statistical Society. Series B (Methodological)*, 48(3):259–302, 1986.

[BCK+11] Arthur Blume, Won Chun, David Kogan, Vangelis Kokkevis, Nico Weber, Rachel Weinstein Petterson, and Roni Zeiger. Google body: 3D human anatomy in the browser. In *ACM SIGGRAPH 2011 Talks*, page 19:1–19:1, 2011. Currently maintained by Zygote Media Group, Inc. as Zygote Body, available at: `http://zygotebody.com/`.

[BPB13] John A. Bogovic, Jerry L. Prince, and Pierre-Louis Bazin. A multiple object geometric deformable model for image segmentation. *Computer Vision and Image Understanding*, 117(2):145–157, 2013.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[BF06] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient N-D image segmentation. *International Journal of Computer Vision*, 70(2):109–131, 2006.

[BJ01] Yuri Boykov and Marie-Pierre Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In *IEEE International Conference on Computer Vision*, volume 1, pages 105–112, Los Alamitos, CA, USA, 2001.

[BK03] Yuri Boykov and Vladimir Kolmogorov. Computing geodesics and minimal surfaces via graph cuts. In *Proceedings of the 9th IEEE International Conference on Computer Vision*, volume 1, pages 26–33, Nice, France, 2003.

[BK04] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.

[BV06] Yuri Boykov and Olga Veksler. Graph cuts in vision and graphics: Theories and applications. In *Handbook of Mathematical Models in Computer Vision*. Springer, 2006.

[BVZ01] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[BKR96] Rainer E. Burkard, Bettina Klinz, and Rüdiger Rudolf. Perspectives of Monge properties in optimization. *Discrete Applied Mathematics*, 70(2):95–161, 1996.

[BSJB11] Jerrold T. Bushberg, J. Anthony Seibert, Edwin M. Leidholdt Jr, and John M. Boone. *The Essential Physics of Medical Imaging*. Lippincott Williams & Wilkins, 3rd edition, December 2011.

[Cat92] Olivier Catoni. Rough large deviation estimates for simulated annealing: applications to exponential schedules. *The Annals of Probability*, 20(3):1109–1146, 1992.

[CG97] Boris V. Cherkassky and Andrew V. Goldberg. On implementing the push-relabel method for the maximum flow problem. *Algorithmica*, 19(4):390–410, 1997.

[CA10] Cevahir Cigla and A. Aydm Alatan. Efficient graph-based image segmentation via speeded-up turbo pixels. In *17th IEEE International Conference on Image Processing*, pages 3013–3016, Hong Kong, Hong kong, September 2010.

[CZK+98] D. Louis Collins, Alex P. Zijdenbos, Vasken Kollokian, John G. Sled, N. J. Kabani, Colin J. Holmes, and Alen C. Evans. Design and construction of a realistic digital brain phantom. *IEEE Transactions on Medical Imaging*, 17(3):463–468, June 1998.

[CM02] Dorin Comaniciu and Peter Meer. Mean shift: a robust approach toward feature analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, May 2002.

[CLRS09] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to algorithms*. The MIT Press, 3rd edition, July 2009.

[CRD07] Daniel Cremers, Mikael Rousson, and Rachid Deriche. A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape. *International Journal of Computer Vision*, 72(2):195–215, 2007.

[CCB+05] Meritxell Bach Cuadra, Leila Cammoun, Torsten Butz, Olivier Cuisenaire, and Jean-Philippe Thiran. Comparison and validation of tissue modelization and statistical classification methods in T1-weighted MR brain images. *IEEE Transactions on Medical Imaging*, 24(12):1548–1565, December 2005.

[DJP+92] Elias Dahlhaus, David S. Johnson, Christos H. Papadimitriou, Paul D. Seymour, and Mihalis Yannakakis. The complexity of multiway cuts. In *Proceedings of the 24th annual ACM symposium on Theory of computing*, pages 241–251, 1992.

[DVS+09] Julien Dardenne, Sébastien Valette, Nicolas Siauve, Noël Burais, and Rémy Prost. Variational tetrahedral mesh generation from discrete volume data. *The Visual Computer*, 25(5):401–410, May 2009.

[DB08] Andrew Delong and Yuri Boykov. A scalable graph-cut algorithm for N-D grids. In *26th IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, USA, June 2008.

[DB09] Andrew Delong and Yuri Boykov. Globally optimal segmentation of multi-region objects. In *12th International Conference on Computer Vision*, pages 285–292, Kyoto, Japan, September 2009.

[DLR77] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[Dic45] Lee R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, July 1945.

[DFG99] Qiang Du, Vance Faber, and Max Gunzburger. Centroidal voronoi tessellations: applictions and algorithms. *SIAM Review*, 41(4):637–676, 1999.

[EK72] Jack Edmonds and Richard M. Karp. Theoretical improvements in the algorithmic efficiency for network flow problems. *Journal of the ACM*, 19(2):248–264, 1972.

[FU00] Alexandre X. Falcão and Jayaram K. Udupa. A 3d generalization of user-steered live-wire segmentation. *Medical Image Analysis*, 4(4):389–402, 2000.

[FUS+98] Alexandre X. Falcão, Jayaram K. Udupa, Supun Samarasekera, Shoba Sharma, Bruce Elliot Hirsch, and Roberto de A Lotufo. User-steered image segmentation paradigms: Live wire and live lane. *Graphical models and image processing*, 60(4):233–260, 1998.

[Faw06] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, June 2006.

[FH04] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):2004, 2004.

[FF62] Lester R. Ford and Delbert R. Fulkerson. *Flows in Networks*. Princeton University Press, 1962.

[FP11] David A. Forsyth and Jean Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 2nd edition, November 2011.

[FAB12] Geoffroy Fouquier, Jamal Atif, and Isabelle Bloch. Sequential model-based segmentation and recognition of image structures driven by visual features and spatial relations. *Computer Vision and Image Understanding*, 116(1):146–165, 2012.

[FZ05] Daniel Freedman and Tao Zhang. Interactive graph cut based segmentation with shape priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 755–762, Los Alamitos, CA, USA, 2005.

[FVS09] Brian Fulkerson, Andrea Vedaldi, and Stefano Soatto. Class segmentation and object localization with superpixel neighborhoods. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 670 –677, October 2009.

[GG84] Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, 1984.

[Gol87] Andrew V. Goldberg. *Efficient Graph Algorithms for Sequential and Parallel Computers*. PhD thesis, Department of Electrical Engineering and Computer Science, MIT, 1987.

[GR98] Andrew V. Goldberg and Satish Rao. Beyond the flow decomposition barrier. *Journal of the ACM*, 45(4):783–797, 1998.

[GT88] Andrew V. Goldberg and Robert E. Tarjan. A new approach to the maximum flow problem. *Journal of the ACM*, 35(4):921–940, 1988.

[Gor12] Subrahmanyam Gorthi. private communication, July 2012.

[GTC11] Subrahmanyam Gorthi, Jean-Philippe Thiran, and Meritxell Bach Cuadra. Comparison of energy minimization methods for 3-D brain tissue classification. In *18th IEEE International Conference on Image Processing*, pages 57–60, September 2011.

[GRC+08] Stephen Gould, Jim Rodgers, David Cohen, Gal Elidan, and Daphne Koller. Multi-class segmentation with relative location prior. *International Journal of Computer Vision*, 80(3):300–316, 2008.

[GFL04] Leo Grady and Gareth Funka-Lea. Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. In *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis*, pages 230–245. Springer, 2004.

[Gra58] Henry Gray. *Anatomy: Descriptive and Surgical*. John W. Parker and Son, London, 1858. Available at: `http://openlibrary.org/books/OL24780759M/Anatomy_descriptive_and_surgical`.

[GPS89] D. M. Greig, B. T. Porteous, and A. H. Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society. Series B (Methodological)*, 51(2):271–279, 1989.

[HXP03] Xiao Han, Chenyang Xu, and Jerry L. Prince. A topology preserving level set method for geometric deformable models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):755–768, 2003.

[IRC12] IRCAD France. 3D-IRCADb, 2012. `http://www.ircad.fr/softwares/3Dircadb/3Dircadb.php`.

[Ish03] Hiroshi Ishikawa. Exact optimization for markov random fields with convex priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1333–1336, 2003.

[IG98a] Hiroshi Ishikawa and Davi Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *Proceedings of the Fifth European Conference on Computer Vision*, pages 232–248, Freiburg, Germany, 1998.

[IG98b] Hiroshi Ishikawa and Davi Geiger. Segmentation by grouping junctions. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 125–131, Santa Barbara, CA, USA, 1998.

[JKD+12] Hector Jacinto, Razmig Kéchichan, Michel Desvignes, Rémy Prost, and Sébastien Valette. A web interface for 3d visualization and interactive segmentation of medical images. In *17th International Conference on 3D Web Technology*, pages 51–58, Los-Angeles, USA, August 2012.

[JJAM11] Julien Jomier, Sebastien Jourdain, Utkarsh Ayachit, and Charles Marion. Remote visualization of large datasets with midas and ParaViewWeb. In *Proceedings of the 16th International Conference on 3D Web Technology*, pages 147–150, 2011.

[KVDP11] Razmig Kéchichian, Sébastien Valette, Michel Desvignes, and Rémy Prost. Efficient multi-object segmentation of 3D medical images using clustering and graph cuts. In *18th IEEE International Conference on Image Processing*, pages 2149–2152, Brussels, Belgium, September 2011.

[KVDP13] Razmig Kéchichian, Sébastien Valette, Michel Desvignes, and Rémy Prost. Shortest-path constraints for 3d multi-object semi-automatic segmentation via clustering and graph cut. *IEEE Transactions on Image Processing*, 2013. Accepted.

[KGV83] Scott Kirkpatrick, C. Daniel Gelatt, and Mario P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[KRBT08] Pushmeet Kohli, Jonathan Rihan, Matthieu Bray, and Philip H. S. Torr. Simultaneous segmentation and pose estimation of humans using dynamic graph cuts. *International Journal of Computer Vision*, 79(3):285–298, 2008.

[KT05] Pushmeet Kohli and Philip H. S. Torr. Efficiently solving dynamic markov random fields using graph cuts. In *10th IEEE International Conference on Computer Vision*, volume 2, pages 922–929, 2005.

[Kol06] Vladimir Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1568–1583, 2006.

[KB05] Vladimir Kolmogorov and Yuri Boykov. What metrics can be approximated by geo-cuts, or global optimization of length/area and flux. In *Proceedings of thr 10th IEEE International Conference on Computer Vision*, volume 1, pages 564–571, Beijing, China, October 2005.

[KZ04] Vladimir Kolmogorov and Ramin Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.

[Lau01] Steffen L. Lauritzen. Causal inference from graphical models. *Monographs on Statistics and Applied Probability*, 87:63–108, 2001.

[Law01] Eugene Lawler. *Combinatorial Optimization: Networks and Matroids*. Dover Publications, 2001.

[LSK+09] Alex Levinshtein, Adrian Stere, Kiriakos N. Kutulakos, David J. Fleet, Sven J. Dickinson, and Kaleem Siddiqi. TurboPixels: fast superpixels using geometric flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2290–2297, 2009.

[Li09] Stan Z. Li. *Markov random field modeling in image analysis*. Springer, 2009.

[LSTS04] Yin Li, Jian Sun, Chi-Keung Tang, and Heung-Yeung Shum. Lazy snapping. *ACM Transactions on Graphics*, 23(3):303–308, August 2004.

[LPPS12] Marius G. Linguraru, John A. Pura, Vivek Pamulapati, and Ronald M. Summers. Statistical 4D graphs for multi-organ abdominal segmentation from multiphase CT. *Medical Image Analysis*, 16(4):904–914, 2012.

[LVS10] Xiaoqing Liu, Olga Veksler, and Jagath Samarabandu. Order-preserving moves for graph-cut-based optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1182–1196, 2010.

[Llo82] Stuart Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, IT-28(2 pt 1):129–137, 1982.

[LSGX05] Herve Lombaert, Yiyong Sun, Leo Grady, and Chenyang Xu. A multilevel banded graph cuts method for fast image segmentation. In *10th IEEE International Conference on Computer Vision*, volume 1, pages 259 – 265 Vol. 1, October 2005.

[LC87] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *ACM SIGGRAPH Computer Graphics*, 21(4):163–169, July 1987.

[Low04] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[LLB+11] Aurelien Lucchi, Yunpeng Li, Xavier Boix, Kevin Smith, and Pascal Fua. Are spatial and global constraints really necessary for segmentation? In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9–16, November 2011.

[MRT07] James Malcolm, Yogesh Rathi, and Allen Tannenbaum. Graph cut segmentation with nonlinear shape priors. In *Proceedings of the IEEE International Conference on Image Processing*, volume 4, pages 365–368, October 2007.

[MFTM01] David Martin, Charles Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the 8th International Conference of Computer Vision*, volume 2, pages 416–423, July 2001.

[MYW05] Talya Meltzer, Chen Yanover, and Yair Weiss. Globally optimal solutions for energy minimization in stereo vision using reweighted belief propagation. In *Proceedings of the 10th International Conference of Computer Vision*, pages 428–435, 2005.

[MB98] Eric N. Mortensen and William A. Barrett. Interactive segmentation with intelligent scissors. *Graphical models and image processing*, 60(5):349–384, 1998.

[Mur03] Kazuo Murota. *Discrete Convex Analysis*. SIAM Society for Industrial and Applied Mathematics, 2003.

[OYH+12] Toshiyuki Okada, Yasuhide Yoshida, Masatoshi Hori, Ronald M. Summers, Yen-Wei Chen, Noriyuki Tomiyama, and Yoshinobu Sato. Abdominal multi-organ segmentation of CT images based on hierarchical spatial modeling of organ interrelations. In *Proceedings of the 3rd International Conference on Abdominal Imaging: Computational and Clinical Applications*, page 173–180, 2012.

[Orl09] James B. Orlin. A faster strongly polynomial time algorithm for submodular function minimization. *Mathematical Programming*, 118(2):237–251, January 2009.

[OF03] Stanley Osher and Ronald Fedkiw. *Level set methods and dynamic implicit surfaces*, volume 153. Springer Verlag, 2003.

[PWL11] Vivek Pamulapati, Bradford J. Wood, and Marius G. Linguraru. Intrahepatic vessel segmentation and classification in multi-phase CT using optimized graph cuts. In *IEEE International Symposium on Biomedical Imaging*, pages 1982 –1985, April 2011.

[Par62] Emanuel Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, September 1962.

[Pea88] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[Pia11] Oleg S. Pianykh. *Digital imaging and communications in medicine (DICOM): A Practical Introduction and Survival Guide*. Springer, 2011.

[PHP+01] Andreas Pommert, Karl Heinz Höhne, Bernhard Pflesser, Ernst Richter, Martin Riemer, Thomas Schiemann, Rainer Schubert, Udo Schumacher, and Ulf Tiede. Creating a high-resolution spatial/symbolic model of the inner organs based on the visible human. *Medical Image Analysis*, 5(3):221–228, September 2001.

[RM03] Xiaofeng Ren and Jitendra Malik. Learning a classification model for segmentation. In *Porceedings of the 9th IEEE International Conference on Computer Vision*, volume 1, pages 10–17, October 2003.

[RRBCK11] Alexis Roche, Delphine Ribes, Meritxell Bach Cuadra, and Gunnar Krüger. On the convergence of EM-like algorithms for image segmentation using Markov random fields. *Medical Image Analysis*, 15(6):830–839, 2011.

[RKB04] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. "GrabCut": interactive foreground extraction using iterated graph cuts. In *ACM SIGGRAPH*, pages 309–314, New York, NY, USA, 2004.

[RKKB05] Carsten Rother, Sanjiv Kumar, Vladimir Kolmogorov, and Andrew Blake. Digital tapestry. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 589–596, San Diego, CA, USA, June 2005.

[Seg01] William Paul Segars. *Development and Application of the New Dynamic NURBS-based Cardiac-Torso (NCAT) Phantom*. PhD thesis, University of North Carolina, May 2001.

[SBZ+09] Sascha Seifert, Adrian Barbu, S. Kevin Zhou, David Liu, Johannes Feulner, Martin Huber, Michael Suehling, Alexander Cavallaro, and Dorin Comaniciu. Hierarchical parsing and semantic navigation of full body CT data. In *SPIE Medical Imaging*, Lake Buena Vista, FL, USA, February 2009.

[SSLS+01] David W. Shattuck, Stephanie R. Sandor-Leahy, Kirt A. Schaper, David. A Rottenberg, and Richard M. Leahy. Magnetic resonance image tissue classification using a partial volume model. *NeuroImage*, 13(5):856–876, May 2001.

[SL74] L. A. Shepp and B. F. Logan. The fourier reconstruction of a head section. *IEEE Transactions on Nuclear Science*, 21(3):21–43, June 1974.

[SM00] Jinabo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.

[SWRC09] Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. TextonBoost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *International Journal of Computer Vision*, 81(1):2–23, 2009.

[STAG06] Zhunag Song, Nicholas. Tustison, Brian Avants, and James C. Gee. Adaptive graph cuts with tissue priors for brain MRI segmentation. In *3rd IEEE International Symposium on Biomedical Imaging*, pages 762–765, April 2006.

[SASW96] Victor Spitzer, Michael J Ackerman, Ann L Scherzinger, and David Whitlock. The visible human male: a technical report. *Journal of the American Medical Informatics Association*, 3(2):118–130, 1996.

[Sta08] Susan Standring, editor. *Gray's Anatomy: The Anatomical Basis of Clinical Practice*. Churchill Livingstone, 40th edition, November 2008.

[SZS+08] Richard Szeliski, Ramin Zabih, Daniel Scharstein, Olga Veksler, Vladimir Kolmogorov, Aseem Agarwala, Marshall Tappen, and Carsten Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1068–1080, 2008.

[TT11] Wenbing Tao and Xue-Cheng Tai. Multiple piecewise constant with geodesic active contours (mpc-gac) framework for interactive image segmentation using graph cut optimization. *Image and Vision Computing*, 29(8):499–508, 2011.

[U.S12] U.S. National Library of Medicine. The visible human project, 2012. `http://www.nlm.nih.gov/research/visible/`.

[VC04] Sebastien Valette and Jean-Marc Chassery. Approximated centroidal voronoi diagrams for uniform polygonal mesh coarsening. *Computer Graphics Forum*, 23(3):381–389, 2004.

[VBM10] Olga Veksler, Yuri Boykov, and Paria Mehrani. Superpixels and supervoxels in an energy optimization framework. In *11th European Conference on Computer Vision*, pages 211–224, Heraklion, Crete, Greece, September 2010.

[VIM04] Judith M. Venuti, Celina Imielinska, and Pat Molholt. New views of male pelvic anatomy: role of computer-generated 3D images. *Clinical Anatomy*, 17(3):261–271, April 2004.

[Ves43] Andreas Vesalius. *De humani corporis fabrica libri septem*. Joannis Oporini, Basel, 1543. An English translation in progress available at: `http://vesalius.northwestern.edu/`.

[VK05] Vladimir Vezhnevets and Vadim Konouchine. "Grow-Cut" - interactive multi-label N-D image segmentation. In *Graphicon*, pages 150–156, 2005.

[VKR08] Sara Vicente, Vladimir Kolmogorov, and Carsten Rother. Graph cut based image segmentation with connectivity priors. In *26th IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, USA, June 2008.

[VS91] Luc Vincent and Pierre Soille. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598, June 1991.

[VN08] Vibhav Vineet and P. J. Narayanan. CUDA cuts: Fast graph cuts on the GPU. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1 –8, June 2008.

[VM08] Nhat Vu and B.S. Manjunath. Shape prior segmentation of multiple objects with graph cuts. In *26th IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, AK, USA, 2008.

[WF01] Yair Weiss and William T. Freeman. On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs. *IEEE Transactions on Information Theory*, 47(2):723–735, 2001.

[Wel47] B. L. Welch. The generalisation of student's problems when several different population variances are involved. *Biometrika*, 34(1-2):28–35, 1947.

[WS06] John Winn and Jamie Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 37–44, New York, NY, USA, June 2006.

[XCB00] X. G. Xu, T. C. Chao, and A. Bozkurt. VIP-Man: an image-based whole-body adult male model constructed from color photographs of the visible human project for multi-particle monte carlo calculations. *Health Physics*, 78(5):476–486, May 2000.

[YPH+06] Paul A. Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C. Gee, and Guido Gerig. User-guided 3d active contour segmentation of anatomical structures: significantly improved efficiency and reliability. *Neuroimage*, 31(3):1116–1128, 2006.

[ZBS01] Yongyue Zhang, Michael Brady, and Stephen Smith. Segmentation of brain MR images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging*, 20(1):45–57, January 2001.

[Zwi95] Uri Zwick. The smallest networks on which the ford-fulkerson maximum flow procedure may fail to terminate. *Theoretical Computer Science*, 148(1):165–170, 1995.

Publications of the candidate

R. Kéchichan, S. Valette, M. Desvignes, and R. Prost. Shortest-Path Constraints for 3D Multi-Object Semi-Automatic Segmentation via Clustering and Graph Cut. *IEEE Transactions on Image Processing*, 2013. Accepted.

H. Jacinto, R. Kéchichan, M. Desvignes, R. Prost, and S. Valette. A Web Interface for 3D Visualization and Interactive Segmentation of Medical Images. In *17th International Conference on 3D Web Technology*, pages 51–58, Los-Angeles, USA, August 2012.

R. Kéchichan, S. Valette, M. Desvignes, and R. Prost. Efficient Multi-Object Segmentation of 3D Medical Images Using Clustering and Graph Cuts. In *18th IEEE International Conference on Image Processing*, pages 2149–2152, Brussels, Belgium, September 2011.

R. Kéchichan, S. Valette, M. Desvignes, and R. Prost. Segmentation 3D Multi-Objet par "Graph Cut" et Diagramme de Voronoï Centroidal. In *GRETSI XXIIIe Colloque*, Bordeaux, France, September 2011.

INSA, Direction de Recherche, Écoles Doctorales, Quinquennal 2011-15

Sigle	École Doctorale	Responsable et ses Coordonnées
CHIMIE	Chimie de Lyon http://www.edchimie-lyon.fr INSA: R. Gourdon	Mr Jean-Marc Lancelin Université de Lyon - Collège Doctoral bât. ESCPE 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cedex Tél: 04.72.43.13.95 directeur@edchimie-lyon.fr
E.E.A.	Électronique, Électrotechnique Automatique http://edeea.ec-lyon.fr Secrétariat: M. C. Havgoudoukian eea@ec-lyon.fr	Mr Gérard Scorletti École Centrale de Lyon 36 av Guy de Collongue 69134 ÉCULLY Cedex Tél: 04.72.18.60.97 Fax : 04.78.43.37.17 gerard.scorletti@ec-lyon.fr
E2M2	Évolution, Écosystème, Microbiologie, Modélisation http://e2m2.universite-lyon.fr INSA: H. Charles	Mme Gudrun Bornette CNRS UMR 5023 LEHNA Université Claude Bernard Lyon 1 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cedex Tél: 04.72.43.12.94 e2m2@biomserv.universite-lyon.fr
EDISS	Interdisciplinaire Sciences-Santé http://ww2.ibcp.fr/ediss Secrétariat: S. Ait Chalal INSA: M. Lagarde	Mr Didier Revel Hôpital Louis Pradel bât. Central 28 av Doyen Lépine 69677 BRON Cedex Tél: 04.72.68.49.09 Fax: 04.72.35.49.16 didier.revel@creatis.univ-lyon1.fr
INFOMATHS	Informatique et Mathématiques http://infomaths.univ-lyon1.fr	Mr Johannes Kellendonk Université Claude Bernard Lyon 1 bât. Braconnier 43 bd du 11 novembre 1918 69622 VILLEURBANNE Cedex Tél: 04.72.44.82.94 Fax: 04.72.43.16.87 infomath@univ-lyon1.fr
Matériaux	Matériaux de Lyon Secrétariat: M. Laboune ed.materiaux@insa-lyon.fr	Mr Jean-Yves Buffière INSA de Lyon MATEIS, bât. Saint Exupéry 7 av Jean Capelle 69621 VILLEURBANNE Cedex Tél: 04.72.43.83.18 Fax: 04.72.43.85.28 jean-yves.buffiere@insa-lyon.fr
MEGA	Mécanique, Énergétique, Génie Civil, Acoustique Secrétariat: M. Laboune mega@insa-lyon.fr	Mr Philippe Boisse INSA de Lyon LAMCOS, bât. Jacquard 25 bis av Jean Capelle 69621 VILLEURBANNE Cedex Tél: 04.72.43.71.70 Fax: 04.72.43.72.37 philippe.boisse@insa-lyon.fr
ScSo	Histoire, Géographie, Aménagement, Urbanisme, Archéologie, Science Politique, Sociologie, Anthropologie Secrétariat: V. Polsinelli INSA: J. Y. Toussaint	Mr Lionel Obaida Université Lyon 2 86 rue Pasteur 69365 LYON Cedex 07 Tél: 04.78.69.72.76 Fax: 04.37.28.04.48 lionel.obaida@univ-lyon2.fr

FOLIO ADMINISTRATIF

THÈSE SOUTENUE DEVANT L'INSTITUT NATIONAL DES SCIENCES APPLIQUÉES DE LYON

NOM: KÉCHICHIAN — Date de soutenance: 2 juillet, 2013

Prénoms: Razmig

TITRE: A Priori de Structure pour la Segmentation Multi-objet d'Images Médicales 3D par Partition d'Images et Coupure de Graphes

NATURE: Doctorat — Numéro d'ordre: 2013-isal-0058

École doctorale: EEA

Spécialité: Traitement d'images

RÉSUMÉ: Nous développons une méthode générique semi-automatique multi-objet de segmentation d'image par coupure de graphe visant les usages médicaux de routine, allant des tâches impliquant quelques objets dans des images 2D, à quelques dizaines dans celles 3D quasi corps entier. La formulation souple de la méthode permet son adaptation simple à une application donnée. En particulier, le modèle d'a priori de proximité que nous proposons, défini à partir des contraintes de paires du plus court chemin sur le graphe d'adjacence des objets, peut facilement être adapté pour tenir compte des relations spatiales entre les objets ciblés dans un problème donné. L'algorithme de segmentation peut être adapté aux besoins de l'application en termes de temps d'exécution et de capacité de stockage à l'aide d'une partition de l'image à segmenter par une tesselation de Voronoï efficace et contrôlable, établissant un bon équilibre entre la compacité des régions et le respect des frontières des objets. Des évaluations et comparaisons qualitatives et quantitatives avec le modèle de Potts standard confirment que notre modèle d'a priori apporte des améliorations significatives dans la segmentation d'objets distincts d'intensités similaires, dans le positionnement précis des frontières des objets ainsi que dans la robustesse de segmentation par rapport à la résolution de partition. L'évaluation comparative de la méthode de partition avec ses concurrentes confirme ses avantages en termes de temps d'exécution et de qualité des partitions produites. Par comparaison avec l'approche appliquée directement sur les voxels de l'image, l'étape de partition améliore à la fois le temps d'exécution global et l'empreinte mémoire du processus de segmentation jusqu'à un ordre de grandeur, sans compromettre la qualité de la segmentation en pratique.

MOTS-CLÉS: segmentation d'images, partition d'images, modèle de Markov, a priori spatial, coupure de graphes.

Laboratoire de recherche: CREATIS; INSA de Lyon, GIPSA-lab; Grenoble INP.

Directeurs de thèse: Rémy PROST, Michel DESVIGNES.

Président de jury:

Composition du jury:

Philippe BOLON	Professeur, Polytech Annecy-Chambéry	Président
Fabrice HEITZ	Professeur, Université de Strasbourg	Rapporteur
Jean-Philippe THIRAN	Professeur, EPF, Lausanne	Rapporteur
Sébastien VALETTE	Chargé de recherche, CNRS	Co-encadrant
Michel DESVIGNES	Professeur, Grenoble INP	Co-directeur
Rémy PROST	Professeur, INSA de Lyon	Co-directeur

Structural Priors for Multiobject Semiautomatic Segmentation of Three-dimensional Medical Images via Clustering and Graph Cut Algorithms

Abstract: We develop a generic Graph Cut-based semiautomatic multiobject image segmentation method principally for use in routine medical applications ranging from tasks involving few objects in 2D images to fairly complex near whole-body 3D image segmentation. The flexible formulation of the method allows its straightforward adaption to a given application. In particular, the graph-based vicinity prior model we propose, defined as shortest-path pairwise constraints on the object adjacency graph, can be easily reformulated to account for the spatial relationships between objects in a given problem instance. The segmentation algorithm can be tailored to the runtime requirements of the application and the online storage capacities of the computing platform by an efficient and controllable Voronoi tessellation clustering of the input image which achieves a good balance between cluster compactness and boundary adherence criteria. Qualitative and quantitative comprehensive evaluation and comparison with the standard Potts model confirm that the vicinity prior model brings significant improvements in the correct segmentation of distinct objects of identical intensity, the accurate placement of object boundaries and the robustness of segmentation with respect to clustering resolution. Comparative evaluation of the clustering method with competing ones confirms its benefits in terms of runtime and quality of produced partitions. Importantly, compared to voxel segmentation, the clustering step improves both overall runtime and memory footprint of the segmentation process up to an order of magnitude virtually without compromising the segmentation quality.

Keywords: image segmentation, image clustering, Markov model, spatial prior, Graph Cut

A Priori de Structure pour la Segmentation Multi-objet d'Images Médicales 3D par Partition d'Images et Coupure de Graphes

Résumé: Nous développons une méthode générique semi-automatique multi-objet de segmentation d'image par coupure de graphe visant les usages médicaux de routine, allant des tâches impliquant quelques objets dans des images 2D, à quelques dizaines dans celles 3D quasi corps entier. La formulation souple de la méthode permet son adaptation simple à une application donnée. En particulier, le modèle d'a priori de proximité que nous proposons, défini à partir des contraintes de paires du plus court chemin sur le graphe d'adjacence des objets, peut facilement être adapté pour tenir compte des relations spatiales entre les objets ciblés dans un problème donné. L'algorithme de segmentation peut être adapté aux besoins de l'application en termes de temps d'exécution et de capacité de stockage à l'aide d'une partition de l'image à segmenter par une tesselation de Voronoï efficace et contrôlable, établissant un bon équilibre entre la compacité des régions et le respect des frontières des objets. Des évaluations et comparaisons qualitatives et quantitatives avec le modèle de Potts standard confirment que notre modèle d'a priori apporte des améliorations significatives dans la segmentation d'objets distincts d'intensités similaires, dans le positionnement précis des frontières des objets ainsi que dans la robustesse de segmentation par rapport à la résolution de partition. L'évaluation comparative de la méthode de partition avec ses concurrentes confirme ses avantages en termes de temps d'exécution et de qualité des partitions produites. Par comparaison avec l'approche appliquée directement sur les voxels de l'image, l'étape de partition améliore à la fois le temps d'exécution global et l'empreinte mémoire du processus de segmentation jusqu'à un ordre de grandeur, sans compromettre la qualité de la segmentation en pratique.

Mots-Clés: segmentation d'images, partition d'images, modèle de Markov, a priori spatial, coupure de graphes.

CREATIS: Centre de Recherche en Acquisition et Traitement de l'Image pour la Santé
Université de Lyon; CNRS UMR 5220; Inserm U1044; INSA-Lyon; Université Lyon 1
7 avenue Jean Capelle, 69621 Villeurbanne, France