UNIVERSITÉ ━━━ ━PARIS-EST

Université Paris Est

École doctorale MSTIC

Thèse soumise pour obtenir le grade de
Docteur de l'Université Paris Est en Informatique

Yongchao XU

Directeur de thèse: Laurent Najman

Co-directeur de thèse: Thierry Géraud

Tree-based shape spaces: Definition and applications in image processing and computer vision

Espaces des formes basés sur des arbres : définition et applications en traitement d'images et vision par ordinateur

Soutenue le 12 décembre 2013

Membres du jury:

Jean Serra	-	(Président)
Lionel Moisan	-	(Rapporteur)
Philippe Salembier	-	(Rapporteur)
Beatriz Marcotegui	-	(Examinateur)
Michael H.F. Wilkinson	-	(Examinateur)
Pascal Monasse	-	(Invité)
Laurent Najman	-	(Directeur de thèse)
Thierry Géraud	-	(Co-Directeur de thèse)

Acknowledgments

I would like to thank Lionel Moisan and Philippe Salembier, who accepted to review this thesis. I am grateful to Jean Serra, president of the jury, and to Beatriz Marcotegui, Michael H.F. Wilkinson and Pascal Monasse, who agreed to be member of my jury. I would like to acknowledge Coloma Ballester who had accepted to be part of the jury but whose presence was made impossible.

I am deeply grateful to Thierry Géraud and Laurent Najman who supervised me during these three years. I was extremely lucky to have both of them as my advisors. The list of the reasons why i would like to thank both of them is too long to be written here, and can hardly be expressed with worlds. We have spent numerous hours together working at ESIEE and EPITA. Thierry Géraud was the first one who introduced me to the world of computer science research. I particularly want to thank him for captivating me with many interesting problems and ideas. He supervised my work since my master's degree internship. He taught me lots of computer skills and how to make a good presentation, from which i will benefit for all my life. He encouraged me to pursuer any subject that I was interested in. He also helped me a lot for my life in France. For all these and for many other reasons, I am endlessly grateful to him. Laurent Najman has done a remarkable work at supervising all the steps of this work. He proposed lots of interesting and exiting ideas that motivated me to dig deeply the problems I was studying, and I benefited a lot from his deep insights to those problems. This PhD work can not be finished without many constructive discussions with him. He has spent numerous nights with me via Email and Skype on writing papers, and he taught me how to write a good scientific paper. It has been a real pleasure to work with him. I am very sincerely grateful to him.

I enjoyed working with Pascal Monasse, and I wish to thank him for his guidance and insight he brought to the problems I was studying. I would like to acknowledge Pierre Moulon for providing me some codes and datasets he developed, and to Michele Bianchi for his preparation of the cellular images.

I thank all my colleagues from LRDE, and especially to Daniela Becker and Olivier Ricou for all the administrative things and the revision of this thesis by Daniela. I am deeply grateful to Guillaume Lazzara, Edwin Carlinet, Roland Levillain, and Jonathan Fabrizio for so many of their helps during the three years, to Didier Verna for his revision of English problems in several of my papers, and to Alexandre Duret-Lutz, Akim Demaille, Reda Dehak, Myriam Robert-Seidowsky, Ala-Eddine Ben-Salem, Etienne Renault, Stefania Calarasanu and Nicolas Bourtry for their encouragements. All the colleagues helped me a lot for my life in France,

and explained to me numerous questions about french culture. I cannot imagine this period of my life in another setting than LRDE, which has become a kind of my second home in France.

I am grateful to all the members of the A3SI laboratory of ESIEE, who hosted me during these three years in a friendly atmosphere. I would like to thank Jean Serra, Hugues Talbot, Yukiko Kenmochi, Jean Cousty, Benjamin Perret, Michel Couprie, Gilles Bertrand, and Silvio Guimaraes for all the constructive and interesting discussions on the subjects of scientific research, as well as general education. Many thanks to all my office mates Vincent Bismuth, Eloise Grossiord, Odyssée Merveille, Elodie Puybareau for their kindness and the tea. A special thanks to all the Ph.D. candidates, particularly Ravi Kiran, Hoai Diem Phuc Ngo, Imen Melki, and former PhD students for their encouragements and their friendship.

I would like to give a special thanks to Geneviève Guillot and Anne Leroy-Willig. Geneviève Guillot was my second year of french engineering's degree internship advisor. She was the first one who introduced me to the world of research. She encouraged me to pursuer a research master's degree and a PhD work. Anne Leroy-Willig introduced me to LRDE where I have done this PhD work and met the enthusiastic colleagues.

And of course, I would like to acknowledge all my friends, my girlfriend Yueyi Wang, my parents Jinxiang Xu and Dongfeng Huang, and my sister Yongfang Xu, for their friendship and love, their support and their patience when I was not available, especially during the last stages of the redaction of this manuscript. None of this could have been made without their enduring support.

Abstract

In a large number of applications, the processing relies on objects or areas of interest, therefore the pixel-based image representation is not well adapted. These applications would benefit from a region-based processing. Early examples of region-based processing can be found in the field of image segmentation, such as the quadtree. Recently, in mathematical morphology, the connected operators have received much attention. They are region-based filtering tools that act by merging flat zones. They have good contour preservation properties in the sense that they do not create any new boundaries, neither do they shift the existing ones.

One popular implementation for connected operators relies on tree-based image representations, especially threshold decomposition representations and hierarchical representations. These tree-based image representations are widely used in many image processing and computer vision applications. Tree-based connected operators consist in constructing a set of nested or disjoint connected components, followed by a filtering of these connected components based on an attribute function characterizing each connected component. Finally, the filtered image is reconstructed from the simplified tree composed of the remaining connected components.

In the work presented in this thesis, we propose to expand ideas about tree-based connected operators. We introduce the notion of tree-based shape spaces, built from tree-based image representations. Many state-of-the-art methods relying on tree-based image representations consist of analyzing this shape space. A first consequence of this change of point of view is our proposition of a local feature detector, called the tree-based Morse regions (TBMR). It can be seen as a variant of the MSER method. The selection of TBMRs is based on topological information, and hence it extracts the regions independently of the contrast, which makes it truly contrast invariant and quasi parameter free. The accuracy and robustness of the TBMR approach are demonstrated by the repeatability test and by applications to image registration and 3D reconstruction, as compared to some state-of-the-art methods.

The basic idea of the main proposition in this thesis is to apply connected filters to the shape space. Such processing is called the framework of shape-based morphology. It is a versatile framework that deals with region-based image representations. It has three main consequences. 1) For filtering purposes, the classical existing tree-based connected operators are generalized. Indeed, the framework encompasses classical existing connected operators by attributes. Besides, this also allows us to propose two classes of novel connected operators: shape-based lower/upper levelings

and shapings. 2) This framework can be used for object detection/segmentation by selecting relevant points in the shape space. 3) We can also employ this framework to transform the hierarchies using the extinction values, in order to obtain a hierarchical simplification or segmentation.

Some applications are developed using the framework of shape-based morphology to demonstrate its usefulness. The applications of the shape-based filtering for retinal image analysis show that a mere filtering step, which we compare to more evolved processings, achieves state-of-the-art results. An efficient shaping used for image simplification is proposed by minimizing Mumford-Shah functional subordinated to the topographic map. For object detection/segmentation, we proposed a context-based energy estimator that is suitable to characterize object meaningfulness. Eventually, we extend the hierarchy of constrained connectivity using the aspect of hierarchy transformation.

Keywords: image processing, mathematical morphology, connected filters, tree representations, hierarchies, local feature detections, retinal image analysis, image simplification/segmentation.

Résumé

Dans un grand nombre d'applications, le traitement repose sur des objets ou des zones d'intérêt, et la représentation d'image à base de pixels n'est pas bien adaptée. Ces applications pourraient bénéficier d'un traitement basé sur régions. Les premiers exemples de traitement basé sur région peuvent être trouvés dans le domaine de la segmentation d'image, par exemple, le quadtree. Récemment, en morphologie mathématique, les opérateurs connexes ont reçu beaucoup d'attention. Ce sont des outils de filtrage basé sur région qui agissent en fusionnant des zones plates. Ils ont de bonnes propriétés de conservation de contour dans le sens qu'ils ne créent pas de nouveaux contours, et qu'ils ne déplacent pas les contours existants.

Une implémentation populaire des opérateurs connexes repose sur une représentation d'image à base d'arbres, notamment les représentations basées sur la décomposition par seuillage et les représentations hiérarchiques. Ces représentations d'image à base d'arbres sont largement utilisées dans de nombreuses applications de traitement d'image et de vision par ordinateur. Les opérateurs connexes à base d'arbres sont construites par la construction d'un ensemble de composantes connexes emboîtées ou disjointes, suivi d'un filtrage de ces composantes connexes basé sur une fonction d'attribut caractérisant chaque composante connexe. Finalement, l'image filtrée est reconstruite à partir de l'arbre simplifié, composé des composantes connexes restantes.

Dans le travail présenté dans cette thèse, nous proposons d'élargir les idées des opérateurs connexes à base d'arbres. Nous introduisons la notion d'espaces de formes à base d'arbres, construit à partir des représentations d'image à base d'arbres. De nombreuses méthodes de l'état de l'art, s'appuyant sur ces représentations d'images à base d'arbres, consistent à analyser cet espace de forme. Une première conséquence de ce changement de point de vue est notre proposition d'un détecteur de caractéristiques locales, appelé les "tree-based Morse regions" (TBMR). Cette approache peut être considérée comme une variante de la méthode des MSER. La sélection des TBMRs est basé sur des informations topologiques, et donc extrait les régions indépendamment du contraste, ce qui la rend vraiment invariante aux changements de contraste; de plus, la méthode peut être considérée sans paramètres. La précision et la robustesse de l'approche TBMR sont démontrées par le test de reproductibilité et par des applications au recalage d'image et à la reconstruction 3D, en comparaison des méthodes de l'état de l'art.

L'idée de base de la proposition principale dans cette thèse est d'appliquer les opérateurs connexes à l'espace des formes. Un tel traitement est appelé la mor-

phologie basée sur la forme. Ce cadre polyvalent traite des représentations d'images à base de région. Il a trois conséquences principales. 1) Dans un but de filtrage, il s'agit d'une généralisation des opérateurs connexes à base d'arbres. En effet, le cadre englobe les opérateurs connexes classiques par attributs. En outre, il permet également de proposer deux nouvelles classes d'opérateurs connexes: nivellements inférieurs/supérieurs à base de forme et "shapings". 2) Ce cadre peut être utilisé pour la détection/segmentation d'objets en sélectionnant les points pertinents dans l'espace des formes. 3) Nous pouvons également utiliser ce cadre pour transformer les hiérarchies en utilisant les valeurs d'extinction, obtenant ainsi une simplification/segmentation hiérarchique.

Afin de montrer l'utilité de l'approche proposée, plusieurs applications sont développées. Les applications à l'analyse d'images rétinienne de filtrage basé sur la forme montrent qu'une simple étape de filtrage, comparée à des traitements plus évolués, réalise des résultats au niveau de l'état de l'art. Une application de "shaping" pour la simplification d'image est proposée, fondée sur une minimisation de la fonctionnelle de Mumford-Shah subordonnée à l'arbre de formes. Pour la détection/segmentation d'objets, nous proposons un estimateur de l'énergie basée sur le contexte. Cet estimateur est approprié pour caractériser la signification d'objet. Enfin, nous étendons le cadre de la connectivité contrainte en utilisant l'aspect de transformation de hiérarchie.

Mots-clefs: traitement d'images, morphologie mathématique, opérateurs connexes, représentations d'arbre, hiérarchies, extraction de caractéristiques locales, analyse d'image rétinienne, simplification/segmentation d'image.

Résumé long

Résumé

Le cadre classique des filtres connexes consiste à enlever d'un graphe certaines de ses composantes connexes. Pour appliquer ces filtres, il est souvent utile de transformer une image en un arbre de composantes, et on élague cet arbre pour simplifier l'image de départ. Les arbres ainsi formés ont des propriétés remarquables pour la vision par ordinateur. Une première illustration de leur intérêt est la définition d'un détecteur de zones d'intérêt, vraiment invariant aux changements de contraste, qui nous permet d'obtenir des résultats à l'état de l'art en recalage d'images et en reconstruction 3D à base d'images. Poursuivant dans l'utilisation de ces arbres, nous proposons d'élargir le cadre des filtres connexes. Pour cela, nous introduisons la notion d'espaces des formes basés sur des arbres : au lieu de filtrer des composantes connexes du graphe correspondant à l'image, nous proposons de filtrer des composantes connexes du graphe donné par l'arbre des composantes de l'image. Ce cadre général, que nous appelons morphologie basée sur les formes, peut être utilisé pour la détection et la segmentation d'objets, l'obtention de segmentations hiérarchiques, et le filtrage d'images. De nombreuses applications et illustrations montrent l'intérêt de ce cadre.

Mots Clefs

Traitement d'image, morphologie mathématique, filtrage connexe, représentation arborescente, segmentation hiérarchique, détection de zones d'intérêt.

A Introduction

En morphologie mathématique, les opérateurs connexes [Serra 1993, Salembier 1995, Salembier 1998, Salembier 2009] ont reçu beaucoup d'attention. Ce sont des outils de filtrage basés sur régions qui agissent en fusionnant des zones plates. Ils ont de bonnes propriétés de conservation des contours dans le sens qu'ils ne créent pas de nouveaux contours, et qu'ils ne déplacent pas les contours existants.

Une implémentation populaire des opérateurs connexes repose sur les représentations d'images en arbres [Salembier 1998, Monasse 2000b, Salembier 2000]. Dans la pratique, pour des questions d'efficacité, les algorithmes des filtres connexes reposent sur ces représentations arborescentes, et comprennent trois étapes : une construction d'arbre représentant l'image à traiter, l'élagage de cet arbre, et la reconstruction de
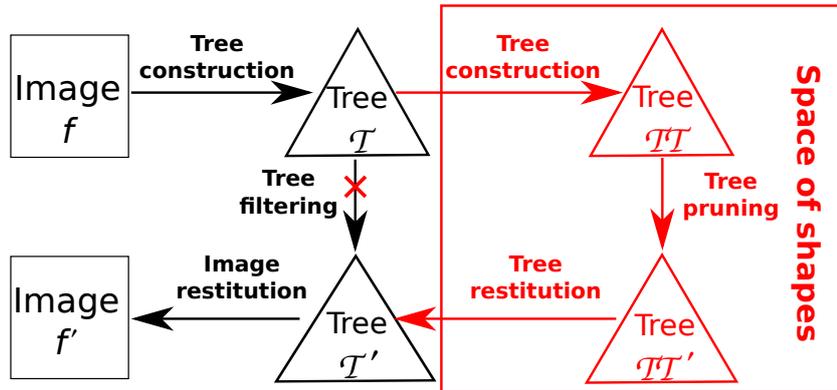
Figure 1: Opérateurs connexes basés sur l'arbre (chemin noir) et notre proposition (en ajoutant le chemin rouge).

l'image correspondant à l'arbre simplifié. Un exemple d'un tel processus est illustré en Figure 1 par le chemin noir. Dans le travail présenté dans cette thèse, nous proposons tout d'abord une première illustration de l'intérêt de ces arbres : un détecteur de zones d'intérêt que nous appelons "*tree-based Morse regions*" (TBMR). Cette approche peut être considérée comme une variante de la méthode MSER [Matas 2002]. La sélection des TBMRs est basée sur l'information topologique ; les régions sont donc extraites indépendamment de leur contraste. Au final, la méthode est donc totalement invariante aux changements de contraste des images à traiter. De plus, TBMR est quasiment sans paramètre. Nous avons obtenu des résultats au niveau de l'état de l'art pour le test de répétabilité [Mikolajczyk 2005], et pour des applications au recalage d'image et à la reconstruction 3D à base d'images.

En allant plus loin dans l'utilisation de ces arbres, nous proposons d'élargir le cadre des opérateurs connexes. Pour cela, nous introduisons la notion d'espaces des formes basés sur des arbres. Un espace des formes est un graphe connexe non-dirigé dont chaque nœud correspond à un nœud (composante connexe) dans l'arbre, et la relation de voisinage est donnée par la relation de parenté de l'arbre. L'idée principale dans cette thèse est d'appliquer les opérateurs connexes à l'espace des formes, au lieu de s'appuyer directement sur l'espace de l'image. Un tel traitement est appelé la morphologie basée les formes. C'est un cadre général ayant trois conséquences principales. 1) Ce cadre peut être utilisé pour la détection et la segmentation d'objets en sélectionnant les nœuds pertinents dans l'espace des formes. 2) Nous pouvons également utiliser ce cadre pour obtenir des simplifications d'images et des segmentations hiérarchiques. 3) Dans un but de filtrage, il s'agit d'une généralisation des opérateurs connexes à base d'arbres. En effet, le cadre englobe les opérateurs connexes classiques par attributs. En outre, il permet également de proposer deux
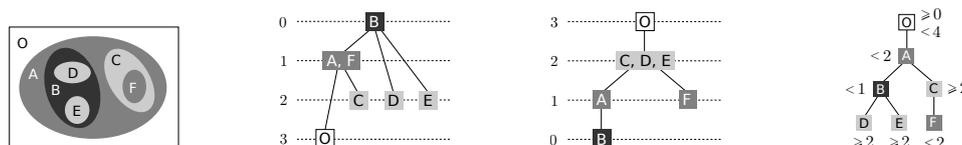
Figure 2: Une image (à gauche), ses Max-tree et Min-tree (au milieu), et son arbre des formes (à droite).

nouvelles classes d'opérateurs connexes : des nivellements inférieurs/supérieurs à base de formes et des *"shapings"*. L'intérêt de ces trois aspects du cadre proposé est démontré par des applications et des illustrations.

Le reste de ce résumé long est structuré de la façon suivante. Nous rappelons l'état de l'art dans la Section B, suivie par notre proposition d'un détecteur de zones d'intérêt dans la Section C. Puis nous introduisons la notion d'espaces des formes dans la Section D, et nous détaillons le cadre de la morphologie basée sur les formes dans la Section E. Nous montrons quelques illustrations et applications de ce cadre dans la Section F. Enfin, la section G conclut le résumé long.

B Travaux liés

B.1 Représentations arborescentes

Nous distinguons deux types de représentations d'images en arbres. Le premier type est basé sur une décomposition d'image par seuillage ; le second type est la famille des hiérarchies de segmentations, une hiérarchie étant constitué d'un ensemble de segmentations allant des plus fines aux plus grossières.

Arbres basés sur la décomposition d'image par seuillage. Pour tout $\lambda \in \mathbb{R}$, l'ensemble supérieur $\mathcal{X}_\lambda$ et l'ensemble inférieur $\mathcal{X}^\lambda$ d'une image $f : \Omega \to \mathbb{R}$ sont définis respectivement par $\mathcal{X}_\lambda(f) = \{p \in \Omega \mid f(p) \geq \lambda\}$ et $\mathcal{X}^\lambda(f) = \{p \in \Omega \mid f(p) \leq \lambda\}$. Les deux ensembles de niveaux supérieur et inférieur ont une structure d'inclusion naturelle : $\forall \lambda_1 \leq \lambda_2$, $\mathcal{X}_{\lambda_1} \supseteq \mathcal{X}_{\lambda_2}$ and $\mathcal{X}^{\lambda_1} \subseteq \mathcal{X}^{\lambda_2}$, ce qui conduit à deux représentations distinctes d'une image, le Max-tree et le Min-tree [Salembier 1998]. Un autre arbre, appelé arbre des formes, a été introduit dans [Monasse 2000b]. Une *forme* est définie comme une composante connexe d'un ensemble supérieur ou inférieur dans laquelle ses trous ont été bouchés. La relation d'inclusion entre les formes donne un arbre unique qui est l'arbre des formes. Un exemple de ce type d'arbre est donné en Figure 2.

x

Hiérarchie de segmentations. Les hiérarchies de segmentations sont devenues très populaires dans le domaine de la segmentation d'images [Guigues 2006]. Un exemple de hiérarchie de segmentations est l'arbre binaire de partitions [Salembier 2000], créé par un processus de fusion de régions. Un deuxième arbre populaire est l'arbre couvrant de poids minimal [Kruskal 1956]. Un dernier exemple est l'α-tree [Ouzounis 2011a], connu aussi sous le nom de hiérarchie de connectivité contrainte [Soille 2008]. Toutes ces hiérarchies de segmentations peuvent produire des cartes de saillance [Najman 1996], et peuvent être données sous la forme équivalente de ligne de partage des eaux ultramétrique [Najman 1996, Najman 2011].

Toutes les représentations en arbres sont multi-échelles au sens de la théorie d'analyse ensembles-échelles [Guigues 2006], ce qui fournit un espace réduit de recherche d'objets. Par ailleurs, les arbres basés sur une décomposition par seuillages sont covariant aux transformations (topologiques) continues, et ils sont aussi invariants aux transformations affines des intensités d'image. De plus, l'arbre des formes et l'α-tree sont des structures auto-duales.

B.2 Opérateurs connexes

Les opérateurs connexes [Serra 1993, Salembier 1995, Salembier 1998, Salembier 2009] ne modifient pas la valeur des pixels individuellement mais agissent au niveau des composantes connexes où le signal est constant, connue par le terme zones plates [Salembier 1995]. Les opérateurs connexes fusionnent les zones plates adjacentes. Ils ne peuvent pas introduire un contour qui n'est pas présent dans l'image originale. Ils ne peuvent pas non plus déplacer les contours existants. Les opérateurs connexes ont été originellement définis via le concept de partition des zones plates. Désignons par $\mathcal{P}$ une partition et par $\mathcal{P}(p)$ la région de la partition qui contient le pixel p. Un ordre partiel entre partitions peut être créé : $\mathcal{P}_1$ est plus "fine" que $\mathcal{P}_2$ (ce que l'on note $\mathcal{P}_1 \sqsubseteq \mathcal{P}_2$), si $\forall p, \mathcal{P}_1(p) \subseteq \mathcal{P}_2(p)$. Les opérateurs connexes sont alors définis comme suit :

Définition 1 *Un opérateur φ est connexe si la partition des zones plates $\mathcal{P}_f$ de l'image f est toujours plus fine que celle $\mathcal{P}_{\varphi(f)}$ de l'image $\varphi(f)$.*

Une implémentation populaire des opérateurs connexes repose sur les représentations arborescentes. Le filtrage consiste alors à concevoir un attribut caractérisant la forme des composantes ou le degré de ressemblance entre la forme des composantes vis-à-vis d'une forme attendue. Il y a deux types d'approches pour filtrer l'arbre (donc pour filter l'image). Le premier type est l'élagage d'arbres. L'autre repose sur le seuillage par attribut.
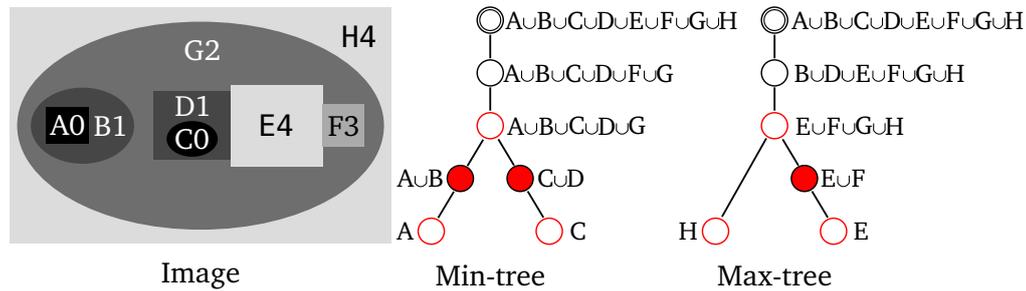
Figure 3: Une image synthétique, son Min-tree (au milieu) et son Max-tree (à droite). Les régions critiques sont représentées par des cercles rouges : ce sont des nœuds ayant plus d'un enfant et et les feuilles. Les TBMRs sont les régions correspondants aux nœuds remplis en rouge.

B.3 Applications liées aux représentations arborescentes

Il existe de nombreuses applications utilisant les représentations arborescentes en traitement d'images et en vision par ordinateur. Nous détaillons brièvement la méthode MSER [Matas 2002] dont la définition originale ne repose pas sur les représentations en arbre. Cependant, cette méthode est facilement compréhensible en utilisant le Max-tree et le Min-tree : elle extrait les régions (nœuds) qui correspondent à des minima locaux de la fonction de stabilité le long du trajet vers la racine de l'arbre. La fonction de stabilité d'un nœud $\mathcal{N}$ est donnée par $\mathcal{A}_q(\mathcal{N}) = (|\mathcal{N}_\Delta^+| - |\mathcal{N}_\Delta^-|)/|\mathcal{N}|$, où $|\cdot|$ est l'aire, Δ est un écart d'intensité de gris, $\mathcal{N}_\Delta^+$ et $\mathcal{N}_\Delta^-$ sont respectivement l'ancêtre le plus bas et le descendant le plus haut de telle sorte que $|f(\mathcal{N}_\Delta^+) - f(\mathcal{N})| \geq \Delta$ et $|f(\mathcal{N}) - f(\mathcal{N}_\Delta^-)| \geq \Delta$. Il est rapporté dans [Mikolajczyk 2005] que la méthode MSER atteint une répétabilité et une précision au niveau de l'état de l'art.

C "Tree-based Morse regions"

D'après la théorie de Morse [Milnor 1963], la topologie d'une image f est directement liée à l'analyse des points critiques : les minima, maxima, et points-selles de f. Plus précisément, suivant [Caselles 2009], nous proposons de choisir des régions critiques dans les Max-tree $S_{\mathcal{T}_M}$ et Min-tree $S_{\mathcal{T}_m}$. Les régions critiques sont les feuilles de l'arbre et les régions résultant d'une fourche dans l'arbre. Pour chaque région critique, une échelle est sélectionnée. Nous associons à chaque région critique R_c la plus grande région contenant R_c et topologiquement équivalente dans son arbre. Nous appelons notre méthode "tree-based Morse regions" (TBMR). Un exemple d'extraction de TBMRs est illustré en Figure 3. Dans la pratique, nous ne considérons pas les TBMRs qui sont trop petits ou trop grands. Les rejets de petites régions est effectuée avant l'analyse, ce qui signifie qu'ils ne pas contribuent
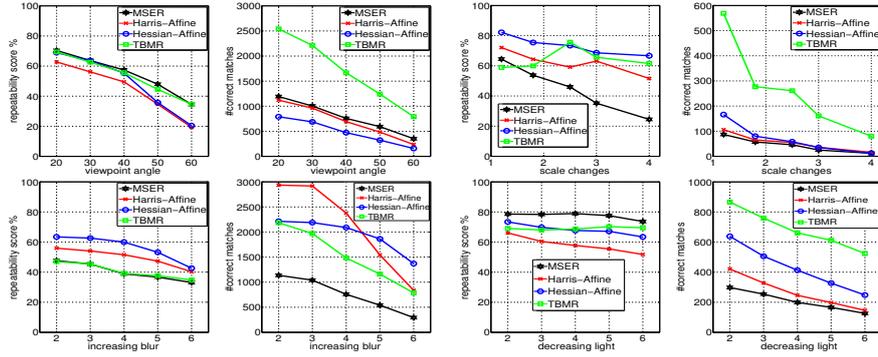
Figure 4: Le score de répétabilité (à gauche) et le nombre de correspondances (à droite) pour les séquences *Wall, Bark, Trees, Leuven* (de haut en bas, et de gauche à droite).

aux changements topologiques. Ce processus élimine aussi un peu de bruit sans modifier les autres composantes. Dans nos expériences, nous avons toujours fixé cette limite inférieure à 30 pixels, et la limite supérieure de taille des composantes considérées est fixée à 1% de la taille de l'image.

La méthode TBMR a plusieurs avantages : comme elle utilise seulement une information topologique, elle est indépendante du contraste de l'image. Elle est également covariante aux transformations (topologiques) continues tels que les translations, les changements d'échelle, ou les rotations. Elle est également robuste aux distorsions géométriques locales. En outre, elle est essentiellement exempte de paramètres : seulement deux paramètres non significatifs sont appliqués (pour ignorer les régions trop petites ou trop grandes).

Des tests de répétabilité [Mikolajczyk 2005] (Figure 4) montrent la précision et la robustesse de l'approche TBMR. Nous obtenons des résultats à l'état de l'art pour l'application de TBMR au recalage d'image et à la reconstruction 3D à base d'images. Le TBMR est meilleur en particulier que MSER pour ces applications.

D Espaces des formes

Il existe des caractéristiques communes à tous les arbres. Ces arbres sont composés d'un ensemble des régions (composantes connexes), de petites à grandes. Chaque région est représentée par un nœud dans l'arbre. Le nœud racine de l'arbre représente le domaine de l'image entière. Par ailleurs, pour deux régions distinctes R_1 et R_2 représentées par deux nœuds $\mathcal{N}_1$ et $\mathcal{N}_2$ de l'arbre, nous avons une propriété d'emboîtement : $R_1 \cap R_2 \neq 0 \Rightarrow R_1 \subset R_2$ ou $R_2 \subset R_1$. À l'exception du nœud racine, chaque nœud $\mathcal{N}$ de l'arbre a un parent unique $\mathcal{N}_p$, et il existe une arête

$(\mathcal{N}, \mathcal{N}_p)$ reflétant le lien de parenté, c'est-à-dire, la relation d'inclusion entre les deux régions qu'ils représentent. En conséquence, les représentations arborescentes d'image peuvent être considérées comme des graphes connexes, ce qui donne la définition d'espace des formes basés sur l'arbre $S_{\mathcal{T}}$.

Définition 2 *Un espace des formes basé sur un arbre $\mathcal{T}$ noté par $S_{\mathcal{T}}$ est un graphe non-dirigé connexe, où $V(S_{\mathcal{T}}) = \{\mathcal{N} \mid \mathcal{N} \in \mathcal{T}\}$, et $E(S_{\mathcal{T}})$ est composé de l'ensemble des arêtes qui modélisent la relation d'inclusion.*

Les espaces des formes basés sur les arbres ont quelques propriétés fondamentales. Tout d'abord, un espace des formes $S_{\mathcal{T}}$ est équivalent à une image f, dans le sens où l'image f peut être reconstruite à partir de l'arbre $\mathcal{T}$. De plus, tous les espaces des formes satisfont le principe de causalité, certainement le principe le plus fondamental de l'analyse multi-échelles [Guigues 2006]. D'après ce principe, pour tout couple d'échelles $\lambda_2 > \lambda_1$, les "structures" trouvées à l'échelle λ_2 devraient trouver une "cause" à l'échelle λ_1. En effet, une région d'un certain niveau dans l'arbre appartient à une branche de l'arbre qui correspond à un ensemble allant d'une région très fine jusqu'à l'image entière. Ainsi, tous les espaces des formes peuvent être considérés comme des représentations multi-échelles. Par ailleurs, contrairement aux espaces d'échelles linéaires, le contour d'une forme donnée (composante connexe) correspond au contour réel de l'image sans "flou".

E Morphologie basée sur les formes

Un espace des formes basé sur l'arbre $S_{\mathcal{T}}$ est un graphe connexe qui est équivalent à l'espace de l'image. Chaque nœud dans l'espace de l'image correspond à un point pondéré par une fonction d'intensité f. La connexité de l'espace de l'image est généralement la 4-connexité (resp. 6-connexité) ou la 8-connexité (resp. 26-connexité) dans les images 2D (resp. images 3D). Par contraste, chaque nœud dans l'espace des formes est une composante connexe pondérée par une fonction d'attribut $\mathcal{A}$. La connexité entre les nœuds de l'espace des formes est donnée par le lien de parenté dans l'arbre des composantes.

Un filtrage très simple consiste à enlever les composantes connexes dont l'attribut est trop faible. Mais il existe de nombreux filtres beaucoup plus sophistiqués. Nous proposons d'appliquer les opérateurs connexes dans les espaces des formes. Nous appelons ce processus *morphologie basée sur les formes*; il est illustré en Figure 1 en ajoutant le chemin rouge. Ainsi, le filtrage se fait non pas dans l'espace de l'image, mais dans l'espace des formes construits sur l'image. En procédant de cette manière, nous généralisons les filtres connexes existants. Nous pouvons utiliser ce cadre non
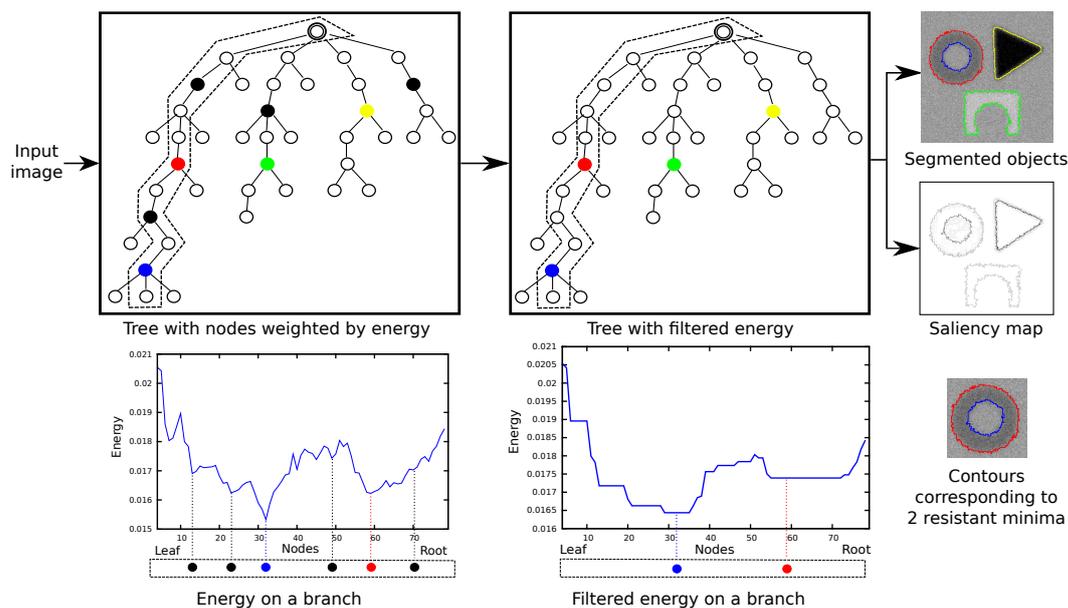
Figure 5: Un exemple de détection d'objets. Dans les arbres, les cercles pleins représentent les minima locaux, et les cercles pleins colorisés, les minima locaux restant après filtrage. En bas, les courbes montrent l'évolution de l'attribut, avant et après filtrage, le long de la branche entourée dans l'arbre. En haut à droite : les objets significatifs détectés (contours colorisés) et une hiérarchie de détection d'objets représentée sous la forme d'une carte de saillance.

seulement pour le filtrage, mais aussi pour la détection/segmentation d'objets et la segmentation hiérarchique.

Filtrage. Le cas classique est celui où l'attribut $\mathcal{A}$ est croissant. Dans ce cas, le Min-tree $\mathcal{TT}$ est isomorphe au premier arbre $\mathcal{T}$. Suivre le chemin rouge est équivalent à suivre le chemin noir. Ceci montre que notre proposition englobe le cas classique. Cependant, un attribut $\mathcal{A}$ décrivant les formes est le plus souvent non-croissant. Dans ce cas là, $\mathcal{TT}$ est différent de $\mathcal{T}$. Si le second attribut est identique au premier attribut, alors l'élagage de TT est équivalent au seuillage de T.

Dans le cas général, le second attribut $\mathcal{AA}$ est différent de $\mathcal{A}$. Cela nous permet d'introduire deux nouvelles familles d'opérateurs connexes. Quand l'espace des formes est basé sur un Max-tree (resp. Min-tree), nous appelons ce type de filtres des nivellements supérieurs (resp. inférieurs) basés sur les formes. Quand l'espace des formes est donné par une représentation arborescente auto-duale, plus particulièrement un arbre des formes, nous appelons "shapings" cette famille de filtres.

Détection/segmentation d'objets. Les espaces des formes fournissent un espace réduit de recherche. Si on veut détecter un seul objet dans l'image, on peut simplement repérer le nœud ayant l'attribut le plus significatif ; il correspond à la forme de l'objet le plus probable. Mais dans le cas général, le nombre d'objets à détecter dans l'image est inconnu. Nous proposons alors d'identifier ces objets comme étant ceux qui correspondent aux minima locaux de l'espace des formes. Cependant, ces minima sont souvent trop nombreux. Pour résoudre ce problème, nous proposons d'appliquer un filtre connexe dans l'espace des formes, ce qui va éliminer les minima non significatifs. Un exemple est donné en Figure 5.

Segmentation hiérarchique. En augmentant la force du filtrage dans la méthode de détection d'objets, de plus en plus minima locaux vont disparaître ou être absorbés par des minima plus significatifs. Cette force de filtrage peut être mesurée par une valeur d'extinction [Vachier 1995] définie sur l'ensemble de minima locaux. Les valeurs d'extinction mesurent les persistances des minima donc des objets. En faisant varier le seuil sur la valeur d'extinction, nous hiérarchisons les minima locaux. Cette hiérarchie peut être utilisée pour simplifier (enlever les objets les moins persistants), ou pour segmenter (garder des objets persistants). Une telle hiérarchie peut se représenter sous la forme d'une image de saillance des contours [Najman 1996, Najman 2011, Guigues 2006] : le contour de chaque minimum peut être pondéré par la valeur d'extinction pour laquelle il disparaît dans la hiérarchie. On obtient ainsi une image de contours dans laquelle les objets les plus significatifs ont les contours les plus brillants.

F Quelques illustrations du cadre

Dans cette section, nous présentons quelques illustrations et applications de notre cadre : en détection d'objets (Section F.1) ; en étendant la notion de connectivité contrainte (Section F.2) ; en filtrages (Section F.3).

F.1 Segmentation d'objets

Nous avons appliqué la méthode de segmentation d'objet à la segmentation de nerfs optiques dans des image de fond d'oeil. Sur la base de données DRI-ONS [Carmona 2008], nous obtenons des résultats à l'état de l'art. Nous avons aussi introduit dans [Xu 2012] un critère original pour la détection d'objets. Ce critère est fondé sur une énergie de type "contours actifs". Une application à la détection d'objets en utilisant ce critère est illustré en Figure 6.

Figure 6: Un résultat de détection d'objets.

F.2 Extension de la connectivité contrainte

Nous avons appliqué notre cadre pour étendre la connectivité con-
trainte [Soille 2008]. Il s'agit ici de simplifier une hiérarchie de zones plates par un
critère croissant. Nous proposons de remplacer l'attribut croissant par un attribut
non croissant inspiré de [Felzenszwalb 2004] et d'utiliser notre cadre pour produire
une segmentation hiérarchique. La Figure 7 montre un exemple d'application. Les
résultats des tests dans la base de données de BSDS500 [Arbelaez 2011] sont donnés
dans la Table 1.

Figure 7: Exemple de segmentation hiérarchique. En haut : l'image originale et la
carte de saillance obtenue ; en bas : deux segmentations extraites de la hiérarchie.

Method	GT Covering			PRI	
	ODS	OIS	Best	ODS	OIS
FH [Felzenszwalb 2004]	0.43	0.53	0.68	0.76	0.79
Guimarães [Guimarães 2012]	0.46	0.53	0.60	0.76	0.81
Ours	**0.50**	**0.57**	**0.66**	**0.77**	**0.82**

Table 1: Evaluation de notre segmentation hiérarchique.

F.3 Illustration des nouveaux filtres

La Figure 8 montre une comparaison entre la méthode de seuillage utilisant l'attribut de circularité, un "shaping" avec le même attribut et un autre "shaping" reposant sur une combinaison de la circularité et du moment d'inertie divisé par l'aire au carré. Dans la Figure 9, nous montrons des exemples de "shaping" reposant sur un critère issu de l'énergie de Mumford-Shah. L'espace des formes utilisé est ici l'arbre des formes [Monasse 2000b]. Cette méthode est décrite dans [Xu 2013]. Nous avons appliqué le nivellement supérieur à base des formes à la segmentation de vaisseaux dans les images rétiniennes. Nous avons obtenu des résultats au niveau de l'état de l'art avec ce simple filtre.

G Conclusion

Dans cet thèse, nous avons introduit la notion d'espaces des formes basés sur les arbres. Une première illustration de l'intérêt des structures arborescentes est la conception d'un détecteur de zone d'intérêt, vraiment invariant aux changements de contraste. Ce détecteur obtient des résultats à l'état de l'art en recalage d'images et en reconstruction 3D à base d'images. Notre principale proposition est le cadre que nous appelons morphologie basée sur les formes. L'idée de base est d'appliquer des filtres connexes dans l'espaces des formes au lieu de l'espace de l'image. Ce cadre nous permet d'introduire deux nouveaux types d'opérateurs connexes : des nivellements supérieurs/inférieurs basés sur les formes et les "shapings". Nous avons montré que ce cadre peut être utilisé pour la détection et la segmentation d'objets. Par ailleurs, ce cadre nous permet d'obtenir des segmentations hiérarchiques.

(a) Image d'entrée.

(b) "Shaping" 1.

(c) Seuillage faible.

(d) Seuillage fort.

(e) Combinaison de seuillage.
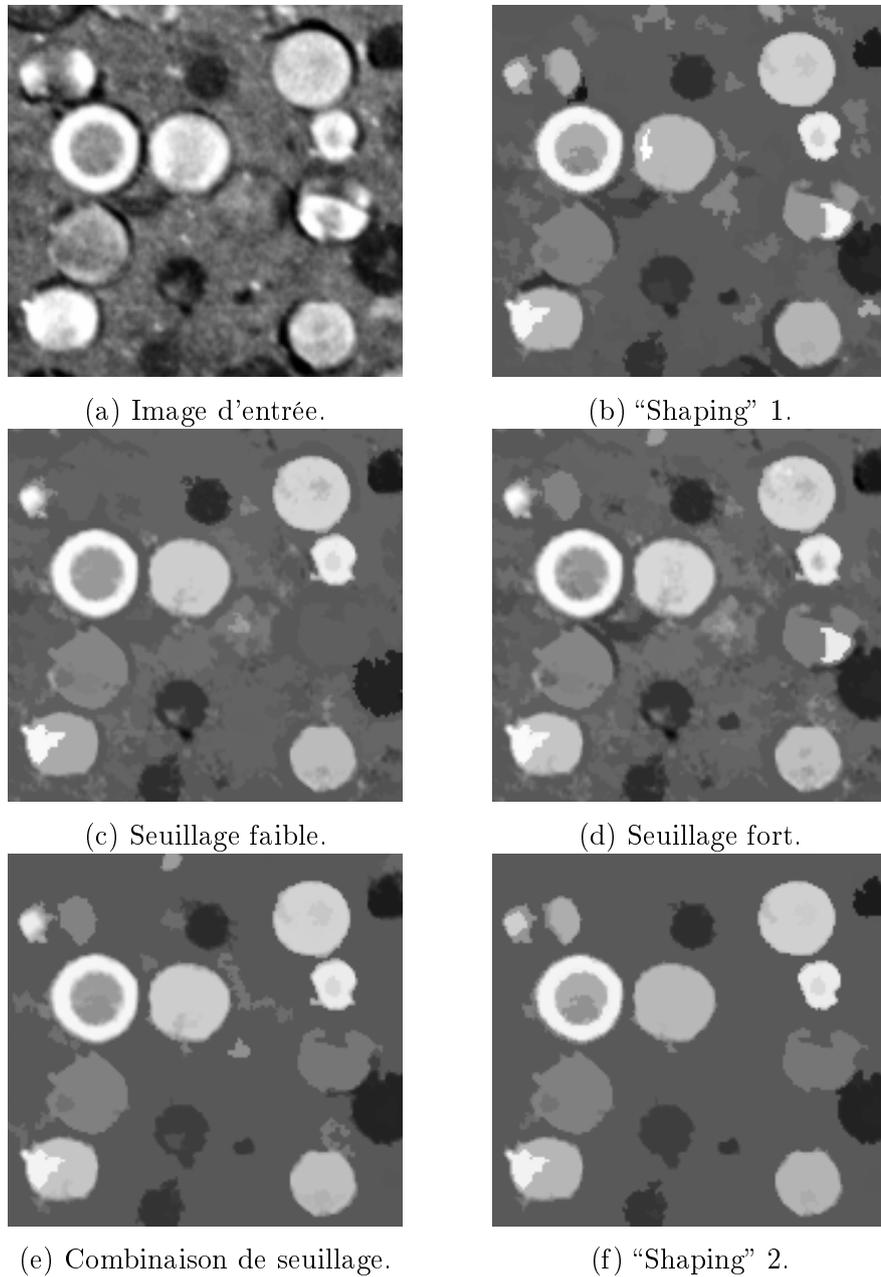

(f) "Shaping" 2.

Figure 8: Comparaison entre le "shaping" et un seuillage dans l'espace des formes : en (b,c,d), le critère est un attribut de forme ; en (e,f), l'attribut est une combinaison de plusieurs attributs de forme.

Figure 9: Deux résultats de pré-segmentations obtenues par "shaping" utilisant un critère fondé sur l'énergie de Mumford-Shah ; l'espace des formes est ici créé à partir de l'arbre des formes.

Contents

Part I

Introduction

Introduction

An image to be processed is usually modeled as a discrete function defined on pixels or voxels over a 2D or 3D domain. There exist also many other image representations which decompose the image into primitive or fundamental elements, so that it can be more easily interpreted. Indeed, an increasing number of applications in image processing and computer vision are not efficiently handled when using the classical pixel-based processing approaches. Content-based image compression or indexing, as well as many biomedical and remote sensing applications are such instances. In these applications, low-level processing would benefit from region-based representations, super-pixels for instance. In this PhD thesis, we are interested in a particular type of region-based representations: the tree-based image representations. Such representations have been popularized by connected filters [Serra 1993, Salembier 1995, Salembier 1998, Salembier 2009]. Besides, there are many applications in image processing and computer vision relying on this kind of image representation. This PhD thesis proposes a general framework to analyze tree-based image representations, so as to process and analyze images. This framework offers several processing aspects, ranging from filtering to object detection/segmentation, and hierarchy transformation. Those aspects lead to several applications developed in the work presented in this thesis. In this introductory chapter, we will first shortly review two types of tree-based image representations in Section 1.1. (the detailed introduction to tree-based image representations will be found later in the first chapter of Part II). The context of this thesis is described in Section 1.2. Our main contributions are summarized in Section 1.3. Finally, Section 1.4 presents the overall structure of this thesis report.

1.1 Image representations

A classical image model is a function defined on pixels of an image domain. It can be seen as a node weighted graph. The set of vertices of this graph is the set of pixels, and the edges of the graph encode the neighborhood relationship between pixels. Many applications in image processing and computer vision interact with some primitives of fundamental elements being more meaningful than the pixels.

There are several image representations specific to some applications, such as the Fourier transform or the wavelet transform. The former is often used in applications such as geometric analysis, image denoising, image deblurring, or registration, and the later is usually used for image compression and texture analysis. The image representations which we are interested in are the region-based representations. They are composed of a set of regions of the original image. These regions are either disjoint or are organized thanks to an inclusion relationship. These image representations can be divided into two categories: hierarchical representations and threshold decompositions.

1.1.1 Hierarchical representations

The first type of the region-based representations is the hierarchy of segmentation, known also as pyramids [Pavlidis 1979, Rosenfeld 1984, Jolion 1994]. A hierarchy of segmentation is composed of a set of partitions going from fine to coarse. It can be represented by a tree structure, whose root node represents the entire image as a single region, and whose leaves correspond to the regions of the finest image partition. The other nodes lying between the root and the leaves represent the regions obtained by the fusion of all the regions represented by its children. The hierarchy of segmentation is a multi-resolution/multi-scale image representation. However, unlike many scale-space based representations, all the regions encoded in this hierarchy are actually "present" in the original image. In fact, the hierarchy of segmentation provides a tremendously reduced space of candidate regions to be processed for those applications that interact with objects, or areas of interest.

A first example of this type of image representation is the quadtree, popularized in the early 1970's [Finkel 1974]. A quadtree is created in a top-down way by recursively subdividing the regions of the image into quadrants or regions. It has been shown to be useful in many applications such as image compression or image segmentation. Another popular strategy to create a hierarchy of segmentation is bottom-up. some noticeable examples are the minimum spanning tree (MST) [Kruskal 1956], the α-tree [Ouzounis 2011a, Nagao 1979, Soille 2008, Meyer 2000], and the binary partition tree (BPT) [Salembier 2000]. Those hierarchies are widely used in image simplification/segmentation, and connected filtering. A hierarchy of segmentation is usually represented through a special type of tree called dendrogram. It can be efficiently represented by a saliency map [Najman 1996], which has been popularized by Arbeláez under the name of "ultrametric contour map" [Arbeláez 2006a, Arbelaez 2011]. A powerful tool to compute a saliency map, so does a hierarchy of segmentation, is the ultrametric watershed [Najman 2009]. Actually, it has been shown in [Najman 2011] that any hierarchical image segmen-

tation is equivalent to an ultrametric watershed. Details about hierarchical representations will be reviewed in Section 2.3.

1.1.2 Threshold decompositions

The second type of region-based representations is based on threshold decompositions [Beucher 1992], as developed in mathematical morphology [Serra 1982, Serra 1988, Soille 2003, Najman 2010]. Unlike the hierarchy of segmentation, which depends usually on image contrast, the image representations based on threshold decompositions is contrast-invariant. It relies only on the pixel-value ordering, and, for any increasing contrast changes, the order between pixels remains the same.

One of the simplest representations given by threshold decompositions is based on the upper level sets' $\{x \,|\, f(x) \geq \lambda\}$ decomposition. The upper level sets are non-increasing with respect to the threshold values λ. This property allows to embed the set of upper level sets (with decreasing threshold values λ) into a tree structure called Max-tree [Salembier 1998]. The root of this tree denotes the entire image domain, and the leaves are the local regional maxima of the image. By duality, the Min-tree is defined as the tree representation based on the lower level sets $\{x \,|\, f(x) \leq \lambda\}$; the leaves of the Min-tree are the local regional minima in the image. Note that the upper/lower level sets of an image f are the same as the ones of the image f' obtained by any increasing contrast change applied to f. Any contrast-invariant processing applied to an image can be interpreted as a geometric processing acting on those level sets and preserving their order. Eventually, the output image is reconstructed from these processed level sets.

The Max-tree and Min-tree representations consider respectively the local maxima and minima in a different way. However, the objects of interest in an image can be bright, or dark. Many applications need to rely on the self-dual property: the bright and dark components are expected to be processed in the same manner. Another tree representation based on the threshold decompositions, called the topographic map [Caselles 1999] or the tree of shapes [Monasse 2000b], fulfills this condition. The topographic map is obtained using the inclusion relationship of the *shapes*, where a shape is defined as the connected component of upper or lower level sets with filled holes. The topographic map is a self-dual, and non-redundant image representation, and it is invariant to increasing contrast changes.

All those tree-based threshold decompositions are contrast-invariant. They are also multi-scale representations composed of a set of included or disjoint regions going from small ones to large ones. These trees have been popularized by connected filters, *i.e.*, filters that act by merging of flat zones. They are also proved to be useful in many applications, such as local feature detection, image indexing, scenery images

analysis, and classification of images. More details about these three morphological trees and some application examples will be presented respectively in Section 2.3 and in Section 2.6.

1.2   Context

In many low-level processing tasks such as image denoising and image filters, the classical pixel-based image representation is used. For example, in the case of linear filters, and classical morphological operators such as dilation and erosion using structuring elements, the processing strategy is to modify the values of each pixel based on a function defined on a local window around this pixel. These tools are not able to deal with the notion of regions.

In a large number of applications, the processing relies on objects or areas of interests, therefore the pixel-based image representation is not well adapted. For instance, in content-based image compression or indexing, one may want to selectively encode areas of interest or act on some objects. In biomedical image analysis and remote sensing, one is usually interested in some type of meaningful regions, such as organs of interest or cells in biomedical images, and buildings or areas of trees in remote sensing images. All those applications would benefit from a region-based processing.

Early examples of region-based processing can be found in the field of image segmentation, such as the classical split-and-merge methods [Rosenfeld 1984]. Recently, in mathematical morphology, the connected operators [Salembier 1998, Salembier 2009] have received much attention. They are region-based filtering tools that act by merging flat zones (connected components having constant values). They have good contour preservation properties in the sense that they do not create any new boundaries, neither do they shift the existing ones. Since the introduction of connected operators, the related literature grows rapidly with theoretical studies, algorithm developments, and many applications.

One popular implementation for connected operators relies on tree-based image representations, notably threshold decomposition representations and hierarchical representations. Tree-based connected operators consist in constructing a set of nested or disjoint connected components, whose filtering is based on an attribute function that characterizes each connected component. Most of the filtering strategies are based on decisions made due to an individual analysis of these connected components. Finally, the filtered image is reconstructed from the simplified tree composed of the remaining connected components.

The tree-based image representations, especially the three morphological trees

popularized by the connected operators have also been proved to be very useful for many applications. For instance, the Max-tree and Min-tree representations have been used for image simplification and segmentation [Ballester 2007], visualization of images [Wilkinson 2001, Westenberg 2007], local feature detection by the method of maximally stable extremal regions (MSER) [Matas 2002], and classification of images [Urbach 2007]. The topographic map has been used for meaningful level lines selection [Cao 2005], preferential image segmentation [Pan 2009b], scenery image analysis [Song 2002, Song 2003], and texture indexing [Xia 2010]. The success of the morphological trees in these applications is due to the inherent multi-scale and contrast-invariant properties of these trees. Besides, these trees can be computed efficiently with existing algorithms. Another tree-based image representation popularized by the connected operators is the binary partition tree (BPT) [Salembier 2000]. This has been demonstrated to be very useful for object detection/segmentation [Vilaplana 2008] due to the fact that the BPT is a multi-scale representation which provides a tremendously reduced search space.

Region-based representations are intensively employed in image segmentation. According to Marr's computational theory of vision [Marr 1983], a number of image analysis systems are composed of two stages: low level analysis (characteristic points, contours, regions ...) and high level vision tasks (object recognition, scene interpretation ...). It is also assumed that the low-level and high-level stages are completely independent. Since the structures of interest, which might be useful for the high-level tasks, can be located in any arbitrary position in the image. They can have any size and any level of contrast. So the low-level process should be uncommitted to these conditions, and output a multi-scale and contrast-invariant general description that is independent from any specific high-level task. So, in the field of image segmentation, one popular strategy is to compute a hierarchy of segmentation, from fine ones to coarse ones [Guigues 2006], instead of computing a single partition with respect to a certain scale. Some examples are the quadtree [Finkel 1974], the minimum spanning tree [Kruskal 1956], the α-tree [Ouzounis 2011a, Nagao 1979, Soille 2008, Meyer 2000], the binary partition tree [Salembier 2000], the ultrametric watershed [Najman 2009], and the ultrametric contour map [Arbelaez 2011]. The hierarchy of segmentation has also a tree structure. This type of region-based representations has been shown to be useful in many applications, such as image simplification/segmentation [Soille 2008], image compression [Samet 1985, Markas 1992], object detection [Vilaplana 2008], and scene labeling by learning hierarchical features [Farabet 2013].

All those region-based representations, popularized by their use in connected filtering and in segmentation, have several properties in common. All of them can

be embedded into a tree structure whose root represents the entire image as a single region. The leaves correspond to small regions. These tree structures are multi-scale image representations in the sense that a parent region is always larger than the children regions. They obey the most fundamental principle of multi-scale analysis, the causality principle [Koenderink 1984]. In fact, a region R at a certain scale is included in all its ancestor nodes till the root node. This region R can be seen as a kind of consequence for its ancestor nodes. Besides, some of these tree-based structures are contrast-invariant image representations. All these features contribute to the success of many applications using these tree-based image representations.

In most of the applications using tree-based image representations, the analysis of the tree is performed individually, which is to say that the regions represented by the tree nodes are considered as completely independent. The decision of filtering a non-relevant region or selecting a region of interest is only based on the attribute describing the underlying region. The structure of the tree, *i.e.* the inclusion relationship between regions is often ignored, which is unfortunate since this relationship contains important contextual information. That kind of information is usually more appropriate than the one defined by a local window in the classical case. Most of the tree analysis strategies fall usually into two categories: pruning-based methods and thresholding-based methods. There are very few applications relying on the tree-based image representations that make use of the tree structures, such as the maximally stable extremal regions (MSER) [Matas 2002], the preferential image segmentation [Pan 2009b], and the texture image indexing [Xia 2010]. However, only some simple information about the tree structure is used. For instance, the comparison of attributes describing a region with the ones describing its parent region and the children region included in it for local minima detection in the tree, or the number of children.

In this PhD thesis, we propose a more general method to deal with region-based image representations. We first introduce the notion of tree-based shape space given by the tree-based image representations. Most of the existing applications relying on those representations can then be seen as a simple analysis (sometimes individual analysis) of this shape space. A first consequence of this interpolation through the notion of the shape space is a novel local feature detection method based on topological selection of regions from the shape space, (it can be seen as a variant of the widely used MSER). The core work presented in this thesis is a framework that we call shape-based morphology, inspired from the connected filters, but applied to the tree-based shape space instead of the image space. It provides a more general and robust way to analyze the tree than the existing strategies. It is a versatile framework, which can be easily adapted to many image processing,

pattern recognition, and computer vision problems.

1.3 Main contributions

The main contribution of this thesis is the proposition of a framework relying on region-based image representations, which are used in an increasing number of applications. We call this framework *shape-based morphology*. It is inspired by the connected operators [Salembier 1998, Salembier 2009]. We first introduce the notion of tree-based shape space which can be defined by any tree-based image representation. It is a new point of view that considers any tree-base image representation as a node-weighted graph, the weights being a certain attribute function describing each region. The main idea of the framework of shape-based morphology is to apply connected operators to this later graph. It is a versatile framework that can be used for many image processing, pattern recognition, and computer vision problems. A number of applications using the framework of shape-based morphology have been proposed, which demonstrate the usefulness and show the high potential of this framework. Finally, the algorithms used in this thesis are also presented to ensure *reproducible research*.

1.3.1 A versatile framework

The notion of tree-based shape space constructed from any tree-based image representation provides a simple point of view for many applications using tree representations. They can be seen as non-relevant points filtering or a selection of points of interest in the shape space. However, in most of these applications, this shape space analysis is performed individually, or very simple information about the tree structure is used. Such information can be for instance the number of children for each node, or the local minima detection based on the comparison between the attribute of a node and the ones of its parent node and children nodes. The framework of shape-based morphology makes use of the complete tree structure, that is mainly the inclusion relationship between neighboring regions. This information can be seen as the context of each node, which is moreover region form adapted, unlike the local window. This "context" might provide a more adequate information that helps deciding whether or not to filter out a component. Consequently, this framework is more general than the classical existing ones and might also be more robust. Depending on how we process this shape space, this framework can be used for three different purposes.

- **Filtering:** This framework includes the classical pruning- or threshold-based

strategies. It also yields introduction of the two novel classes of connected operators called the shape-based lower/upper levelings, and self-dual morphological shapings.

- **Object detection/segmentation:** Supposing that the main structures of the objects of interest are present in the shape space (which is usually a reasonable hypothesis), we can apply this framework for object detection/segmentation. To do this, we can spot the local minima or maxima of a certain attribute function describing each region of the shape space. Connected operators in the shape space help to remove the spurious minima or maxima, so does the meaningless objects. Besides, a saliency map [Najman 1996] representing a soft object detection can be obtained by using the notion of extinction values [Vachier 1995].

- **Hierarchy transformation:** Employing the principle of the soft object detection, we can produce with our framework a saliency map representing a hierarchical image segmentation from any tree-based image representation.

1.3.2 A number of applications

We have developed a number of applications using the framework of shape-based morphology, which demonstrate the usefulness and show the high potential of the framework.

- **Tree-based Morse regions (TBMR):** The TBMR in Chapter 4 is a direct result from the fact that MSER [Matas 2002] can be seen as a relevant selection of points from the shape space. The TBMR is a variant of MSER, whose selection is based on topological information inspired from the Morse theory [Milnor 1963]. Unlike MSER, the TBMR is truly invariant to contrast changes. Experimental results show that it achieves a comparable repeatability score, but extracts a significantly higher number of features compared with the state-of-the-art methods. The applications to image registration and 3D reconstruction demonstrate its accuracy and robustness.

- **Hierarchical image simplification:** The simplification method presented in Chapter 5 is an efficient morphological shaping. It quickly minimizes the piecewise-constant Mumford-Shah functional subordinated to the topographic map. A hierarchical version is also obtained using by the aspect of hierarchy transformation of the framework. Experimental results show the usefulness of this fast simplification method.

- **Retinal image analysis:** We have applied the shape-based filtering of our framework to retinal image analysis, including blood vessel segmentation and optic nerve head (ONH) segmentation. Quantitative benchmarks in Chapter 6 demonstrate that some simple filtering achieves state-of-the-art results, when compared with more evolved methods.

- **Object detection/segmentation:** We have presented in Chapter 7 a novel efficient ratio-cut estimator, which is context-based and which can be interpreted as an active contour. Some examples of the application of this estimator to the topographic map show the usefulness and robustness of this proposed estimator and the object detection/segmentation aspect of the framework.

- **Extending constrained connectivity:** We have applied in Chapter 8 the hierarchy transformation aspect of the framework to the α-tree, (known also as the hierarchy of constrained connectivity), with an non-increasing attribute function inspired from the work of [Felzenszwalb 2004]. Quantitative benchmarks on the BSDS500 dataset show that this saliency map-based hierarchy might represent better contents of input images.

1.3.3   Reproducible algorithms

We present the efficient algorithm to compute the morphological trees based on the union-find algorithm, and the algorithm to incrementally compute attributes (relying on accumulated information on region, contour, and context). We have also developed a fast algorithm to compute minimal information along the contours for some attribute such as the number of false alarms [Desolneux 2001, Cao 2005]. Finally, we have also introduced an efficient algorithm for disjoint level lines selection from the topographic map representation, which provides a simplified tree that helps to improve the visualization of the topographic map.

1.4   Manuscript contents

This thesis is divided into three main parts.

The **first part** presents the main concept of the work presented in this thesis. It is composed of two chapters.

- **Chapter 2: Reviewing tree-based connected operators.** This chapter is an introduction to the background of our work. We review the connected operators, which constitues the main context of our PhD work. Many region-based image representations that can be organized into a tree structure are

detailed. We then give some application examples that rely on these trees to demonstrate their usefulness. The novelty of this chapter is that we introduce the notion of the tree-based shape space, which is the basis of our proposed framework. Eventually, we provide a simple understanding about some existing applications in the context of the shape-space.

- **Chapter 3: Shape-based morphology framework.** This key chapter presents the core proposition of this thesis: the framework of *shape-based morphology*. The basic idea of this framework is to apply connected operators over the tree-based shape space. It consists of two tree constructions: one is constructed from the image, and the second one is constructed from the first tree representation. Our framework provides a simple definition of MSER [Matas 2002], we can easily obtain an extension of MSER. Besides, this framework has three main consequences. 1) For filtering purposes, the classical existing connected operators are generalized, and two novel types of filters are introduced. They are shape-based lower/upper levelings and self-dual morphological shapings. 2) This framework can be used for object detection/segmentation by selecting relevant points from the shape space. 3) This framework provides a method to transform any tree representation into a hierarchy representing a hierarchical image simplification/segmentation.

The **second part** of this thesis presents different applications that we have developed using the framework of shape-based morphology.

- **Chapter 4: Tree-Based Morse Regions (TBMR).** The first application is based on a simple analysis of the tree-based shape space. We present a variant of the maximally stable extremal regions (MSER) that we call tree-based Morse regions (TBMR). The main idea is to select in a covariant way the regions from the tree-based shape space defined by the Min-tree and Max-tree representations. The selection is based on some topological information inspired from the Morse theory. It is truly contrast-invariant and quasi parameter-free, as compared to the MSER approach. Besides, TBMR extraction features the same complexity as MSER. Experimentally, TBMR achieves a repeatability on a par with state-of-the-art methods, but obtains a significantly higher number of features. The applications of TBMR to image registration and 3D reconstruction demonstrate its accuracy and robustness.

- **Chapter 5: Hierarchical image simplification.** In this chapter, we first propose an efficient self-dual morphological shaping, that very quickly leads

to a locally optimal solution for the piecewise-constant Mumford-Shah functional minimization being subordinated to the topographic map. It selects a set of salient level lines, and yields a simplified image from the remaining level lines. This is achieved by constructing only one tree representation. Using the same principle of this example of shaping, as well as the aspect of hierarchy transformation in the framework of shape-based morphology, we obtain a saliency map representing a hierarchical image simplification. Experimental results demonstrate the efficiency, usefulness, and robustness of our method, when applied to image simplification, color image pre-segmentation, and autophagosome counting in cellular images.

- **Chapter 6: Shape filtering.** We show several applications using the filtering aspect of the framework of shape-based morphology. These shape-based filters introduced in this framework are applied to retinal image analysis, including blood vessel segmentation and optic nerve head (ONH) segmentation. Quantitative evaluations demonstrate that some simple shape-based filters, as compared to more evolved processings, can achieve state-of-the-art results. Besides, we also illustrate an example among many variant of morphological shapings.

- **Chapter 7: Object segmentation on the shape spaces.** We present an application of the object detection/segmentation aspect of our framework. In this chapter, we first introduce a novel efficient ratio-cut estimator, which is context-based and which can be interpreted as an active contour. It is used as the attribute function that describes each region in the shape space. A first example is applied to the topographic map [Monasse 2000b]. The estimator can be computed incrementally on this shape space. Experimental results in synthetic and real images demonstrate the robustness and usefulness of the proposed context-based energy estimator. They also show that the shape-based morphology is a versatile framework which is well-suited for object segmentation tasks.

- **Chapter 8: Extending constrained connectivity.** This chapter focuses on the aspect of hierarchy transformation of our framework. As a first example, we use the shape space given by the α-tree (known also as the hierarchy of constrained connectivity). This application can be seen as an extension of constrained connectivity [Soille 2008] by transforming the α-tree into another hierarchy of segmentation represented by a saliency map. Experiments on the BSDS500 dataset show that this later saliency map might represent better the contents of input images. This demonstrates the usefulness of the hierarchy

transformation aspect in the framework of shape-based morphology.

The **third part** presents those algorithms used in this thesis.

- **Chapter 9: Algorithms to compute information on the tree.** In this chapter, we detail the algorithms we have used for reproducible research. To implement them, we use our C++ image processing library Olena [Levillain 2010]. We first show the algorithm for tree construction based on the union-find algorithm. Then we show how to efficiently compute the attributes relying on accumulating information on region, contour, and context. The computation of minimal information along the contours for some attribute such as the number of false alarms [Desolneux 2001, Cao 2005] is then illustrated. Finally, we present also an algorithm of disjoint level lines selection relying on the topographic map representation. It provides a simplified tree that helps to improve the visualization of the topographic map.

Chapter 10 concludes the thesis by summarizing the framework of shape-based morphology, and the applications developed in this framework. We also present some possible improvements and future research on these applications. Several important factors that ensure the performance of our framework are then discussed. We give some suggestions for additional studies of the three aspects of the framework. Finally, we consider some possible applications of this framework.

Part II

Concept

Reviewing tree-based connected operators

This chapter presents the background of the core concept in this thesis. We review the connected operators in Section 2.2, which are the main context of our PhD work. Many region-based image representations that can be organized into a tree structure are then detailed in Section 2.3. One popular implementation of the connected operators is based on those tree-based image representations; it is detailed in Section 2.5. The novelty of this chapter is the introduction of the notion of *tree-based shape space* in Section 2.4. This shape space is the basis of our proposed framework, and it provides a simple understanding about many existing applications (see Section 2.6) relying on the use of tree-based image representations.

2.1 Preliminaries

An image can be seen as an undirected graph represented by a pair $G = (V, E)$, where V is the finite set of vertices and E is the set of edges. Each vertex $v \in V$ represents a pixel or a voxel of the image domain, and each edge $e \in E \subseteq V \times V$ models the neighborhood relationship (classically, 4 or 8-connectivity for 2D images, and 6 or 26-connectivity in 3D cases) between the two vertices composing e.

A graph (V, E) is said to be *connected* if, for any $x, y \in V$, there exists a path from x to y, which is a sequence of $n > 1$ vertices $(x_0 = x, x_1, \ldots, x_n = y)$ such that every $x_i \in V$, and every $(x_i, x_{i+1}) \in E$. Usually, the image domain is connected.

A binary image X is a subset of the image domain that induces a subgraph (V_X, E_X), such that V_X is the set of vertices representing the set of points of X, and $E_X = V_X \times V_X \cap E$. The binary set X is said to be connected if the subgraph (V_X, E_X) is connected. A *connected component* C of X is a connected subset of X with the maximal extent. This means for any C' such that $C \subseteq C' \subseteq X$, if C' is connected, then we have $C' = C$. More details about the notion of connectivity can be found in [Serra 1998, Braga-Neto 2003, Ouzounis 2007, Ouzounis 2011b, Serra 2012a].

(a) Original binary image   (b) Output binary image
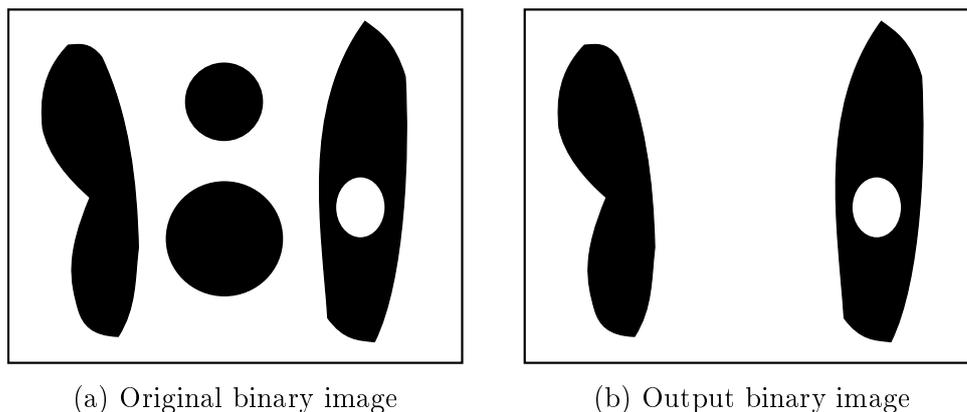
Figure 2.1: An example of binary connected operator.

In the following, X and Y denote the binary sets, the grayscale image $f \colon V \to \mathbb{R}$ or $\mathbb{Z}$ is a mapping $\mathcal{F}$ from V to $\mathbb{R}$ or $\mathbb{Z}$, which assigns a gray level to each vertex. It is a node weighted graph denoted as (G, F), where F is an element of $\mathcal{F}$. As a convention, operators on binary images are denoted with capital Greek letter; the corresponding lowercase letter is used to denote the grayscale version.

2.2   Connected operators

2.2.1   General definition

Let us start with the notion of connected operators [Serra 1993, Salembier 1995, Salembier 1998, Salembier 2009] for sets and then extend it to the case of grayscale images.

Definition 1 *An operator Ψ working on an arbitrary binary image X is said to be connected when the set of difference $X \setminus \Psi(X)$ is exclusively composed of connected components of X or of its complement X^c.*

This means that the connected operators for sets acts only by preserving or removing the connected components of foreground and of background. An example of the binary connected operator is given in Figure 2.1, where the round foreground objects are removed, the others are intact.

The extension of connected operators to grayscale images relies on the notion of *partition* of *flat zones* [Salembier 1995]. A flat zone $F_h(f)$ of a grayscale image f is a connected component of the level set

$$F_h(f) = \{x \in V \mid f(x) = h\} \tag{2.1}$$

Note that there may be multiple disjoint flat zones for each gray level h.

A partition of an image is a set of disjoint, non-void connected components $\{C_i\}$ such that

1 $\forall i, C_i \neq \emptyset$,

2 $\forall i \neq j, C_i \cap C_j = \emptyset$,

3 $\cup C_i = V$.

It is obvious that the set of all the flat zones of a gray level image f forms a partition of the image domain. Now let us denote $\mathcal{P}$ as the partition and $\mathcal{P}(x)$ as the region of $\mathcal{P}$ that contains x. A partition $\mathcal{P}_1$ is said to be finer than (written as $\mathcal{P}_1 \sqsubseteq \mathcal{P}_2$) a partition $\mathcal{P}_2$, if $\forall x, \mathcal{P}_1(x) \subseteq \mathcal{P}_2(x)$. That means if $\mathcal{P}_1$ is finer than $\mathcal{P}_2$, then any pair of points belonging to the same region of $\mathcal{P}_1$ belongs to a unique region of $\mathcal{P}_2$.

Definition 2 *An operator ψ working on any grayscale image f is connected if the partition of flat zones of $\psi(f)$ is always coarser than the partition of flat zones of the input image f.*

The definition 2 shows that the grayscale connected operators acts by merging flat zones. Consequently, the regions of the output partition of flat zones are created by union of regions of the input partition. They do not introduce any new contour, and keep perfectly the location and shape of the contours, which make the connected operators well known for the good contour preservation properties. An example of connected operator compared to the linear filter and the classical morphological operator based on structural element is shown in Figure 2.2

The connected operators are usually considered as filtering tools in the sense that they transform an input grayscale image into a filtered grayscale image. And as the conception of grayscale connected operators relies on the notion of partition, They are also claimed to bridge the gap of classical filtering and segmentation [Jones 1999, Gatica-Perez 2001]. Indeed, some theoretical notions about the extension of connected operators to pure segmentation applications are presented in [Serra 2006, Ronse 2008], known as connective segmentation. Some application examples of the connected operators to segmentation will also be shown in Chapter 7 and Chapter 8.

One of the most successful implementation of such connected operators is based on a reconstruction process. The readers are referred to [Salembier 2009] for more details. A special case called leveling will be reviewed in Section 2.2.3. Another popular implementation relies on the tree-based image representations (detailed in Section 2.3). See Section 2.5 for details about this efficient implementation.

(a) Input image
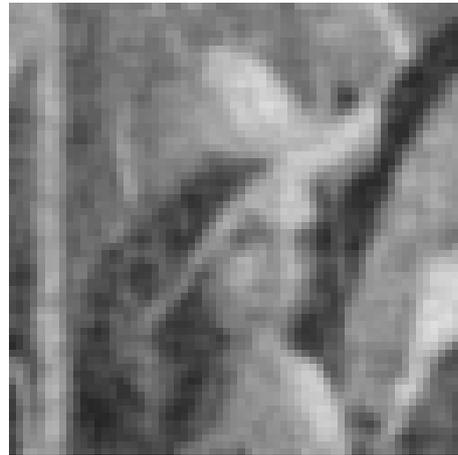
(b) Gaussian filter
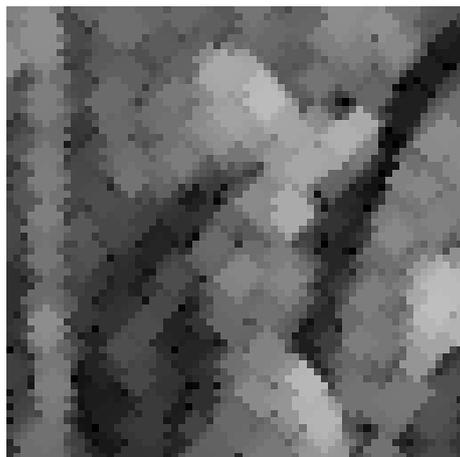
(c) Opening with disk

(d) Connected operator

Figure 2.2: A connected operator example compared with a linear filter, and a classical morphological opening with structuring element. The Gaussian filter in (b) blurs image, and the opening using a disk as structuring element in (c) creates new contours. The grain filter [Caselles 2002] does not create any new contour, neither shifts the existing contours.

2.2.2 Attribute filters

The attribute filters [Breen 1996, Salembier 1998, Heijmans 1999] interact with connected components instead of individual points in the case of classical morphological operators originally developed by Matheron and Serra [Serra 1982, Serra 1988]. They are connected operators that act by preserving or by removing the connected components based on some attribute criterion. A subset of these filters, called *shape filters* [Urbach 2007], has been used for extraction of vessels in 3D angiograms [Wilkinson 2001].

Before detailing the attribute filters, let us review several properties usually discussed for morphological operators Ψ working on binary images.

- An operator Ψ is **increasing** if $X \subseteq Y \Rightarrow \Psi(X) \subseteq \Psi(Y)$;

- An operator Ψ is **extensive** if $X \subseteq \Psi(X)$;

- An operator Ψ is **anti-extensive** if $\Psi(X) \subseteq X$;

- An operator Ψ is **idempotent** if $\Psi(\Psi(X)) = \Psi(X)$;

- An operator Ψ is **Self-dual** if $\Psi(X) = \Psi(X^c)^c$.

For the morphological operators ψ working on grayscale images, the corresponding properties is defined by replacing $\subseteq$ with $\leq$, and by replacing the complement operator ^{c} by $-$. A grayscale connected operator ψ is said to be self-dual if it process symmetrically the dark and bright image components, $\forall f, \psi(f) = -\psi(-f)$. A morphological filter is an increasing and idempotent operator [Heijmans 1999]. However, sometimes this condition is relaxed to mean any idempotent operator, as in the case of shape filters [Urbach 2007].

Let T be an attribute criterion : $\mathcal{P}(V) \to \{\text{true, false}\}$, where $\mathcal{P}(V)$ is the set of all subsets of V. Typically, T is an assessment of a connected component C based on the comparison of its attribute (some interesting feature) $\mathcal{A}$: $\mathcal{P}(V) \to \mathbb{R}$ to a given threshold λ

$$T(C) = (\mathcal{A}(C) \geq \lambda). \tag{2.2}$$

Then the *trivial* attribute filter Γ_T on a connected component C returns the connected component C itself if $T(C)$ is true, and $\emptyset$ otherwise, with also $\Gamma_T(\emptyset) = \emptyset$. And the binary *connected opening* Γ_x of X at point $x \in V$ gives the connected component of X containing x if $x \in X$, and $\emptyset$ otherwise.

Definition 3 *The anti-extensive binary attribute filter Γ^T working on a binary image X is given by*

$$\Gamma^T(X) = \bigcup_{x \in X} \Gamma_T(\Gamma_x(X)). \tag{2.3}$$

If $\mathcal{A}$ is increasing, which means $X \subseteq Y \Rightarrow \mathcal{A}(X) \leq \mathcal{A}(Y)$, Γ^T is then an attribute opening, otherwise, Γ^T is an attribute thinning [Breen 1996]. A binary attribute thinning example is shown in Figure 2.1, the attribute is the roundness which is a non-increasing attribute.

The extension of the attribute filters to grayscale images is based on image thresholding decomposition. For a given grayscale image f, thresholding f in a decreasing order from h_{max} to h_{min} yields a stack of nesting upper level sets. Each upper level set at level h is a binary image given by

$$\mathcal{X}_h(f) = \{x \in V \mid f(x) \geq h\}. \tag{2.4}$$

Each binary image $\mathcal{X}_h(f)$ contains a set of connected components known also as peak components P_h^i [Salembier 1998]. A peak component P_h^x with level $h \leq f(x)$ containing x is defined as below

$$P_h^x(f) = \Gamma_x(\mathcal{X}_h(f)). \tag{2.5}$$

Note that for any two peak components $P_{h_1}^i$ and $P_{h_2}^j$ at respectively level $h_1 \leq h_2$, either $P_{h_1}^i \cap P_{h_2}^j = \emptyset$, or $P_{h_2}^j \subseteq P_{h_1}^i$. This inclusion relationship yields a tree structure with the name of Max-tree [Salembier 1998]. See also Section 2.3 for more details.

The attribute filters for grayscale images consist of preserving or removing the peak components. For the increasing attribute $\mathcal{A}^{\uparrow}$, if P_h^x is preserved, for any $h' \leq h$, $P_{h'}^x$ is also preserved.

Definition 4 *The grayscale anti-extensive attribute filter γ^T for a given image f at some point x is defined by*

$$(\gamma^T(f))(x) = \vee \{h \mid x \in \Gamma^T(\mathcal{X}_h(f))\}. \tag{2.6}$$

For the non-increasing attribute $\mathcal{A}$, different *filtering rules* (see Section 2.5.2 for more details) are defined.

The extensive attribute filters for binary images and grayscale images can be easily defined by the duality relationship with the anti-extensive versions defined above. An example of self-dual grayscale attribute filter is given in Figure 2.2(d), the attribute is the area. This filter is also known as the grain filter and is detailed in Section 2.6.1.1.

2.2.3 Leveling: an example of connected operator

The levelings is a subclass of connected operators, it was first introduced in [Meyer 1998] by imposing some constraints in the definition of connected operators. The levelings enlarges the flat zones by suppressing many details while keep perfectly the sharpness of the transitions zones preserved. Usually, the levelings are considered as the intersection of two subclasses the lower levelings and upper levelings.

Definition 5 *An operator ψ is an lower leveling working on a grayscale image f if and only if for any pair of neighboring points (x, y) : $\psi(f)(x) > \psi(f)(y) \Rightarrow \psi(f)(y) \geq f(y)$.*

Definition 6 *An operator ψ is an upper leveling working on a grayscale image f if and only if for any pair of neighboring points (x, y) : $\psi(f)(x) > \psi(f)(y) \Rightarrow \psi(f)(x) \leq f(x)$.*

Let us remark that the lower leveling (resp. upper leveling) removes the details of the regional minima (resp. maxima), so it enlarges the dark (resp. bright) flat zones.

Definition 7 *An operator ψ working on a grayscale image f is a leveling if and only if for any pair of neighboring points (x, y) : $\psi(f)(x) > \psi(f)(y) \Rightarrow f(x) \geq \psi(f)(x)$ and $\psi(f)(y) \geq f(y)$.*

The definition 7 states that if there is a transition in the output image after leveling, the transition exists in the initial image. Because $\psi(f)(x) > \psi(f)(y) \Rightarrow f(x) \geq \psi(f)(x) > \psi(f)(y) \geq f(y)$. Furthermore, the interval of the transition in the output image $[\psi(f)(y), \psi(f)(x)]$ is contained in the interval of the transition in the input image $[f(y), f(x)]$.

An example of the leveling with the use of markers is illustrated in Figure 2.3. The leveling function g is obtained by increasing (resp. decreasing) the marker function h as little as possible until a flat zone is created or the function g hits the function f on $\{h < f\}$ (resp. on $\{h > f\}$).

The readers are referred to [Meyer 1998, Meyer 2004] for more details about the properties of levelings. In particular, it has been shown that the opening and closing by reconstruction [Vincent 1993] are levelings.

2.3 Tree-based image representations

This section briefly review two types of image representations that can both be organized into a tree structure. The first one is based on image threshold decompositions,
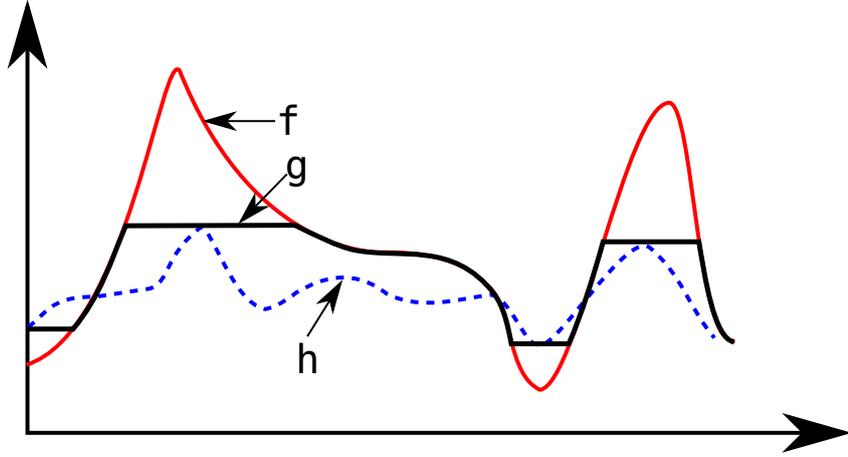
Figure 2.3: An illustration of leveling. f = reference function; h = marker function; g = associated leveling.

including the Min/Max-tree [Salembier 1998] and topographic maps [Caselles 1999], known also as the tree of *shapes* [Monasse 2000b]. The second type of tree-based image representation is the hierarchy which is composed of a set of segmentations from fine to coarse, including the Binary Partition Tree (BPT) [Salembier 2000], α-tree [Ouzounis 2011a] based on the notion of constrained connectivity [Soille 2008], the quadtree [Finkel 1974, Pietikainen 1981, Rosenfeld 1983, Samet 1984] (but not used in any application presented in this thesis), and the minimum spanning tree [Kruskal 1956, Morris 1986]. All these trees belong to the more general family of hierarchical image segmentations. It has been shown in [Najman 2011] that any hierarchical image segmentation is equivalent to a ultrametric watershed [Najman 2009], known also as the saliency map [Najman 1996].

2.3.1 Min/Max-tree

One of the simplest tree-based image representations is the Max-tree [Salembier 1998]. It is based on the inclusion relationship between peak components defined by the upper level sets in Eq. (2.4). Each tree node $\mathcal{N}_k$ with level h represents a peak component P_h^i such that $P_h^i \cap L_h \neq \emptyset$. Note that in practice, only the vertices of P_h^i having level h denoted as $C_h^i = \{x \in P_h^i \mid f(x) = h\}$ are stored in the node $\mathcal{N}_k$. So the peak component P_h^i is actually given by the set of vertices stored in all the nodes of the subtree rooted at $\mathcal{N}_k$. The links between nodes known as parenthood reflects the inclusion relationship. The highest node of the tree is the *root*, which is the whole image domain. And the leaves of the Max-tree correspond to the regional maxima of the image. A simple example of is shown in Figure 4.2,
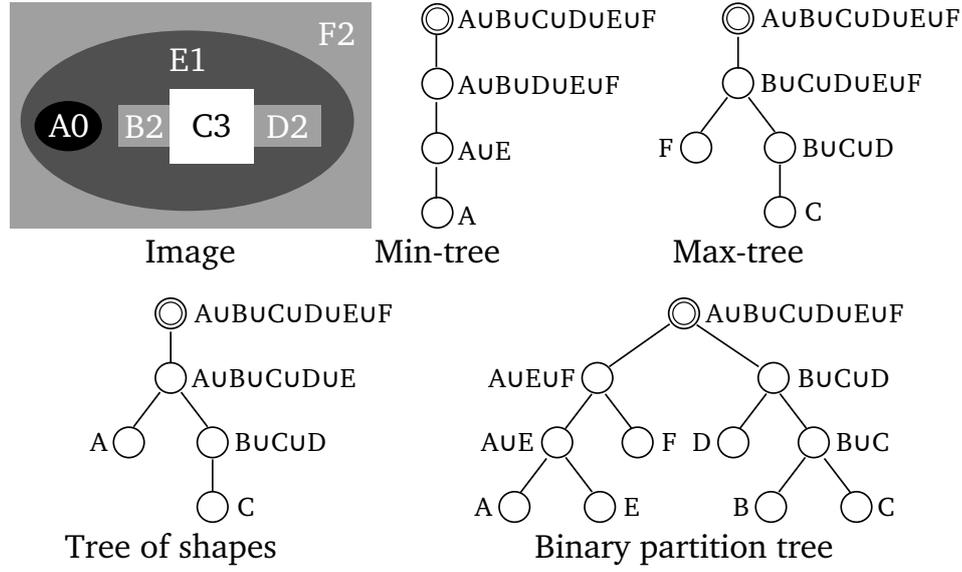
Figure 2.4: Tree-based image representations. The root is at the top represented by double circle. The numbers in the original image denote the pixel values.

where the leaves C and F are two regional maxima.

By duality, the dual structure of the Max-tree named Min-tree is based on the lower level sets decomposition defined as below

$$\mathcal{X}^h(f) = \{x \in V \,|\, f(x) \le h\}. \tag{2.7}$$

The inclusion relationship between connected components of the lower level sets yields the structure of Min-tree. The leaves of the Min-tree are the regional minima of the image (See Figure 4.2). Note also that both the data of the family of upper level sets in Eq. (2.4) and lower level sets in Eq. (2.7) permit us to reconstruct the image [Caselles 1999, Monasse 2000b, Caselles 2009].

$$f(x) = \sup\{h \in \mathbb{R} \,|\, x \in \mathcal{X}_h\} = \inf\{h \in \mathbb{R} \,|\, x \in \mathcal{X}^h\}. \tag{2.8}$$

Note that the Max-tree (resp. Min-tree) is very appropriate for implementation of anti-extensive (resp. extensive) connected operators, since the pruning of the Max-tree (resp. Min-tree) enlarges the flat zones of regional maxima (resp. minima).

Various algorithms have been proposed to compute efficiently the Max-tree and Min-tree. They can be classified into two categories. The first one is based on the flooding procedure [Salembier 1998, Nistér 2008] which generally starts at the root, and performs a depth-first or breadth-first flooding process to build the final tree. And the second one [Najman 2006, Berger 2007] is a union-find [Tarjan 1975]

based approach. It consists usually of two passes. A first pass sorts the pixels, and the second pass, in reverse order, build the tree while performing the union-find process. This type of algorithm has a quasi-linear time complexity when pixel values have a low quantization. A detailed comparison of these algorithms can be found in [Carlinet 2013].

2.3.2   Topographic map

The Max-tree and Min-tree permit us to efficiently handle the bright and dark objects, but in a asymmetrical way. Several authors then proposed to consider the level lines (topological boundaries of the connected components given by the upper or lower level sets) instead of upper or lower level sets. This leads to a unique tree representation describing the image. In the literature, this is tree is called topographic map [Caselles 1999], inclusion tree or tree of *shapes* [Monasse 2000b], monotonic tree or [Song 2003] or level line tree [Song 2007].

More specifically, according to the proposition of Monasse and Guichard [Monasse 2000b], this tree is obtained by merging the Max-tree and Min-tree through the concept of *shapes*, a *shape* is defined as the connected components of upper or lower level sets with the holes filled. It has been shown that these *shapes* can be structured into a tree representation. Note that the concept of *shape* here is a special case of the notion of shape for the tree-based shape spaces which will be defined in Section 2.4. In what follows, we use the term of *topographic map* for this tree-based image representation.

The topographic map along with the Max/Min-tree feature several interesting properties. First of all, they are invariant to local contrast changes. And they inherently embed a morphological scale-space (the parent of a node is a larger connected component). Moreover, the topographic map is a non-redundant and complete image representation. Another important property of the topographic map is its self-duality, which make the bright object on dark background represented in the same manner as the dark object on bright background. This implies that the bright objects and dark objects are treated in the same way. The leaves of the topographic map can be either regional maxima or minima. However, note that not all the regional maxima and minima are represented by the leaves of the topographic map. An example is illustrated in Figure 2.4. The regional minima A and regional maxima C correspond to the leaves, whereas the regional maxima F is not a leaf.

The first efficient algorithm to compute the topographic map is the fast level lines transform (FLLT) proposed by Monasse and Guichard in [Monasse 2000b]. It takes a region-growing approach to build the Max-tree and the Min-tree, then finds for each hole in a connected component the connected component in the other tree

corresponding to it, and puts that connected component as descendant of the one containing this hole. An improved version, fast level set transform (FLLT) is then introduced in [Monasse 2000a]. Both FLLT and FLST have an average complexity of $O(N \log N)$, where N is number of pixels in image. In [Song 2007], Song proposed a top-down approach to compute the topographic map. It begins with the image boundary, and for each level line under scrutiny, find all its child level lines. The algorithm computes level lines directly instead of level set components. It takes a complexity of $O(N + t)$, where t is the total length of all level lines. Recently, Géraud *et al.* proposed a quasi-linear algorithm to compute the topographic map for nD images [Géraud 2013]. It shares the same process as algorithms in [Najman 2006, Berger 2007] based on union-find process. This algorithm will be detailed in Section 9.1. Besides, an algorithm of disjoint level lines selection facilitating the visualization of the corresponding topographic map will also be presented in Section 9.5.

2.3.3 Binary Partition Tree (BPT)

The Max/Min-tree and the topographic map are extremum oriented image representations. They describe the image as a set of connected components starting from the extrema. These connected components are either disjoint or included in another. Nevertheless, the real objects in the scene may not coincide with nodes of those trees. On the contrary, the Binary Partition Tree (BPT), first introduced by Salembier [Salembier 2000] reflects the similarity between neighboring regions. It represents a set of regions obtained from an initial partition. The leaves of BPT represent the regions of that initial partition, and the remaining tree nodes represent regions that are obtained by merging regions represented by children. The root of the tree represents the entire image domain. The BPT consists of a set of regions of different scales. This is why it can be viewed as a hierarchical region-based image representation. Large regions are represented by nodes close to the root whereas small details can be found at lower levels. This representation is considered as a compromise between representation accuracy and the processing efficiency. In fact, the BPT does not include all possible merging of regions of the initial partition seen as a region adjacency graph (RAG) , only the most "likely" or "useful" mergings are represented in the BPT. The connectivity encoded in the BPT is binary in the sense that each region is explicitly connected to its sibling which forms their parent. Consequently, the BPT only encodes part of the neighborhood relationship of the initial RAG. However, this representation allows the fast implementation of some sophisticated processing techniques.

More specifically, this tree should be build in such a way that the most "interest-

ing" regions are represented. This might be application dependent, but a possible way which is suitable for a large number of applications is to create the tree by tracking the merging steps performed by a region merging based segmentation algorithm (See for example [Morris 1986, Garrido 1998]). Starting from a given initial partition (that can be the partition of flat zones or even each pixel is a region), the merging algorithm proceeds iteratively by 1) computing the similarity measure between all pair of neighboring regions, 2) selecting the most similar neighboring regions and merging them into a new region, 3) updating the neighborhood and the similarity measures. The algorithms iterate the steps 2) and 3) until a single region covering the entire image domain has been formed. Two important aspects of the merging steps are the merging order and the region model. The readers are referred to [Salembier 2000, Vilaplana 2008, Salembier 2009] for more details. An example of such a tree-based representation is given in the Figure 2.4.

2.3.4 α-tree

The α-tree [Ouzounis 2011a], known also as hierarchy of single linkage components [Nagao 1979], constrained connectivity [Soille 2008] or quasi-flat zones [Meyer 2000] is based on the notion of α-connectivity [Soille 2008]. For a pair of neighboring points x and y, let $d(x,y)$ be the dissimilarity measure between them, then two points p and q are said to be α-connected if there is a path $\{p \rightsquigarrow q\}$ which is a chain of pairwise adjacent points commonly given in the form of $\{p \rightsquigarrow q\} \equiv \langle p = p_1, \ldots, p_n = q \rangle$, such that for any pair of adjacent points (p_i, p_{i+1}), $d(p_i, p_{i+1}) \leq \alpha$ always holds. Based on the notion of α-connectivity, a α-connected component [Soille 2008] (known also as single linkage component [Nagao 1979] or quasi-flat zone [Meyer 2000]) α-CC(p) containing a point p is defined as

$$\alpha\text{-CC}(p) = \{p\} \cup \{q \mid p \text{ and } q \text{ are } \alpha\text{-connected}\}. \qquad (2.9)$$

Assuming that the dissimilarity measure between a pair of neighboring points x and y is the intensity difference $d(x,y) = |f(x) - f(y)|$, and setting $\alpha = 0$, the 0-connected component is actually the flat zone. Note that for two neighboring points p and q, $d(p,q) > \alpha$ does not imply that p and q do not belong to the same α-connected component but only that there is no direct linkage between them. Some examples of α-connected components are shown in Figure 2.5. Such defined α-connected components based on the local dissimilarity measure are known to suffer from leakage effects. One possible solution proposed in the literature to this problem, is the introduction of global constraints, for example, the range ω [Soille 2008], *i.e.*, the maximal dissimilarity between all the pairs of points within a given α-connected

(a) 0-CCs   (b) 1-CCs   (c) 2-CCs   (d) 3-CCs

(e) Partition of flat zones   (e) Associated α-tree
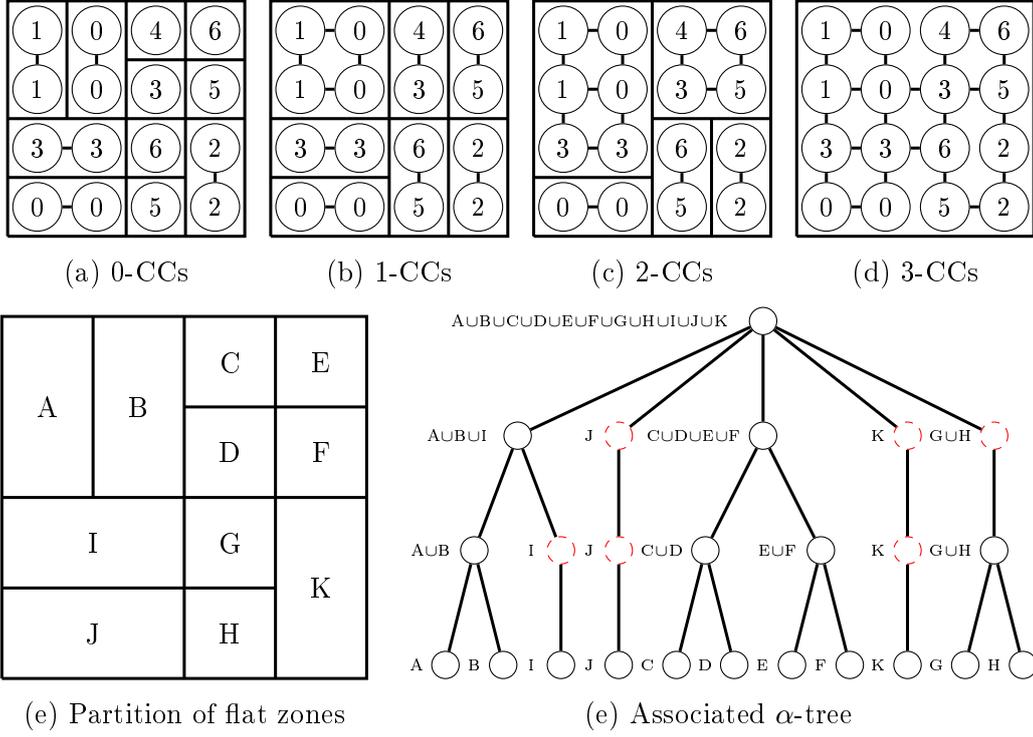
Figure 2.5: A 4×4 image and its partitions into α-connected components for α range from 0 to 3. (a) 0-CCs, (b) 1-CCs, (c) 2-CCs, (d) 3-CCs. Note the fine to coarse partition hierarchies. (e) The associated α-tree that encodes the fine to coarse partition hierarchies. The red dashed circles represent redundant nodes of the tree-based image representation.

component, which leads to the following definition of (α,ω)-CC(p):

$$(\alpha,\,\omega)\text{-CC}(p) = \bigvee \{\alpha_i\text{-CC}(p) \mid \alpha_i \leq \alpha, \text{R}\big(\alpha_i\text{-CC(p)}\big) \leq \omega\}, \qquad (2.10)$$

Where $\text{R}\big(\alpha\text{-CC(p)}\big)$ denotes the maximal dissimilarity within α-CC(p).

Note also that there exists an inclusion relationship between the α-connected components:

$$\forall\, x \in V, \text{ if } \alpha_1 \leq \alpha_2 \Rightarrow \alpha_1\text{-CC}(x) \subseteq \alpha_2\text{-CC}(x) \Rightarrow \mathcal{P}_{\alpha_1} \sqsubseteq \mathcal{P}_{\alpha_2}. \qquad (2.11)$$

Where $\mathcal{P}_\alpha$ is the partition of α-connected components. This inclusion relationship yields the structure of α-tree [Ouzounis 2011a], a hierarchical image representation of multi-scale partition. A simple example of α-tree is given in Figure 2.5. The interested readers are referred to [Soille 2008, Ouzounis 2011a] for more details about the α-tree. Some algorithms to compute this tree are available in [Najman 2011, Ouzounis 2011a, Najman 2013, Ouzounis 2012b].

2.3.5   Quadtree

The quadtree first proposed in the early 1970's [Finkel 1974] is a data structure in which each node has exactly four children or no children at all (a leaf node). For image processing applications, the quadtree [Rosenfeld 1983] is often used to decompose the image by recursively subdividing it into homogeneous quadrants or regions. The regions may be square or rectangular, or may have any arbitrary shapes. The quadtree has been demonstrated to be useful in applications such as image compression [Samet 1985, Markas 1992] or image segmentation [Pietikainen 1981, Spann 1985, Smith 1994, Pavlidis 1977].

More concretely, suppose that we have a given homogeneous criterion $\mathcal{C}$, (*e.g.*, the region range is under a given threshold value t). Then based on this homogeneous criterion, we can recursively subdivide a given image into homogeneous pieces. Let us also assume that each subdivision is into subquadrants. If the whole image to begin with is homogeneous under the criterion $\mathcal{C}$, we are done. Otherwise, split it into quadrants. And then examine the homogeneity for each of them, if a given quadrant is homogeneous, we are done for it, if not, split it into quadrants, and so on until all the quadrants are homogeneous under $\mathcal{C}$.

The result of the subdivision process can be represented by a tree structure. The root node of the tree represents the whole image, each internal node has exactly four children and represents a non homogeneous region to be subdivided into quadrants represented by the four children, and each leaf node represents a homogeneous block. An example of the quadtree is shown in Figure 2.6.

Note, however, the homogeneous blocks represented by the leaves are not necessary maximal homogeneous regions, as it is possible that the union of some adjacent blocks is still homogeneous. Consequently, for the image segmentation purpose, we must allow merging of some adjacent blocks as long as the union remains homogeneous. However, this "split-and-merge" process does not result a quadtree structure any more.

2.3.6   Minimum Spanning Tree (MST)

Given a image f which can also be seen as a nodes weighted graph (V, E), a spanning tree of that graph is a subgraph that is a tree connects all the vertices (*i.e.*, points) together. An image can have many spanning trees. An image can also be represented by an edge weighted graph, where each edge $e \in E$ is weighted by the dissimilarity measure between the two points linked by this edge e. Then use this to weight the spanning tree by computing the sum of the weights of the edges in that spanning tree. A minimum spanning tree (MST) [Kruskal 1956] is then a spanning tree having
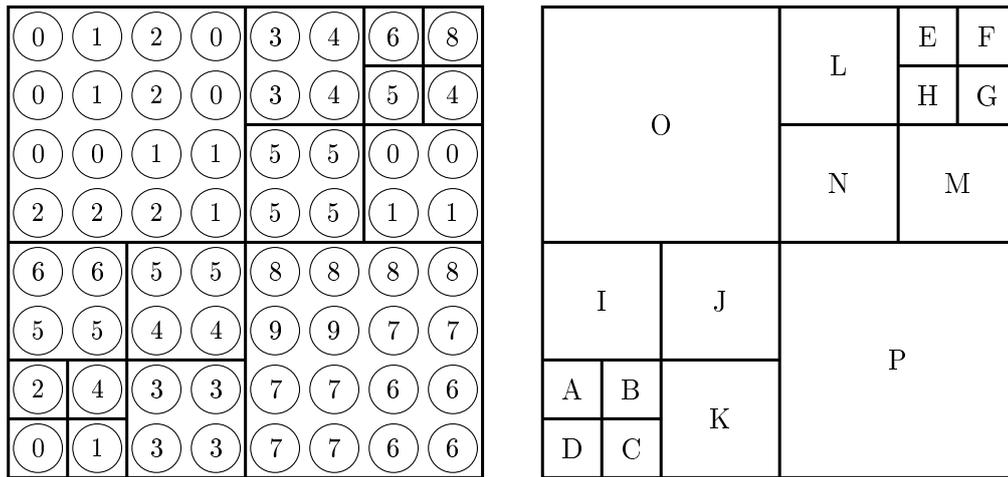
(a) A synthetic image

(b) Finest partition
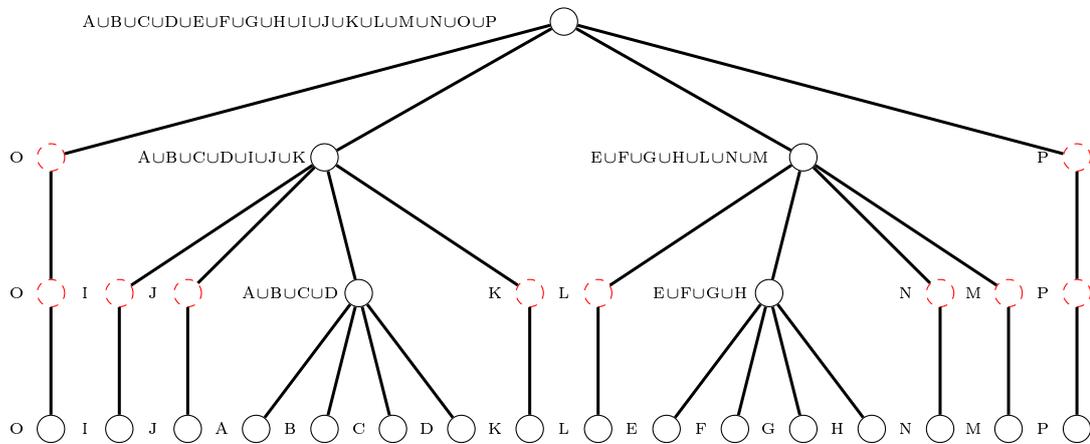
(c) Associated quadtree

Figure 2.6: An example of quadtree. The homogeneous criterion $\mathcal{C}$ is that the global range $\omega \leq 3$.

a weight less than or equal to the weight of any other spanning tree. Note that the minimum spanning tree may not be unique.

The minimum spanning tree is a typical and well-known problem of combinatorial optimization. It has been applied for many years to image analysis problems [Suk 1984, Morris 1986, Ma 2000, Felzenszwalb 2004, Cousty 2009]. Kruskal's algorithm [Kruskal 1956] is a well-known greedy algorithm that computes a MST for a connected edge weighted graph. It can be described as follows:

- Create a forest (a set of trees), where each vertex is a separate tree,

- Create a set S_E that contains all the edges in the graph, and sort this set S_E in weight increasing order,

- For each edge $e \in S_E$ in increasing order,

    - if e links two different trees, then add this edge e to the forest, and merging the two trees into a single one.

    - otherwise, do nothing.

At the termination of the algorithm, the forest has only one component and is a minimum spanning tree of the graph.

A Min-tree of the MST represents a set of partitions from fine to coarse, that can be organized into a tree structure. The root node represents the entire image as a single region. The regions correspond to the leaf nodes are the flat zones. The in-between nodes are obtained by incremental merging of regions. A simple example of the MST is depicted in Figure 2.7. An efficient algorithm based on Kruskal algorithm was presented in [Najman 2013], where the authors propose a quasi-linear algorithm that computes a so called *binary partition tree by altitude ordering*. A linear post-processing of this tree can give a Min-tree of the MST, which is proved in [Cousty 2013] to be equivalent to the α-tree.

2.3.7   Hierarchical image segmentations

The Binary Partition Tree (BPT) in Section 2.3.3, the α-tree in Section 2.3.4, the quadtree in Section 2.3.5, and the Minimum Spanning tree 2.3.6 exhibits several features in common. They are all multi-resolution/scale image representations that consist of a set of partitions from fine to coarse. The root represent the entire image as a single region, while the leaves correspond to the regions of the finest image partition (*e.g.*, partition of flat zones). In fact, they belong to a more general family, the hierarchical image segmentations, known also as the pyramids [Pavlidis 1979,
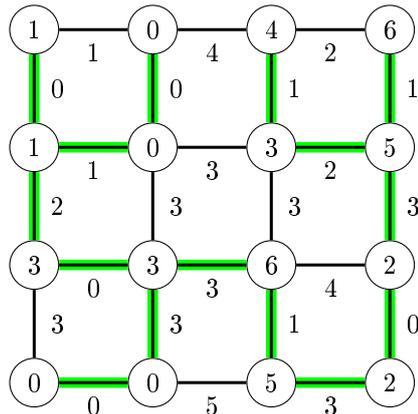
(a) An image and the MST
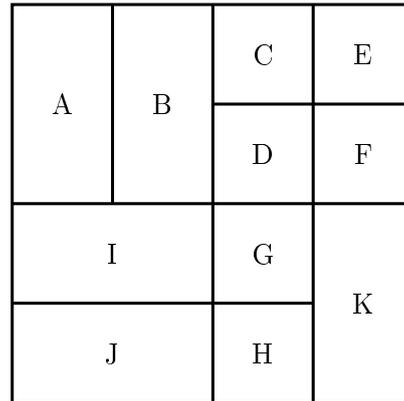
(b) Partition of flat zones
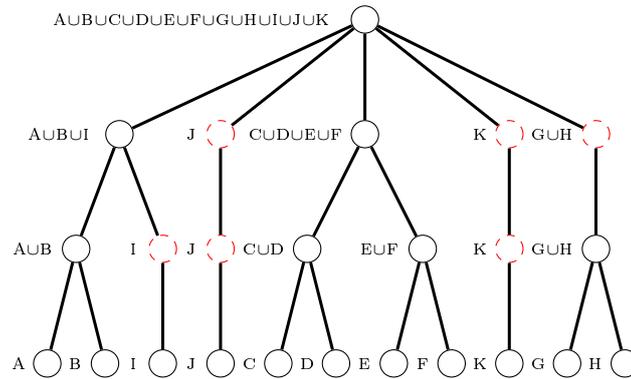
(c) Min-tree of the MST

Figure 2.7: An example of the Minimum Spanning Tree (MST). The edges in the MST of image (a) are highlighted in green. The Min-tree of the MST represents a set of partition from fine to coarse, which is equivalent to the α-tree.

Rosenfeld 1984, Jolion 1994]. A hierarchy of image segmentation H is a chain of nesting image partitions $\mathcal{P}_i$, *i.e.*,

$$H = \{\mathcal{P}_i \mid 0 \leq i \leq n, \forall j, k, 0 \leq j \leq k \leq n \Rightarrow \mathcal{P}_j \sqsubseteq \mathcal{P}_k\}, \qquad (2.12)$$

where $\mathcal{P}_n$ is the partition $\{V\}$ of V into a single region, and $\mathcal{P}_0$ represents the finest partition of the image $G = (V, E)$. In another word, a hierarchy H is a set of regions $\{R\}$, such that

(i) $\{V\} \in H$,

(ii) for each region $R \in \mathcal{P}_0$, $R \in H$,

(iii) for each pair of distinct regions (R, R'), where $R \in H, R' \in H, R \cup R' \neq \emptyset \Rightarrow R \subset R'$ or $R' \subset R$.

The (iii) means that two distinct regions in the hierarchy of segmentation are either disjoint or nested.

An *indexed hierarchy* on the image domain V is a pair (H, λ), where H denotes a given hierarchy on V and λ is a positive function (*e.g.*, the scale), defined on H such that for two nesting regions $R, R' \in H, R \subset R'$, we have $\lambda(R) < \lambda(R')$.

A hierarchy of image segmentation is usually represented using a special type of tree called *dendrogram*. The root node represents the entire image $\{V\}$, and the leaves are the regions of the finest partition $\mathcal{P}_0$, while an intermediary node $\mathcal{N}$ represents the merging of regions represented by the nodes just below node $\mathcal{N}$, known as the children of node $\mathcal{N}$. An example of such a hierarchical image segmentation represented by a dendrogram is shown in Figure 2.8 and Figure 2.9.

The most fundamental principle of hierarchical image segmentations being multi-scale representations is the causality principle [Koenderink 1984]. From this principle, for any couple of scales $\lambda_2 > \lambda_1$, the "structures" found at scale λ_2 should find a "cause" at scale λ_1. Following the original idea of Babaud [Babaud 1986], used also by Morel and Solimini [Morel 1995] and in the work of Guigues et al. [Guigues 2006], the causality principle is applied to the edges associated to the set of partitions represented by a hierarchy of segmentation H. In this case, the relationship "finer than" between the set of partitions spanned by a hierarchy H behaves as a scale parameter if and only if for all $\lambda_2 > \lambda_1$, which implies $\mathcal{P}_{\lambda_1} \sqsubset \mathcal{P}_{\lambda_2}$, the boundaries of partition $\mathcal{P}_{\lambda_2}$ are in a one-to-one mapping with a subset of the boundaries of $\mathcal{P}_{\lambda_1}$ (their "cause").

Instead of performing a single partition of the input image, such a hierarchical image segmentation H describes the image contents using multiple representations with decreasing resolution, so increasing scale. A best or optimal partition is then

(a) Original image   (b) A dendrogram of hierarchical segmentation
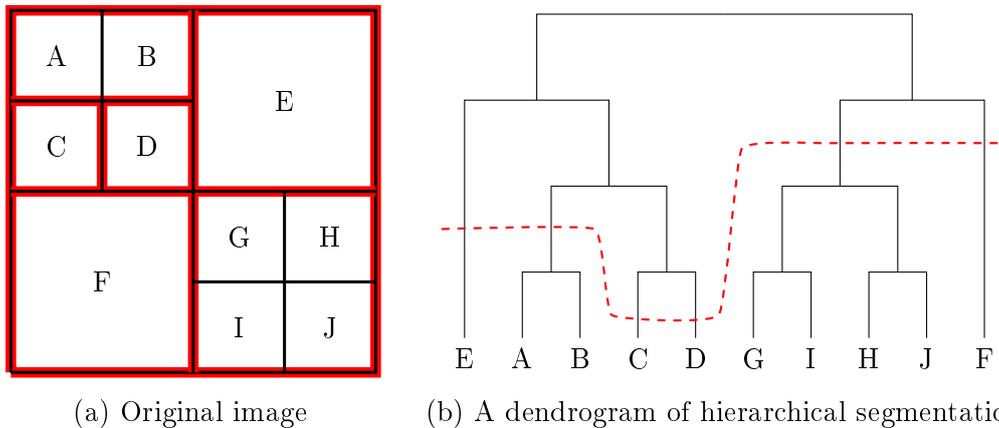
Figure 2.8: A synthetic image (a), and its associated dendrogram (b) representing a hierarchy of image segmentation. Note that there are possible partitions that are not spanned by the dendrogram. The red dashed curve in (b) is a cut of the dendrogram (*i.e.*, hierarchy), the partition given by this cut is illustrated with red boundaries in the original image (a).

selected from the set of partitions spanned by that hierarchy H based on some sophisticated criterion [Guigues 2006, Serra 2012b, Serra 2013, Cardelino 2013, Kiran 2013a, Kiran 2013b, Kiran 2014] through the notion of cuts [Guigues 2006]. A cut of a hierarchy H is a subset of H which intersects any path from the base to the top of H exactly once. Equivalently, a cut is a partition $\mathcal{P}$ of V whose regions are taken from the regions represented by nodes in H. The red dashed curve in Figure 2.8(b) represents a cut of the hierarchy, and its associated partition is given by the red boundaries in the original image in Figure 2.8(a). The Figure 2.9(c). represents a partition corresponding to a cut obtained from the hierarchy in Figure 2.9(b).

There exist many methods for computing a hierarchical image segmentation, which can be divided in two classes: bottom-up (*e.g.*, BPT, *alpha*-tree, MST), top-down or split-and-merge (*e.g.*, quadtree). Some recent reviews of those approaches can be found in [Marfil 2006, Soille 2008].

2.3.8 Saliency maps or ultrametric watersheds

A useful representation of hierarchical image segmentations reviewed in Section 2.3.7 was originally introduced in the PhD work of Najman [Najman 1994] under the name of *saliency map* [Najman 1996]. A saliency map is obtained by stacking a family of hierarchical contours. This representation is then rediscovered independently by Guigues et al. through the notion of scale-set theory for visualization purposes

[Guigues 2006], and it is then popularized by Arbeláez under the name of *ultrametric contour map* for boundary extraction and comparing hierarchies [Arbeláez 2006a, Arbelaez 2011].

Roughly speaking, for a given indexed hierarchy (H, λ), the corresponding saliency map can be obtained by valuating each point of the image domain $v \in V$ with the highest value λ such that it appears in the boundaries of some partition represented by the hierarchy H. Given a hierarchical image segmentation, it is easy to assign importance to contours, which defines a duality between closed, non-self intersecting weighted contours and that hierarchy. The low level (resp. upper level) of a hierarchy respects to weak (resp. strong) contours, and is thus an over-segmentation (resp. under-segmentation), which can be obtained by thresholding the saliency map with low (resp. high) value.

Recently, it is stated by Najman that any hierarchical image segmentation (*i.e.*, saliency map) is equivalent to a *ultrametric watershed* [Najman 2009, Najman 2011] in the framework of edge-weighted graphs (G, F_e), where F_e is a positive function weighting the edge $u = \{x, y\}$ between neighboring points x and y by a dissimilarity measure (*e.g.*, $F_e(u)$ denotes the absolute difference between intensity of point x and y). An ultrametric watershed is a *topological watershed* [Couprie 1997, Couprie 2005, Bertrand 2005] null on the minima applied on edge-weighted graph [Najman 2009]. The topological watershed was originally introduced by Couprie and Bertrand [Couprie 1997] on nodes weighted graph (G, F_n), and having a fundamental property, that preservers the contrast between the regional minima of F, where the contrast between two regional minima m_1 and m_2 is defined as the minimal altitude to which one must climb in order to go from m_1 to m_2, known also as the *connection value CV*. For two points $x \in V$ and $y \in V$, it is the number

$$CV(x, y) = \min\{h \,|\, x \in \mathcal{X}^h(V), y \in \mathcal{X}^h(V)\}. \tag{2.13}$$

In other words, the connection value between two point x and y is the altitude of the lowest lower level set that contains both x and y (rule of the least common ancestor). The connection value is very similar with the notion of *ultrametric distance*. Recall that a *distance* is a proper dissimilarity that obeys the triangular inequality: $d(x_1, x_2) \leq d(x_1, x_3) + d(x_2, x_3)$, where x_1, x_2, and x_3 are three points in the space. An ultrametric distance obeys the ultrametric inequality [Krasner 1944]: $ud(x_1, x_2) \leq \max\big(ud(x_1, x_3), ud(x_2, x_3)\big)$. It is stronger than the triangular inequality.

There exists general connection between indexed hierarchies and ultrametric distances which goes back to Benzécri [Benzécri 1973] and Johnson [Johnson 1967]. They proved that there is a bijection between them. In fact, the ultrametric distance

associated to a indexed hierarchy (H, λ) on V is the following:

$$ud(R_1, R_2) = \min\{\lambda(R') \mid R' \in H, R_1 \subseteq R', R_2 \subseteq R'\}. \qquad (2.14)$$

In other words, the ultrametric distance $ud(R_1, R_2)$ between two regions in V that are also presented in the Hierarchy H is given by the smallest region in H which contains both R_1 and R_2. Conversely, each ultrametric distance ud is associated to one and only one nodes in a indexed hierarchy. Taking this associated ultrametric distance as the positive function defined on H, the ultrametric watershed is easy to compute. There exists efficient and proven algorithms to compute the ultrametric watersheds (*i.e.*, saliency maps).

The Figure 2.9(d) depicts an ultrametric watershed corresponding to the hierarchical image segmentation represented in image in Figure 2.9(b). One segmentation in Figure 2.9(c) is obtained by thresholding the ultrametric watershed with certain value. With the equivalence between hierarchical image segmentation, saliency map, ultrametric watershed, the reading order for interpretation in Figure 2.9 can be (a), (d), (c), (b) instead of the classical reading order (a), (b), (c), (d). Each thresholding of the ultrametric watershed (*i.e.*, saliency map) yields a image segmentation of a certain resolution level.

2.4   Tree-based shape spaces

All the image representations reviewed in Section 2.3 can be divided into two classes: the thresholding decompositions based on trees $\mathcal{T}_t$ (*e.g.*, Min/Max-tree, topographic map), and the hierarchy of image segmentation based on trees $\mathcal{T}_h$ (*e.g.*, BPT, α-tree, quadtree, MST). A major difference between $\mathcal{T}_t$ and $\mathcal{T}_h$ is that any cut (except the root) of $\mathcal{T}_t$ yields a subset of the image domain, whereas any cut of a hierarchy of segmentation $\mathcal{T}_t$ gives a partition of the image domain. An example of several levels of the tree $\mathcal{T}_t$ and $\mathcal{T}_h$ are depicted respectively in Figure 2.10(a) and in Figure 2.10(b).

All these image representations are composed of a set of regions (*i.e.*, connected components) from small to large. These regions can be organized into a tree structure. Each region is represented by a node $\mathcal{N}$ in that tree. The root node of the tree represents the entire image domain, and for any two distinct regions R_1 and R_2 represented by two nodes $\mathcal{N}_1$ and $\mathcal{N}_2$ in the tree, we have a nesting property:

$$R_1 \cap R_2 \neq \emptyset \Rightarrow R_1 \subset R_2 \text{ or } R_2 \subset R_1. \qquad (2.15)$$

Except the root node, each node $\mathcal{N}$ in the tree has a unique parent node $\mathcal{N}_p$, and there exists an edge $(\mathcal{N}, \mathcal{N}_p)$ linking them reflecting the parenthood relation-

(a)

(b)

(c)

(d)
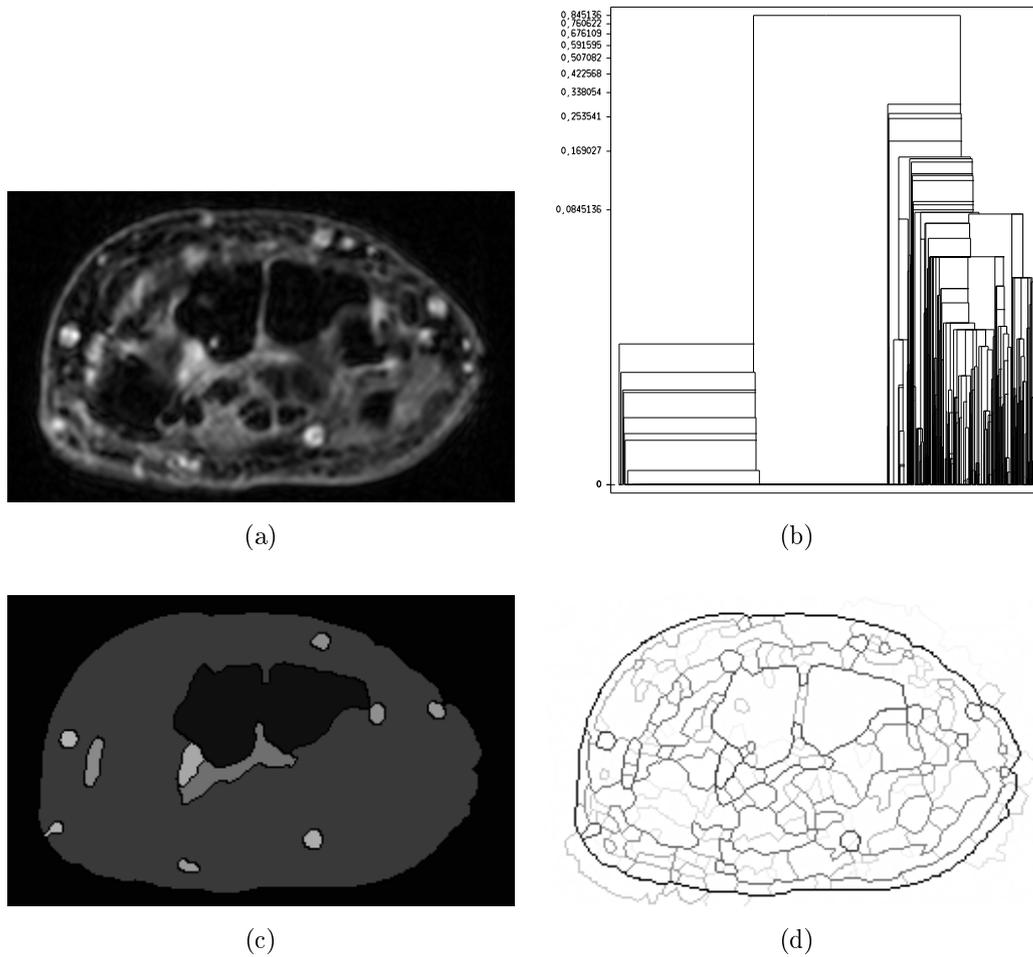
Figure 2.9: An example of a hierarchical segmentation produced by the method of Guigues *et al.* [Guigues 2006]. (a) Original image; (b) Dendrogram of the hierarchical segmentation; (c) One segmentation (*i.e.*, a cut) extracted from the hierarchy; (d) An ultrametric watershed corresponding to the hierarchical segmentation. The classical order for reading the images is (a), (b), (c), (d). But based on the notion of saliency map, the reading order can also be (a), (d), (c), (b) [Najman 2011].
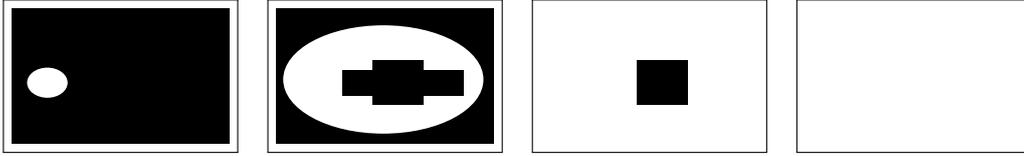
(a) Several levels of a thresholding decomposition based tree (*e.g.*, Min-tree).
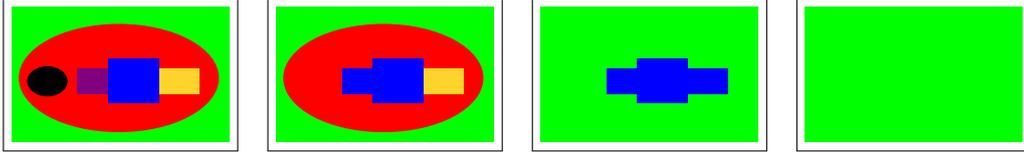
(b) Several levels of a hierarchy of image segmentation (*e.g.*, binary partition tree).

Figure 2.10: An example of the two classes of trees for the image shown in Figure 2.4.

ship, *i.e.*, the inclusion relationship between the two regions they represent. Consequently, the tree-based image representations can be seen as connected graphs $G_\mathcal{T} = (\mathcal{T}_\mathcal{N}, \mathcal{T}_e)$, where $\mathcal{T}_\mathcal{N} = \{\mathcal{N} \,|\, \mathcal{N} \in \mathcal{T}\}$ denotes the set of nodes in the tree $\mathcal{T}$, and $\mathcal{T}_e = \{(\mathcal{N}, \mathcal{N}_p) \,|\, \mathcal{N}, \mathcal{N}_p \in \mathcal{T}, \mathcal{N} \neq \mathcal{N}_p\}$ is the set of edges representing the parenthood between the nodes. In this undirected graph $G_\mathcal{T}$, each node has not only its parent but also its children as its neighbors. For instance, the neighbors for the node $A \cup E$ of the binary partition tree in Figure 2.4 are the nodes A, E, and $A \cup E \cup F$.

For each node $\mathcal{N}$ in all the tree-based image representations, we can assign an attribute function $\mathcal{A}$ that characterizes some interesting feature of the region (*i.e.*, connected component) represented by that node $\mathcal{N}$. The attribute function can be any measurement as simple as the region area, or the compactness of the region, or some shape attribute that describes the region form, or the importance of the region boundary, or even some specifically designed feature based on certain prior information. A tree with its nodes weighted by an attribute function can be seen as a nodes weighted graph $(G_\mathcal{T}, F_\mathcal{A})$, where $F_\mathcal{A}$ is an element of nodes mapping $\mathcal{F}$ given by the attribute function $\mathcal{A}$.

2.4.1 Definition of tree-based shape space

All the tree-based image representations depicted in Section 2.3 provide a *tree-based shape space* $S_\mathcal{T}$ (the name *shape space* is sometimes used in the sequel) defined as:

Definition 8 *A tree-based shape space* $S_\mathcal{T}$ *is defined as a set of regions* $\{R_i\}$ *(R_0 is the entire image domain), that can be organized into a tree structure* $\mathcal{T}$ *for which*

the root node represents R_0, and any pair of regions satisfies the nesting property defined by Eq. (2.15) (i.e., they are either nested or disjoint). The neighborhood of the shape space is defined by the inclusion relationship between those regions.

A tree-based shape space $S_{\mathcal{T}}$ is built from a tree-based image representation, which is equivalent to the image in the sense that the image can be reconstructed from the set of tree nodes (the tree structure is not needed). Hence, a tree-based shape space is an equivalent image representation. Besides, for the tree-based shape spaces built from thresholding decomposition based trees, they are invariant to affine contrast changes.

Note also that the tree-based shape space $S_{\mathcal{T}}$ is a connected graph which is similar to the image space. Each node in the image space is an individual point weighted by some intensity function f, the adjacency of the image space is usually 4-connectivity (resp. 6-connectivity) or 8-connectivity (resp. 26-connectivity) in 2D images (resp. 3D images), while each node in the shape space is a individual connected component weighted by some attribute function $\mathcal{A}$, the adjacency of the nodes is given by the parent-child relationship between the nodes (*i.e.*, inclusion relationship between the regions represented by those nodes) in the shape space. The similarity between the shape space and the image space shed some light on a novel concept: applying connected operators on the shape space. This will be developed in Chapter 3.

2.4.2   Tree-based shape spaces versus scale-space

Following Marr's computational theory of vision [Marr 1983], a number of image analysis systems are based on a bottom-up architecture, made up of two stages: *low level* analysis (characteristic points, contours, regions ...) and *high level* vision tasks (object recognition, scene interpretation ...). It is also assumed that the low level and high level stages are completely independent. The interesting structures that can be useful for high level tasks can be located at arbitrary position in the image, can have any size, and can be very salient as well as very poorly contrasted. So the low level processes should be uncommitted in terms of position, size and contrast [Lindeberg 1994], and its output should be a multi-scale, and contrast invariant general description independent from any specific high level task. The tree-based shape spaces are very suitable for such desired low level analysis. Many applications based on low level analysis using the tree-based shape spaces are given in Part III.

A popular multi-scale analysis tool which is widely used in image processing and computer vision is the *scale space*. It is a formal theory handling image structures at different scales, by representing an image as a one-parameter family of smoothed

images, the scale-space representation, parametrized by the size of the smoothing kernel used for suppressing the fine-scale structures [Lindeberg 1994]. The scale parameter σ can be seen as a third dimension of a 2D image. The main type of the scale-space is the linear Gaussian scale-space [Lindeberg 1994]. For a given 2D image $f(x,y)$ its linear Gaussian scale-space representation is a family of smoothed images $L(x,y;\sigma)$ obtained by the convolution of $f(x,y)$ with the Gaussian Kernel:

$$g(x,y;\sigma) \;=\; \frac{1}{\sqrt{(2\pi\sigma)}}e^{-(x^2+y^2)/2\sigma}, \qquad\qquad (2.16)$$

such that

$$L(\cdot,\cdot;\sigma) \;=\; g(\cdot,\cdot;\sigma) * f(\cdot,\cdot). \qquad\qquad (2.17)$$

There exist also some non-linear scale-spaces in the literature in order to correct some shortcomings of linear scale space, such as edge localization. The first type of such examples is based on non-linear diffusion [Perona 1990, Harvey 1997]. However, these techniques usually tend to be computationally expensive. Another type of non-linear scale-spaces are based on mathematical morphology. Such morphological scale-spaces are usually more computationally efficient. Examples are dilation-erosion scale space [Jackway 1996], open and close scale-spaces [Chen 1989, Park 1996, Jackway 1998], area morphology based scale-space [Bangham 1996], and morphological levelings based scale-space [Meyer 2000].

The tree-based shape space inherently embeds a similar morphological scale space, because the parent of a node is always larger. In consequence, a tree-based shape space obeys the causality principle, the most fundamental principle of multi-scale analysis [Koenderink 1984]. Indeed, a region R at certain scale is included in all its ancestor nodes till the root node. This region R can be seen as a kind of cause of all its ancestor nodes.

The scale-spaces based on smoothing kernel with different scale parameters blur the contours in image, and they are not invariant to contrast changes. The tree-based shape spaces are also multi-scale analysis tools. But contrary to the scale-spaces, the tree-based shape spaces do not apply a convolution with a kernel. As the connected components in a shape space are already presented in the original image, the contours of the regions in a shape space are actual contours in the original image. Consequently, they are very precised in terms of contours shapes and locations. This good contour preservation property, as well as those interesting features discussed in Section 2.4.1 make the differences between the scale-space and tree-based shape space in applications depicted in Part III.
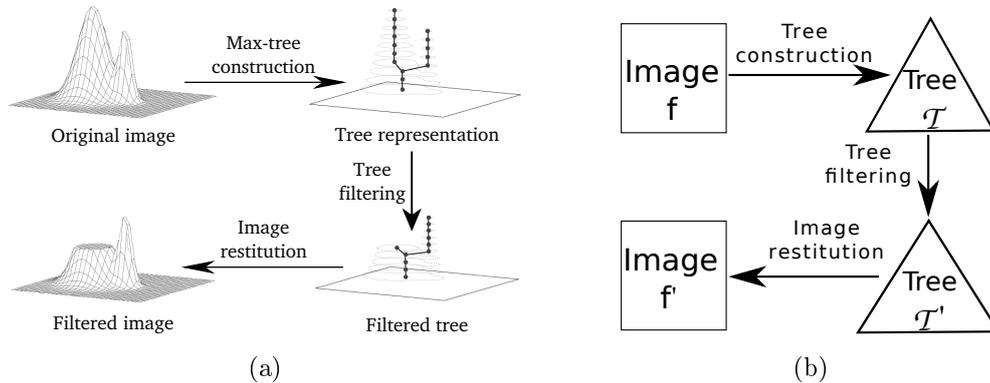
Figure 2.11: An example (a) and a schematic overview (b) of tree-based connected operators implementation [Salembier 1998, Salembier 2000]. In (a), the tree filtering strategy is a tree pruning process, and for those pixels contained in the pruned nodes, they take the value of the lowest preserved ancestor in the image reconstruction step.

2.5   Tree-based implementation

2.5.1   Schematic overview

One popular implementation of connected operators described in Section 2.2 relies on transforming the image into an equivalent representation, the tree-based image representation. Image filtering then involves the design of an attribute function $\mathcal{A}$ that weights some interesting feature of a node of the tree (*e.g.*, a certain shape attribute $\mathcal{A}_s$ that measures how much a node fits a given shape), a tree filtering that simplifies the tree based on the attribute function $\mathcal{A}$, and an image reconstruction step from the simplified tree that yield a filtering result [Salembier 2000, Salembier 2009]. An example of such tree-based implementation of connected operators is illustrated in Figure 2.11 (a) [Salembier 2000], and a schematic overview of such process is depicted in Figure 2.11 (b). Roughly speaking, a tree-based implementation of connected operators consists of three steps: tree construction, tree filtering and image reconstruction. Several available tree-based image representations are already reviewed in Section 2.3. The choice depends on the input image and the application, in general, the type of tree is chosen to make the connected components that we are interested with or that we want to discard be present in the tree. The tree filtering step and image reconstruction step will be respectively detailed in Section 2.5.2 and Section 2.5.3.

2.5.2   Tree filtering strategies

Once the tree is constructed, the connected operators involve a tree filtering step. This is the critical step of the entire process. In general, based on the filtered nodes, the tree filtering strategies can be divided into two classes: the tree pruning and non-pruning strategies. Tree pruning strategies consist of pruning the whole sub-trees rooted in some specific nodes, while keeping the nodes above those specific nodes intact. It can be seen as cutting the sub-branches of the tree. An example of tree pruning is illustrated in Figure 2.11 (a). If a node is filtered by a pruning strategy, then all its descendants are also filtered. Whereas, for the non-pruning strategies, a descendant of a filtered node might be preserved. The use of pruning or non-pruning strategies depends on the property of the attribute function $\mathcal{A}$. We distinguish the attribute function $\mathcal{A}$ based on whether $\mathcal{A}$ is increasing or not.

2.5.2.1   Increasing attributes

One of the simplest case is when the attribute function is increasing, which means:

$$\forall \mathcal{N} \in \mathcal{T}, \mathcal{A}(\mathcal{N}) \leq \mathcal{A}(\mathcal{N}_p), \tag{2.18}$$

where $\mathcal{N}_p$ is the parent of node $\mathcal{N}$. Let $\mathcal{A}^{\uparrow}$ denote the increasing attribute. Some instances of increasing attributes $\mathcal{A}^{\uparrow}(\mathcal{N})$ for a given node $\mathcal{N} \in \mathcal{T}$ are listed as follows:

- $\mathrm{Area}(\mathcal{N}) = \{\#p \,|\, p \in \mathcal{N}\}$;

- $\mathrm{Height}(\mathcal{N}) = \max\limits_{p\in\mathcal{N}} f(p) - \min\limits_{p\in\mathcal{N}} f(p)$;

- $\mathrm{Volume}(\mathcal{N}) = \sum\limits_{p\in\mathcal{N}} \left(\max\limits_{p\in\mathcal{N}} g(p) - g(p)\right)$ with $g = \pm f$ (depends on orientation);

- The diameter of the largest circle that can fit into $\mathcal{N}$ [Breen 1996];

- The diameter of the smallest circle that encloses $\mathcal{N}$ [Breen 1996].

In the case of these increasing attributes $\mathcal{A}^{\uparrow}$, the tree filtering is rather straightforward, it is performed by pruning the nodes whose attribute function $\mathcal{A}^{\uparrow}$ is under a given threshold, which can be seen as an attribute thresholding. The increasingness of the attribute function $\mathcal{A}^{\uparrow}$ make this attribute thresholding a pruning strategy.

2.5.2.2   Non-increasing attributes

In practice, many attribute functions $\mathcal{A}$ are non-increasing attributes, especially for those shape attributes $\mathcal{A}_s$ that describe the form of the shapes. Some instances of non-increasing attributes for a node $\mathcal{N}$ is given as follows:

- Perimeter of $\mathcal{N}$ denoted by $\mathrm{P}(\mathcal{N})$;

- Compactness$(\mathcal{N}) = \frac{4\pi \mathrm{Area}(\mathcal{N})}{\mathrm{P}^2(\mathcal{N})}$, many other shape attributes that measure the circularity are reviewed in the work of Montero and Bribiesca [Montero 2009];

- Elongation$(\mathcal{N}) = l_{\max}(\mathcal{N})/l_{\min}(\mathcal{N})$, where $l_{\max}$ and $l_{\min}$ denote respectively the major and minor axes of the best fitting ellipse having the same moments as the region represented by $\mathcal{N}$;

- Sharpness$(\mathcal{N}) = \frac{\mathrm{Volume}(\mathcal{N})}{\mathrm{Height}(\mathcal{N}) \times \mathrm{Area}(\mathcal{N})}$;

- The maximum geodesic distance in the region of node $\mathcal{N}$ [Breen 1996].

For the non-increasing attributes $\mathcal{A}$, the tree filtering is not straightforward. Salembier *et al.* [Salembier 1998, Salembier 2009] and Urbach *et al.* [Urbach 2007] propose three pruning strategies (*Min, Max, Viterbi*) and an attribute thresholding strategy. The nodes filtering decisions of these rules for a given attribute threshold t are described as follows:

- *Min*: A node $\mathcal{N}$ is removed if $\mathcal{A}(\mathcal{N}) < t$ or if there exists one of its ancestors $\mathcal{N}_a$ such that $\mathcal{A}(\mathcal{N}_a) < t$. An example of such tree filtering is depicted in Figure 2.12 (b).

- *Max*: A node $\mathcal{N}$ is removed if $\mathcal{A}(\mathcal{N}) < t$ and for all its descendants $\mathcal{N}_d$ $\mathcal{A}(\mathcal{N}_d) < t$ holds. An example of the *Max* rule is depicted in Figure 2.12 (c).

- *Viterbi*: The removal and preservation of nodes is determined by a cost optimization process with Viterbi algorithm [Viterbi 1979]. From a leaf to the root, each transition of removal decision is assigned a cost. For each leaf node, the branch with the lowest cost to the root node is taken. Note that this is a pruning strategy, since the cost of a transition from a node $\mathcal{N}$ preservation to the removal of its parent $\mathcal{N}_p$ is assigned an infinitely great cost. So it is impossible to preserve a node while filtering one of its ancestor. More details about the *Viterbi* rule can be found in [Salembier 1998].

- *Attribute thresholding*: A node $\mathcal{N}$ is removed if and only if $\mathcal{A}(\mathcal{N}) < t$. The content of the removed nodes are merged with their the lowest preserved ancestors. This is rather a straightforward approach to handle the non-increasing attributes. Attribute thresholding strategy is originally proposed by Breen and Jones under the name attribute thinnings [Breen 1996]. Based on the image reconstruction rules which will be described in Section 2.5.3, two different tree filtering rules relying on attribute thresholding strategy are proposed: *Direct*

(a) Input   (b) Min   (c) Max   (d) Direct   (e) Subtractive
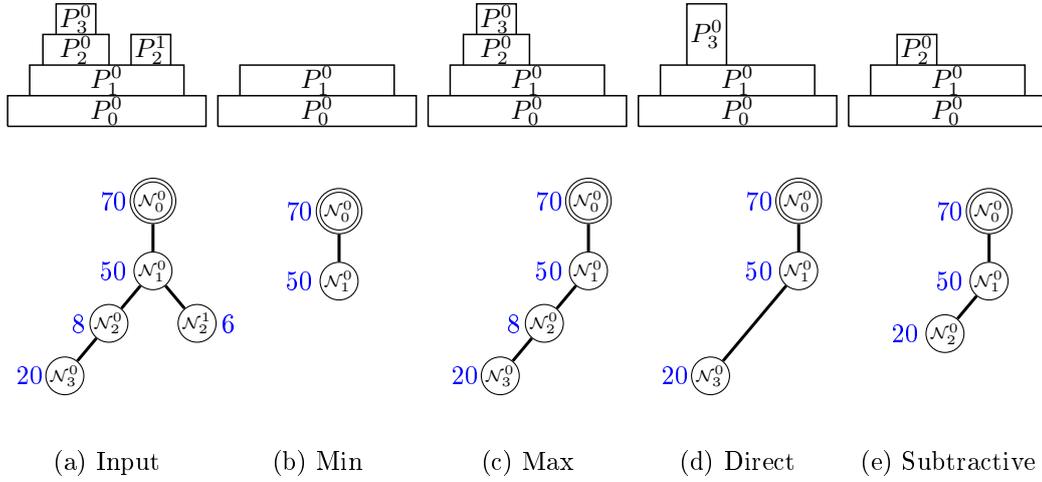
Figure 2.12: An example of tree filtering and image reconstruction in the case of non-increasing attribute. Top row: Input image and reconstructed images from the corresponding filtered trees. Bottom row: Input tree and filtered trees using different tree filtering rules for the non-increasing attribute.

> rule proposed by Salembier et al. [Salembier 1998] and *Subtractive* rule originally mentioned in [Breen 1996], then popularized by Wilkinson and Westenberg [Wilkinson 2001], and Urbach et al. [Urbach 2007]. An example of *Direct* rule and *Subtractive* rule of tree filtering are illustrated respectively in Figure 2.12 (d) and (e).

Let us remark that all those rules dealing with non-increasing attributes have some drawbacks. The pruning strategies cannot deal with the case where two interesting objects are present in the same branch. The two attribute thresholding based rules are although rather simple, they share the practical problems of image thresholding strategies. It is often impossible to retrieve at the same time all the expected objects with one unique threshold. Thresholding based strategies are not robust because the filtering decisions are made locally and do not depend on neighboring nodes (in the case of an increasing attribute, the decision is also local, but the decisions on various levels are known thanks to its increasing property).

2.5.2.3 Individual shape analysis in the shape space

For attribute thresholding based tree filtering strategies, including the tree pruning strategy in the case of increasing attributes, the regions in the tree-based shape space are considered individually. The nodes' removal decisions are made locally upon the attribute values of themselves. The relationship (*i.e.*, neighborhood relationship

given by the parenthood) between those regions in the tree-based shapes is not used, which means the edges of the nodes weighted graph with attribute function representing a tree-based shape space are not utilized. So the attribute thresholding strategies are not robust.

For pruning strategies dealing with the non-increasing attributes, the parenthood relationship between regions in the tree-based shape space is utilized. But only partial of the neighborhood relationship is used, because for each node, it has its parent and its children as its neighbors, whereas only the parent is taken into account. Besides, the pruning strategies maybe interesting in some cases, but two interesting regions might be nested so that they are present in a same branch, so the pruning strategies filter either some interesting regions or many non-interesting regions remain.

2.5.3   Image reconstruction rules

In the case of pruning strategies, some entire sub-trees are removed, whereas, the nodes above those sub-trees are intact, the image reconstruction is trivial. For those pixels contained in the nodes that belong to the removed sub-trees, they take the value of the lowest preserved nodes. Figure 2.11 (a), Figure 2.12 (b) and (c) are such instances. In the case of attribute thresholding based strategies dealing with non-increasing attributes, the removed nodes can be anywhere in the tree, and descendants of a removed node might be preserved. Two image reconstruction rules are proposed: direct rule [Salembier 1998] and subtractive [Breen 1996, Wilkinson 2001, Urbach 2007].

- *Direct*: As the same as the pruning strategies, the contents of the removed nodes merge with the lowest preserved ancestors, and take their values. This is the most straightforward way to reconstruct the image from the simplified (*i.e.*, filtered) tree. However, the local contrasts for the preserved nodes are no longer maintained, and if we subtract the filtered image from the original image, some artificial shapes that should not be removed by the tree filtering step may present in the residual image f_r.

- *Subtractive*: As above, but the gray level differences between those removed nodes and their parents are taken into account by all their descendants, so that the local contrast of preserved shapes remains unchanged. The shapes in the residual image f_r are exactly those removed shapes during the tree filtering step. More specifically, if a node $\mathcal{N}$ is removed, and if the tree is a Max-tree (resp. Min-tree), the descendants of node $\mathcal{N}$ are lowered (resp. augmented) by the same amount of local contrast as node $\mathcal{N}$ itself. However, if the tree

(a) Original image   (b) Direct rule   (c) Subtractive rule

(d) Average reconstruction   (e) Median reconstruction
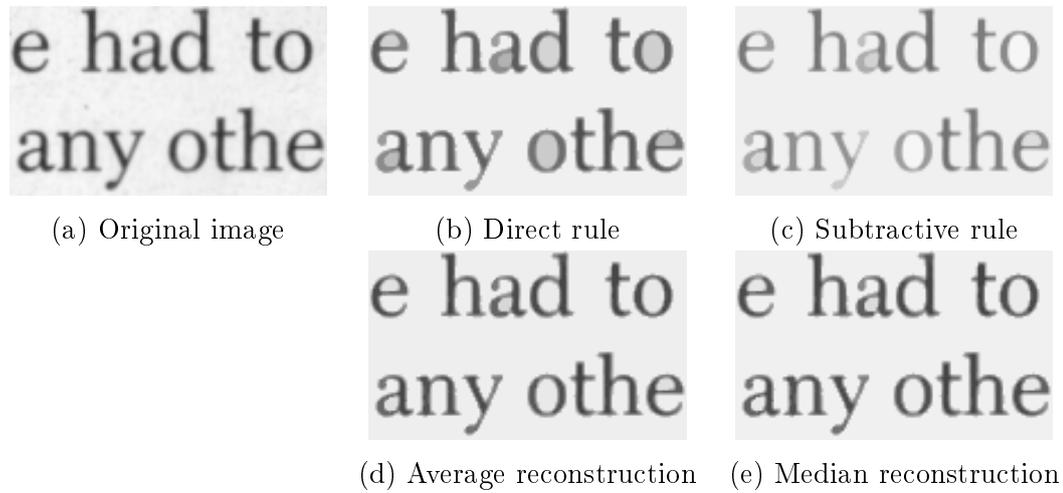
Figure 2.13: An example illustrating different image reconstruction rules. The filtering is performed by removing some nodes having small average of gradient's magnitude in the topographic map.

is a topographic map which is self-dual structure, the accumulated gray level lowering and augmentation are applied to the descendants. More details and the formal definition of subtractive rule can be found in [Urbach 2007].

Other image reconstruction rules might also be interesting, for instance, instead of using the value of the lowest preserved ancestor, we can take the average or the median value of the removed contents to reconstruct the image from a simplified tree. This is not morphological reconstruction, but it might give some interesting result, especially for the case of a topographic map being a self-dual representation. An example of these reconstruction rules is shown in Figure 2.13.

2.6 Some examples using the tree-based shape spaces

Since the proposition of those tree-based image representations described in Section 2.3, many applications are developed using these tree representations in image processing and computer vision. The goal of this section is not to develop these applications, but to show the usefulness of the tree-based shape spaces that we introduced in Section 2.4 and that are inspired from those tree-based image representations. We thus firstly give a short review of some applications from filtering, simplification to segmentation, and visualization of images in image processing relying on the tree-based shape spaces. Besides, some applications in computer vision based on that space are also reviewed. They can all be seen as the extraction of "relevant" shapes from the tree-based shape spaces. The definition of "relevant" de-

(a) Original image, 24127 shapes

(b) $t = 10$, 4643 shapes, PNSR $= 38.4$

(c) $t = 100$, 2119 shapes, PNSR $= 32.9$

(d) $t = 1000$, 958 shapes, PNSR $= 23.3$
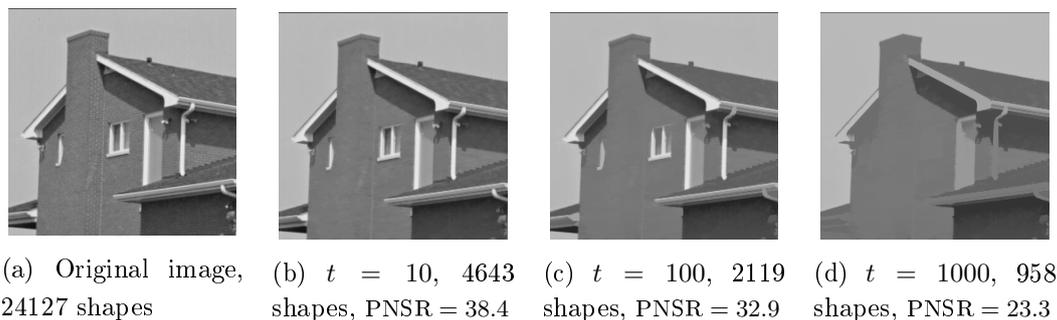
Figure 2.14: Grain filtering with different grain sizes.

pends on applications. Note that there are many other applications relying on the tree-based shape spaces that are not presented here.

2.6.1 Applications in image processing

2.6.1.1 Grain filter

The grain filter was originally and formally introduced by Caselles and Monasse [Caselles 2002]. The grain filters consist in removing the level sets of an image whose area is smaller than a given threshold t. They are known as extrema killers, because they are self-dual filters that remove all the connected components starting from the extrema and having an area smaller than t. Unlike the classical morphological filters based on structural elements and the linear Gaussian filters, the grain filters only remove small level sets, leaving others unchanged. An example of the comparison between these filters is given in Figure 2.2. Note that the result of a grain filter with a area threshold t can still have flat zones whose area is smaller than t, because these flat zones are presented in the nodes whose area is big enough. An efficient implementation of the grain filters is to use the topographic map representation along with the area as the attribute function. An example of the grain filter with different threshold values is illustrated in Figure 2.14.

2.6.1.2 Meaningful level lines extraction

Meaningful level lines (boundaries of level sets) extraction was proposed by Cao et al. [Cao 2005]. This proposed meaningful level lines extraction is based on statistical arguments, which leads to a parameter free algorithm. It is an improvement of the previous work of Desolneux *et al.* [Desolneux 2001] which proposes a parameterless algorithm using the topographic map to detect contrasted level lines, called meaningful boundaries. More specifically, for a given shape τ in the topographic

map, the authors define a meaningfulness for the boundary of that shape τ, know as a level line. The meaningfulness is defined using the *a contrario* model, which was introduced by Desolneux et al. [Desolneux 2001, Desolneux 2008] within the framework of Computational Gestalt Theory. It is based on the so-called Helmholtz principle, which states that: "we naturally perceive whatever could not happen by chance". So the *a contrario* models first assume uniform background or noise model, then define the interesting events as large deviations from this model. these deviations are measured by the *Number of False Alarms (NFA)*. For the meaningful level lines extraction case, the NFA reflects the meaningfulness. Then extract one level line with the smallest NFA among each monotone branch of the topographic map, where a monotone branch is a branch of the topographic map for which each node has only one child and the gray level value of each node is either strictly increasing or strictly decreasing. By this method, only around 1% level lines are selected that coincide with pieces of edges in the image, and the image represented by these extracted level lines has nearly no loss of shape contents, and they delivers accurate shape elements. The details about the definition of NFA and meaningful level lines extraction can be found in [Desolneux 2001, Cao 2005].

2.6.1.3 Image simplification and segmentation

Since the beginning of the introduction of the morphological trees and the binary partition tree, they are used to implement connected operators which act by merging flat zones, so there exist many works that achieve image simplification and segmentation relying on the "relevant" shapes extraction from the tree-based shape spaces. For instance, the work of Salembier *et al.* [Salembier 1998] which proposes a set of filtering strategies that simplify images using the Max-tree representation. Then in [Salembier 2000], the authors propose to use the binary partition tree to achieve image simplification, segmentation, and information retrieval. In [Wilkinson 2001], the authors propose to use the Max-tree and some shape attributes to simplify images so that the filament is enhanced. Ballester *et al.* propose in [Ballester 2007] to use the topographic map with variational models (*e.g.*, minimize the piecewise constant Mumford-Shah functional [Mumford 1989]) for segmentation and encoding. In [Lu 2007], the authors propose to simplify the binary partition tree, so the image by analyzing the second order statistics of some evolvement functions using a knee function. Knee values show the reluctancy of each merge, which helps to filter the tree and yield a simplified image.

A segmentation result can be simply obtained by an horizontal cut of a hierarchy of segmentations. The works of Najman et al. [Najman 1996, Najman 2009] are such instances. In [Felzenszwalb 2004], the authors propose an efficient graph-based

image segmentation algorithm using the minimum spanning. It consists in first of
all sorting the edges belongs to minimum spanning tree by increasing importance
of pixel edges linking two pixels , and then for each edge of MST in this pre-sorted
order, if the two regions linked by that edge is not merged yet, examine if we can
merge them based on the difference maximal intra-dissimilarity and minimal inter-
dissimilarity controlled by a parameter K. In [Guigues 2006], the authors propose
the scale-sets theory that can provide an optimum cut (usually not horizontal) very
efficiently subordinated to some energy functional to minimize. In [Soille 2008],
Soille proposes an image simplification and segmentation method based on the con-
strained connectivity which yields the α-tree. Then a local range α combined with
a global range ω or not is used to cut the hierarchy that provides a simplification
or segmentation result. In [Serra 2013, Kiran 2014], Serra and Kiran propose a new
approach to find optimal cuts in hierarchies of partitions by energy minimization. It
relies on the notion of h-increasingness, and allows to find optimal cuts in one pass.
Recently, Cardelino et al. [Cardelino 2013] propose to select the best (optimal) par-
tition (cut) from a hierarchy of segmentations based on the use of *a contrario* model,
they assign a meaningfulness reflected by the NFA to each possible partition given
by that hierarchy. The optimal partition is simply given by the most meaningful
partition.

2.6.1.4   Preferential image segmentation

In [Pan 2009b], the authors propose a novel preferential image segmentation method,
which preferentially segments objects that have intensities and boundaries similar
to those of objects in a database of prior images. This method relies on the topo-
graphic map representation. It consists in first of all constructing the topographic
map for both images. For the preferential object O_p to which the similar objects in
another image f' from the database is to be segmented, the intensity and bound-
ary information of the corresponding shape in its topographic map is known. The
proposed method first of all select a reduced set of shape candidates in the cor-
responding topographic map of f' based on the intensity information, such as the
number of direct children, the relative area change between the shape and its direct
children, the rough similarities of the boundaries based on the compactness. Then
a curve matching step introduced in [Lisani 2003] is applied on the boundaries of
the reduced shape candidates in the topographic map representing f' and the pref-
erential shape O_p. The candidate shape which best matches the preferential curve
is finally selected as the preferential segmentation result. Experimental results on a
large image dataset show that this application using topographic map is promising.
The readers are referred to [Pan 2009b] for more details about this application.

2.6.1.5 Visualization of images

For 3D images visualization, despite the fact that they can not be directly printed on paper, one usually displays slices, or voxels, or isosurfaces, and expects that the content of the 3D image is understandable from the figure. However, in practice, visualization of 3D images is extremely interesting and difficult. For the visualization by voxels and isosurfaces, some interesting objects may be hidden inside some surrounding structures, or surrounded by many noises, which make the interesting objects visualization impossible or difficult. In [Wilkinson 2001], the authors propose to use the Max-tree combined with moment of inertia (minimal for a sphere, and increases rapidly as the object becomes more elongated) as attribute function to filter the 3D images of vessels. In the simplified 3D image, the filament (*i.e.*, vessels) is enhanced so that we can better visualize the vessels in the image. In [Westenberg 2007], the authors also propose some other attribute function, such as elongation, flatness and sparseness to filter the Max-tree of a 3D image, which results a simplified image where the interesting objects are more visible.

2.6.2 Applications in computer vision

2.6.2.1 Maximally Stable Extremal Regions (MSER)

Maximally Stable Extremal Regions (MSER) was originally proposed by Matas *et al.* [Matas 2002]. It is a method of blob detection in images, which belongs to the family of local feature detection. It is used to establish the correspondences between image elements from two images to be compared, and is widely used in stereo matching, object recognition, and tracking. The MSER was originally defined as follows:

Region Q is a connected component in image.

(Outer) Region Boundary $\partial Q = \{q \in V \backslash Q : \exists p \in Q : e = (p, q) \in E\}$, which means the boundary ∂Q of Q is the set of pixels adjacent to at least one pixel of Q but not belonging to Q.

Extremal Region $Q \subset V$ is a region such that either $\forall p \in Q, q \in \partial Q : f(p) > f(q)$ holds or $\forall p \in Q, q \in \partial Q : f(p) < f(q)$ holds.

Maximally Stable Extremal Region (MSER). Let $Q_1, \ldots, Q_{i-1}, Q_i, \ldots$ be a sequence of nesting extremal regions ($Q_i \subset Q_{i+1}$). Extremal region Q_{i*} is maximally stable if and only if $q(i) = |Q_{i+\Delta} \backslash Q_{i-\Delta}|/|Q_i|$ has a local minimum at $i*$, where $| \cdot |$ denotes cardinality, and $\Delta \in \mathbb{Z}$ is a parameter of the method.

The stability function $q(i)$ checks for regions that remain stable over a certain number of thresholds. If an extremal region $Q_{i+\Delta}$ is not significantly larger than an extremal region $Q_{i-\Delta}$, the extremal region Q_i is selected as a maximally stable

extremal region.

The set of extremal regions can be organized into a tree structure: component tree [Najman 2006] (either a Min-tree or a Max-tree), and the MSER is easily understandable using Min-tree and Max-tree representations. As shown in [Donoser 2006], MSER extracts the regions (nodes) that correspond to local minima of the stability function along the path to the root of the tree. The stability function of a given node is given by the difference between the area of some (grand-)parent and some (grand-)child, divided by the area of the node itself. It is reported [Mikolajczyk 2005] that MSER achieves state-of-the-art repeatabilities and regions accuracies. It is also very efficient. Nister and Stewenius propose in [Nistér 2008] a linear algorithm (similar to the one of Salembier *et al.* [Salembier 1998]) to compute the MSER.

A variant of MSER is proposed by Perdoch et al. in [Perdoch 2007] called the Stable Affine Frame (SAF) for which only local stability is required. Many more features are obtained with a comparable repeatability score. However, it is much slower than MSER.

2.6.2.2   Topological approach to hierarchical segmentation using Mean-Shift

Mean shift is popular method to segment images and videos. Pixels are represented by feature points which encodes the spatial information of that point in the original image and the color information, and the segmentation is driven by the point density in feature space. It has been shown by Cheng [Cheng 1995] and Comaniciu and Meer [Comaniciu 2002] that mean shift is equivalent to a steepest ascent on a density function underlying the image data. In [Paris 2007], Paris and Durand propose an efficient scheme to evaluate that density function. They apply the Morse theory [Milnor 1963] on the explicit representation of the underlying density function to extract the density modes corresponding to the clusters. Based on the Morse theory, it has been shown that the density modes are unions of cells of the Morse-Smale complex. This approach leads to a fast method to compute mean-shift segmentations. Besides, With the notion of topological persistence introduced by Edelsbrunner et al. [Edelsbrunner 2000], they build a hierarchical segmentation at the computational cost of a single-level clustering.

The core structure used in this method is the Max-tree representation constructed from the underlying density function defined on the feature space, not directly from the original image. The Max-tree representation make the analysis of maxima and saddle points very efficiently. In fact, the filtering using the topological persistence in [Paris 2007] is a filtering by height applied on the Max-tree representation. It has been shown that this method achieves an accuracy equivalent

to previous technique but runs faster then all the previous work, especially on large images and videos. Besides, a hierarchical segmentation can be obtained with no additional computational cost based on the extinction values using height, known also as dynamic. This hierarchical segmentation is useful for multi-scale analysis.

2.6.2.3 Scenery images analysis

A first example of scenery images analysis making use of the tree-based shape spaces is proposed by Song and Zhang in [Song 2002] to separate the foreground and background regions in images. More specifically, they made two assumptions about the image background. Compared to the foreground regions, the background regions are (1) smoother in intensity; and (2) peripheral in location. In an input image, the harshness of the foreground is characterized by intensity peaks and valleys, which correspond to the nodes of topographic map representing small textons whose height are high. The complete system is consist of first of all selecting these nodes from the topographic map as foreground objects, then connecting those textons controlled by the distance between them. A noise removal step is also applied by removing small connected components that contains very few textons after textons connecting step. Then repair the boundary by connecting all the close points. finally, a hole filling step is performed to achieve the final foreground region. It has been shown in [Song 2002] that such scheme can effectively locate background region in images which satisfy the two assumptions.

Lately, Song and Zhang propose in [Song 2003] a method for analyzing scenery images to support semantics-based image retrieval. The method makes use of the monotone tree (*i.e.*, topographic map). The structural elements of an image are modeled as sub-branches of the topographic map. These structural elements are classified and clustered on the basis of such properties as color, spatial location, harshness and shape. Each cluster corresponds to some semantic feature. It has been show in [Song 2003] that such scheme is efficient for analysis and retrieval of scenery images.

2.6.2.4 Classification of images

In [Urbach 2007], the authors propose a multiscale and multishape morphological method for pattern-based analysis and classification of gray-scale images using connected operators relying on the Max-tree representation. Pattern spectra [Maragos 1989, Serra 1982] are commonly used tools for image analysis and classification, which can be computed using a technique from mathematical morphology known as granulometries [Breen 1996, Serra 1982]. Intuitively, a size granulometry can be considered as a set of sieves of different grades, each allowing

details of certain size classes to pass. In [Urbach 2004, Urbach 2007] the authors propose to use the joint shape-size pattern spectra computed from connected operators which makes use of the Max-tree representation. Compared with previous existing methods, which use structuring elements, it is stated out that the proposed method in [Urbach 2007] features three main advantages: 1) The computation time is independent of the dimensions of the pattern spectrum, since it does not depend on the number of scales or shapes used. 2) The method can make use of exact shape attributes to construct a joint 2D shape-size pattern spectra. 3) It is significantly less sensitive to noise and it is rotation-invariant. The benchmark of the classification performance on four image datasets shows that their proposed method achieves better or equal classification performance to the best competitor but with a 5 to 9-fold speed gain.

Morphological profiles [Epifanio 2007, Tuia 2009] are widely used for the classification of high-resolution remote sensing images. Recently, Luo and Zhang propose in [Luo 2013] a robust auto-dual morphological profiles for the classification of high-resolution satellite images. They make use of the topographic map representation, and extract some other feature profiles (perimeters, scales, total variations, etc.) that are more robust than the intensity variations from that representation. The efficiencies of their proposed method are validated by experimental results on two datasets of remote sensing images.

2.6.2.5 Texture indexing

In [Xia 2010], Xia *et al.* propose a texture analysis scheme, the shape-based invariant texture indexing, which is invariant to local geometric and radiometric changes. Their proposed method makes use of the topographic map representation, which is a multi-scale and contrast invariant representation of images. More specifically, for each texture image, its corresponding topographic map is firstly computed, then for each shape in the topographic map, assign some invariant texture feature based on the second order moments (having many invariant properties, such as translation, scale, rotation, etc.) of that shape. Then each image is associated with a 1D histogram of the texture features for those shapes belonging to the corresponding topographic map. Then the comparison two textual images is performed through the distance between their 1D histograms of texture features of their shapes. Their proposed texture indexing approach is validated by performing classification and retrieval experiments on three texture databases. Their obtained results outperform previous state of the art in locally invariant texture analysis.

2.6.2.6 Object detection

In [Vilaplana 2008], Vilaplana *et al.* propose an approach of object detection based on the use of binary partition trees. The BPT provides a tremendously reduced search space for the object detection task. In order to handle the compromise between computational complexity reduction and accuracy, the authors propose to distinguish two zones in the BPT: the *accuracy space* providing preciseness to the description (lower scales) and the *search space* for the object detection task (higher scales). These two zones are defined by specifying a point of merging sequence why is obtained by assessing a stop criterion. A rough object detection is obtained by searching the node in the search space which is most likely to be a meaningful object. Then the region represented by this detected object node is refined by introducing a shape fitting step. The object refinement step is performed in the accuracy space. Various experimental results in [Vilaplana 2008] illustrate the generality and the efficiency of their proposed method for object detection.

2.7 Conclusion

We have reviewed the tree-based connected operators in this chapter. They represent the context of the core concept of our PhD work. Many tree-based image representations are reviewed. And some application examples that rely on those tree-based image representations are shortly introduced to demonstrate their usefulness for many problems in image processing and computer vision. The novelty of this chapter is the introduction of the tree-based shape space. It is the corner stone of our proposed framework described in Chapter 3. This notion provides us with a new point of view of the classical connected operators. They can be seen as an atomic analysis of this shape space, without using the structure of the tree. Besides, many applications in image processing and computer vision relying on tree representations can be seen as relevant/representative points selection from the shape space. In some cases, the selection/decision is performed individually, whereas, the local structure of the tree is sometimes used to guide the selection/decision.

Shape-based morphology framework

This chapter presents the core concept of our PhD work presented in this thesis: the framework of *shape-based morphology*. The basic idea of this framework (see Section 3.1) is to apply the connected operators in the tree-based shape space. It consists of two tree constructions: one is constructed from the image, and the second one is constructed from the first tree representation. The principle explaining the rationale behind this framework is detailed in Section 3.2. We will show that this framework allows us to provide a simple definition of the MSER method [Matas 2002] and an extension of this method (see Section 3.3). Besides, this framework has three main consequences. 1) For filtering purpose, detailed in Section 3.4, it is a generalization of the classical existing connected operators. Two novel kinds of filters are introduced: the shape-based lower/upper levelings, and the self-dual morphological shapings. 2) As shown in Section 3.5, this framework can be used to object detection/segmentation by selecting relevant points from the shape space. 3) This framework provides a way to transform any tree representation into a hierarchy representing a hierarchical image simplification/segmentation. This aspect will be detailed in Section 3.6.

3.1 Overview of the framework

3.1.1 Classical tree-based connected operators

As reviewed in Section 2.5 and depicted in Figure 2.11 (b), the classical tree-based connected operators is consisted of three steps: tree construction, tree filtering, and image reconstruction from the simplified tree. The core process of such process relies on the tree filtering step. For an increasing attribute function $\mathcal{A}^\uparrow$, the tree filtering is simply performed by pruning the leaves which is equivalent to applying an attribute thresholding. For a non-increasing attributes (the most usual cases, *e.g.*, many shape attributes), the pruning strategies reviewed in Section 2.5.2 do not take into account the possibility that several relevant objects can have some

inclusion relationship, meaning that they are on the same branch of the tree. For attribute thresholding strategies, one simply removes the nodes of the tree for which the attribute is lower than a given threshold. Such a thresholding does not take into account the intrinsic parenthood relationship of the tree, the regions in the shape space are analyzed individually. It is often impossible to retrieve all expected objects with one unique threshold. Figure 3.1 shows the evolution of a shape attribute, the circularity, along two branches of the topographic map. The light round shape and the dark one are both meaningful round objects when compared to their context. However, their attribute values are very different. In order to obtain the light one, a high threshold is required, but then some non-desired shapes appear in the background in Fig. 3.1 (f).

3.1.2   Connected operators on shape space

The founding idea of the framework that we call *shape-based morphology* is to apply connected filters on the tree-based shape space $S_{\mathcal{T}}$ defined in Section 2.4, being the space of all the nested or disjoint components of the image, structured into a graph by the parenthood relationship (*i.e.*, the neighbors of a node are its children and its parent). Each node in the shape space is weighted by an attribute function $\mathcal{A}$. This process of shape-based morphology is illustrated by the black+red path of Figure 3.2. Note that the process depicted in the red block of Figure 3.2 is exactly a tree-based connected operator applied on the tree-based shape space $S_{\mathcal{T}}$ induced by a tree-based image representation $\mathcal{T}$. It is composed of three steps: tree $\mathcal{TT}$ construction, tree $\mathcal{TT}$ filtering, and a simplified tree $\mathcal{T}'$ (equivalent to the simplified image f' of the connected operators applied to the space of image) reconstruction from the simplified tree $\mathcal{TT}'$. The use of tree-based shape space make this process act by merging flat zones, which implies that the shape-based morphology still belongs to the family of connected operators.

This surprising and simple idea of the shape-based morphology framework has several deep consequences:

- This framework allows us to give a new point of view of the widely used MSER [Matas 2002], and it also gives us the possibility to extend the definition of MSER by analyzing the tree-based shape space based on some attribute function other than the stability function used in the original definition of MSER [Matas 2002]. This will be detailed in Section 3.3.

- For the filtering purpose, it will be shown in Section 3.4 that this framework encompasses some usual attribute filtering operators. Novel connected filters based on non-increasing criterion can also be proposed. When the first tree $\mathcal{T}$

(a)

(b)   (c)   (d)

(e)   (f)   (g)
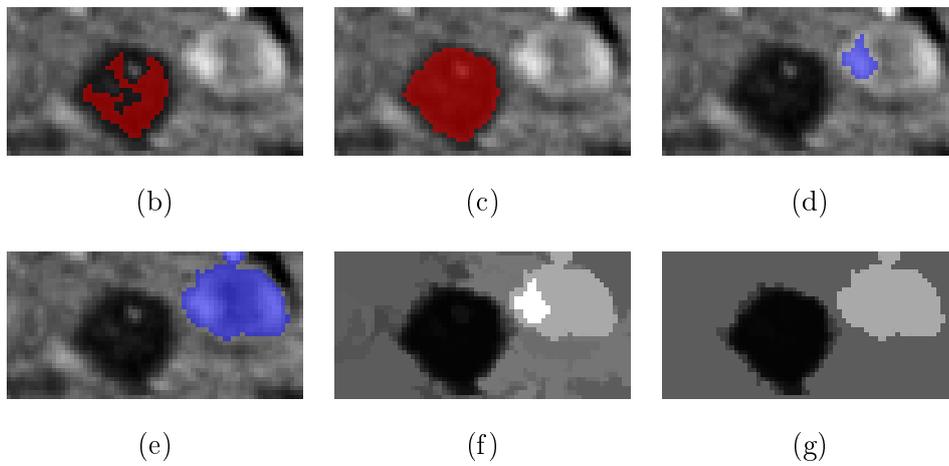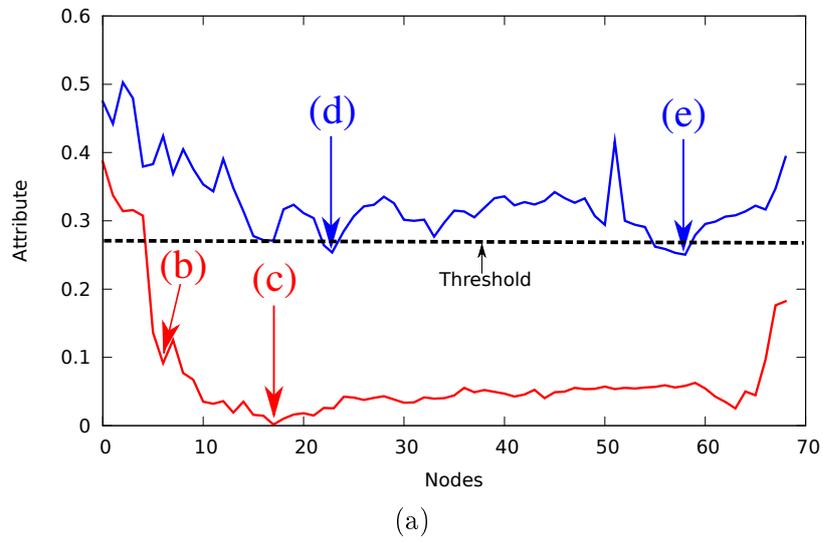
Figure 3.1: (a) Evolution of the circularity attribute on two branches of $\mathcal{T}$ being the topographic map; (b to e) Some shapes; (f) Result of attribute thresholding; (g) Result of a shaping.
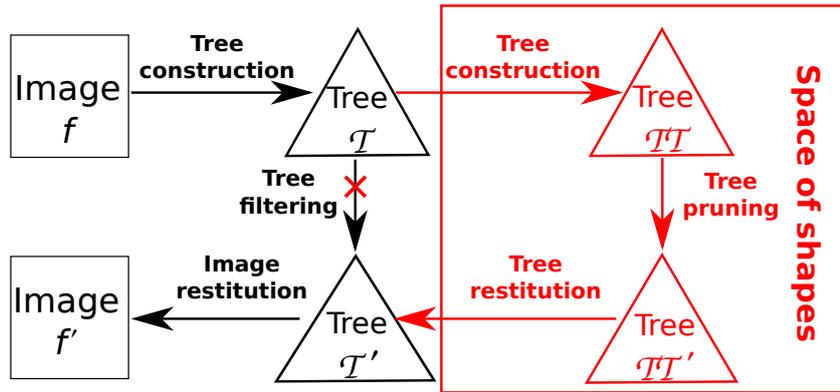
Figure 3.2: Classical connected operators (black path) and our proposed shape-based morphology scheme (black+red path).

is respectively a Min-tree or a Max-tree [Salembier 1998], such filters are new morphological lower or upper levelings [Meyer 1998], [Meyer 2004]. When the first tree $\mathcal{T}$ is the topographic map, it gives rise to a novel family of self-dual connected filters that we call *morphological shapings*.

- This framework can also be used for object detection and segmentation purpose, where the basic idea is that the local minima of some attribute function $\mathcal{A}$ correspond to meaningful objects, and the morphological filtering in shape spaces helps to remove meaningless local minima, so does the meaningless objects. This will be detailed in Section 3.5,

- By expanding the idea of the principle of object detection, we can weight the contours of regions represented by the local minima in the shape space by the filtering force (known as extinction value [Vachier 1995]) for which a local minimum disappear, the obtained image of weight is a saliency map which is equivalent to a hierarchy of image segmentation. Consequently, the framework of shape-based morphology provides an efficient way to transform hierarchy (see Section 3.6).

Roughly speaking, it will be shown in the sequel that the proposed framework of shape-based morphology is a versatile framework that can be used for extending MSER, image filtering, object detection/segmentation, and transformation of hierarchy.

3.2 Principle of the shape-based morphology

Let us explain the principle of the shape-based morphology that acts by applying connected operators on the tree-based shape space, instead of on the space of image directly. We will detail each step of the framework depicted in Figure 3.2.

3.2.1 First tree construction

As illustrated in Fig. 3.2, the first step of shape-based morphology is to construct a tree-based image representation $\mathcal{T}$, which is equivalent to the image f. Different trees are reviewed in Section 2.3. The choice of the kind of tree depends on the targeted application. The criterion to choose a particular tree among the set of trees is that the shapes (objects) of interest, are present as connected components of this tree. As stated in Section 2.3, all trees can be constructed efficiently.

During this tree construction, we are able to compute incrementally a lot of information, based on which the attribute $\mathcal{A}$ (some interesting feature) characterizing the corresponding connected components will be obtained. The attribute $\mathcal{A}$ can be as simple as for instance, the shape area (increasing attributes), or some more evolved ones which are non-increasing and usually more interesting. That is for example the case of compactness, or elongation, etc [Westenberg 2007].

As explained in Section 2.4, a tree-based shape space $S_\mathcal{T}$ given by a tree-based image representation $\mathcal{T}$ with nodes weighted by an attribute function $\mathcal{A}$ is equivalent to a node weighted graph $(G_\mathcal{T}, F_\mathcal{A})$, where $G_\mathcal{T} = (\mathcal{T_N}, \mathcal{T}_e)$, $\mathcal{T_N}$ is the set of nodes $\{\mathcal{N} \,|\, \mathcal{N} \in \mathcal{T}\}$, and $\mathcal{T}_e$ is the set of edges $\mathcal{T}_e = \{(\mathcal{N}, \mathcal{N}_p) \,|\, \mathcal{N}, \mathcal{N}_p \in \mathcal{T}, \mathcal{N} \neq \mathcal{N}_p\}$ that encode the parenthood relationship between nodes. $F_\mathcal{A}$ is an element of nodes mapping $\mathcal{F}$ given by the attribute function $\mathcal{A}$. The Figure 3.3 depicts a simple tree $\mathcal{T}$ with four regional minima (represented by red circles) on the graph.

3.2.2 Second tree construction

Just like the first tree construction, a second tree-based image representation $\mathcal{TT}$ is constructed in this second step. This second tree $\mathcal{TT}$ is built from the graph $(G_\mathcal{T}, F_\mathcal{A})$, whereas the first tree was built from the input image. The second tree $\mathcal{TT}$ is either a Min-tree or Max-tree representation. The choice between them is based on the application and on the nature of the attribute function $\mathcal{A}$. In the case of non-desired shapes filtering, the criterion is to keep the vertices in $G_\mathcal{T}$ (*i.e.*, nodes $\mathcal{N} \in \mathcal{T}$) representing the non-desired shapes near the leaves of the second tree $\mathcal{TT}$. For example, if we want to filter out the non-desired shapes, and if $\mathcal{A}$ encodes the probability for a shape to be of a desired type, the minima of the space of shapes are the shapes that are less probable to be of that type, compared to their parents
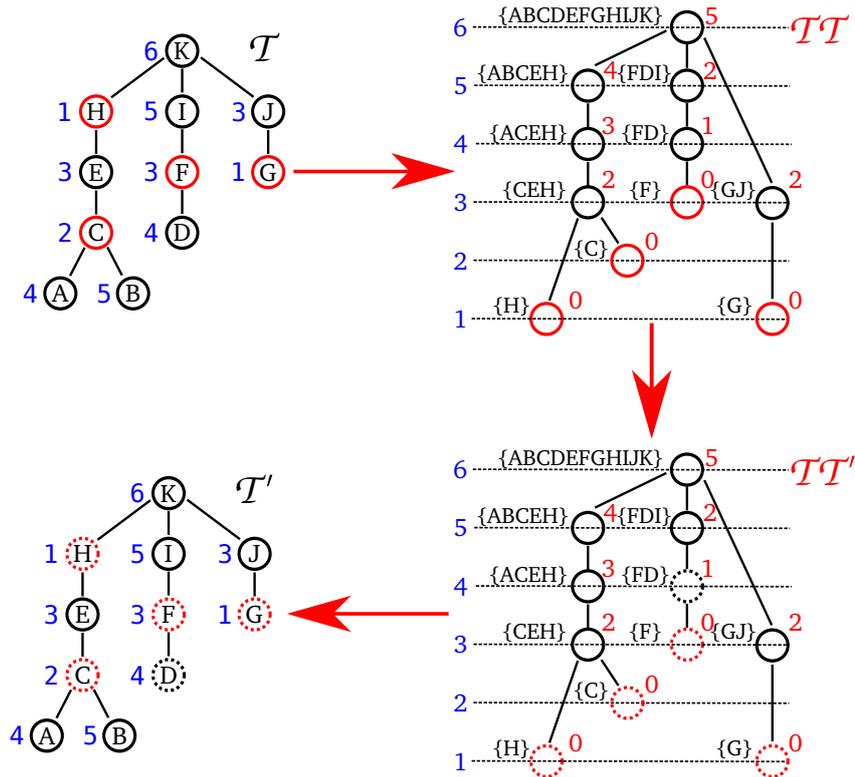
Figure 3.3: An example of the workflow of shape-based morphology with $\mathcal{TT}$ being the Min-tree of $\mathcal{T}$. Circles with capital letter inside: nodes $\mathcal{N}$; Blue values: first attribute $\mathcal{A}$; Circles without letter inside: nodes $\mathcal{NN}$; Red values: second attribute $\mathcal{AA}$ (being the height of the first attribute $\mathcal{A}$ in this example); Dashed circles: filtered nodes with a given threshold 2 for the second tree pruning.

and children, *i.e.*, the neighbors of the node representing the shape in the graph $(G_\mathcal{T}, F_\mathcal{A})$. Then the choice of $\mathcal{TT}$ will be a Min-tree representation. However, if we want to keep the shapes being very likely to the desired shapes, the criterion is chosen to make the leaves of $\mathcal{TT}$ correspond to the desired shapes. Then $\mathcal{TT}$ will be the Max-tree. Each node $\mathcal{NN}$ of the second tree $\mathcal{TT}$ is a set of neighboring connected components with similar type of shapes. Remark also that two different "objects" located in the same branch of $\mathcal{T}$ are now possibly present on two different branches of $\mathcal{TT}$ (*e.g.*, in Figure 3.3 the node C and H of $\mathcal{T}$ are now in two distinct branches of $\mathcal{TT}$).

A second attribute $\mathcal{AA}$ characterizing each node $\mathcal{NN}$ of the second tree $\mathcal{TT}$ is required to apply the second tree processing. It is always designed to be an increasing attribute in order to make the second tree filtering be a simple pruning strategy. The design of this second attribute $\mathcal{AA}$ is quite flexible. Usually, it can be also computed incrementally during $\mathcal{TT}$ construction, based on the first attribute $\mathcal{A}$ (For instance, the height, or the volume of $\mathcal{A}$), or based on the contextual information on the image domain around the shapes that $\mathcal{NN}$ represents. An example of this later case is the area, a second example is the total variation inside the region formed by the union of element "region" $R_\mathcal{N}$ of shapes contained in the node $\mathcal{NN}$: $\{\mathcal{N} \,|\, \mathcal{N} \in \mathcal{T}, \mathcal{N} \in \mathcal{NN}\}$, where the induced elemental "region" $R_\mathcal{N}$ for a given node $\mathcal{N}$ is defined as $R_\mathcal{N} = \{p \,|\, p \in \mathcal{N}, p \notin C(\mathcal{N})\}$, $C(\mathcal{N})$ denotes all the direct children of the node $\mathcal{N}$. A such example of second attribute function $\mathcal{AA}$ is depicted in Figure 3.4.

The Figure 3.3 gives an example of $\mathcal{TT}$. It is a Min-tree of $\mathcal{T}$. The four regional minima of the graph $(G_\mathcal{T}, F_\mathcal{A})$ are presented as four leaves (red circles) of $\mathcal{TT}$. The second attribute $\mathcal{AA}$ in this example is the height of $\mathcal{A}$.

3.2.3   Second tree filtering

The second tree filtering is performed based on the second attribute $\mathcal{AA}$, which is properly designed to be increasing. The second tree filtering is a pruning based on comparing $\mathcal{AA}$ to a given threshold. Let us remark that depending on the application, two different pruning strategies can be used. For the purpose of filtering out some non-desired shapes, the nodes to be pruned are the subtrees rooted just above the leaves. In the case of selecting the shapes corresponding to the desired shapes (represented by the leaves of $\mathcal{TT}$), the pruning strategy removes the nodes, that are closer to the root node of $\mathcal{TT}$. It is equivalent to preserving the subtrees containing the leaves.

The second tree in Fig. 3.3 is filtered by pruning the nodes $\mathcal{NN}$ whose attribute value $\mathcal{AA}$ (the height of $\mathcal{A}$) is less than 2. This is an example of filtering the

(a)   (b)
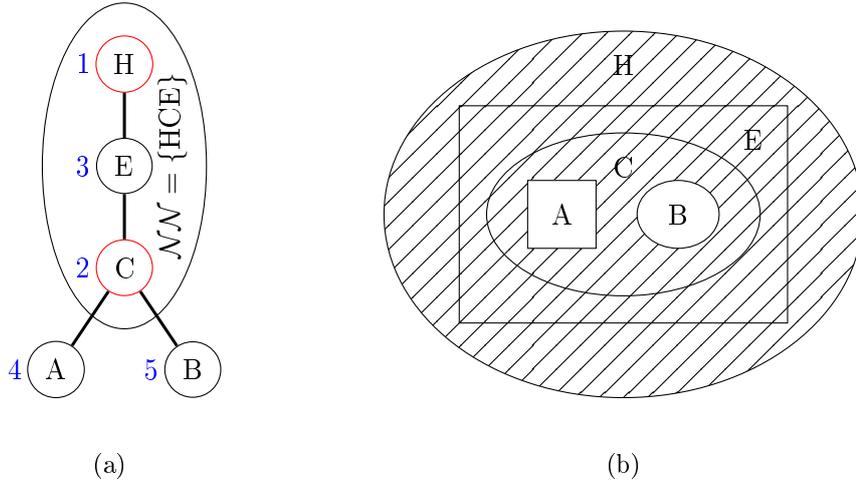
Figure 3.4: Illustration of second attribute $\mathcal{AA}$ being the total variation inside the region given by the union of element "region" of shapes contained in a node $\mathcal{NN}$. (a) A branch of the tree in Figure 3.3, the nodes inside the ellipse form the node $\mathcal{NN} = \{\text{HCE}\}$ in the second tree $\mathcal{TT}$. (b) The underlying region of image represented by the branch of tree $\mathcal{T}$ in (a), the total variation inside the region covered with oblique lines correspond to $\mathcal{AA}(\mathcal{NN})$.

non-desired shapes around the leaves.

Let us mention also another interesting variant of the second tree filtering strategy. It is based on the extinction value [Vachier 1995] on the basis of the increasing attribute $\mathcal{AA}$. If for example, the interesting shapes corresponding to nodes being minima of $\mathcal{A}$ (a context-based energy estimator [Xu 2012] (see also Chapter 7) and the Number of False Alarms (NFA) [Cao 2005] are such instances) on the graph $(G_{\mathcal{T}}, F_{\mathcal{A}})$, and we want to keep the shapes which are very likely to be interesting ones. The second tree $\mathcal{TT}$ will be the Min-tree. Let $\prec$ be a strict total order on the set of minima $m_1 \prec m_2 \prec \cdots \prec m_n$ in a decreasing order of significance, such that $m_i \prec m_{i+1}$ whenever $\mathcal{A}(m_i) < \mathcal{A}(m_{i+1})$. The extinction value $\mathcal{E}$ for a minimum m_i is defined by:

$$\mathcal{E}(m_i) = \mathcal{AA}(\mathcal{NN}_{m_i}^{m_i-}) - \mathcal{AA}(\mathcal{NN}_{m_i}), \tag{3.1}$$

where $\mathcal{NN}_{m_i}$ is the leaf of $\mathcal{TT}$ corresponding to the node m_i of the graph $(G_{\mathcal{T}}, F_{\mathcal{A}})$, and $\mathcal{NN}_{m_i}^{m_i-}$ denotes the lowest node on $\mathcal{TT}$ containing $\mathcal{NN}_{m_i}$ that does not contain any leaf $\mathcal{NN}_{m_{i'}}$ with $i' < i$.

The extinction value of a minimum measures the significance of this minimum. The filtering strategy based on the extinction values is to preserve (or remove) only the blobs determined by a minimum whose extinction value is higher than a given
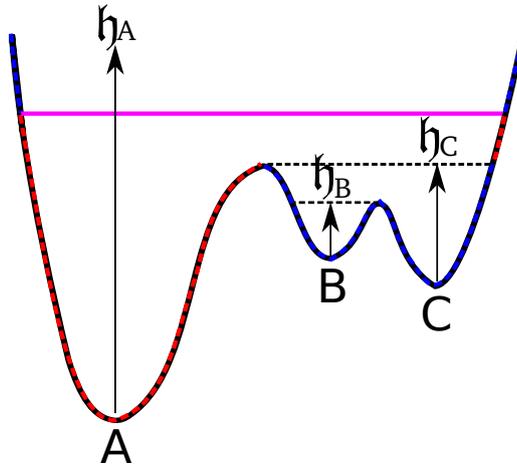
Figure 3.5: Illustration of the extinction values of three minima. A, B, and C are three local minima, and $\mathfrak{h}_m$ denotes the extinction value of height for the corresponding minimum m.

value λ and whose attribute $\mathcal{A}\mathcal{A}$ is also below λ. The advantage of this strategy is that it preserves only these shapes which are meaningful enough compared to their context. For example, the red part in Fig. 3.5 is preserved for the value given by the purple line, but the blobs corresponding to minima B and C are removed. In this example, $\mathcal{A}\mathcal{A}$ is the height and the extinction value of the height is also known as the dynamic of a minimum.

3.2.4 Tree restitution

The step of tree restitution is trivial. The simplified tree $\mathcal{T}'$ is reconstructed by removing the set of nodes $\{\mathcal{N}_i\}$ contained in the series of filtered nodes $\{\mathcal{N}\mathcal{N}_k\}$. For example, the simplified tree $\mathcal{T}'$ in Fig. 3.3 is obtained by removing the nodes (dashed circles) contained in filtered nodes of $\mathcal{T}\mathcal{T}'$. Note that, in this example, the nodes C and H are removed, while the node E lying between them in the same branch is preserved. None of the existing pruning strategies described in Section 2.5.2 can achieves such a result. Indeed, the nodes E and J with $\mathcal{A} = 3$ are preserved, while the node F with $\mathcal{A} = 3$, and even the node D with $\mathcal{A} = 4$ are filtered out. Such a behavior cannot be obtained with a threshold-based strategy. In fact, shape-based morphology is more flexible than conventional connected operators and brings some new possibilities.

3.3   Definition of MSER on the shape spaces

3.3.1   A point of view of MSER using the shape space

As discussed in Section 2.6.2.1 and as shown in [Donoser 2006], the MSER is easily understandable using the tree-based shape spaces given by the Min-tree and Max-tree representations. Here we will provide a novel point of view of MSER in spirit of corner detection.

The corners in images are robust features having many invariant or covariant properties. Corner detection is frequently used in many applications of computer vision for its robustness and reasonable number of points, such as motion detection, image registration, image mosaicing, panorama stitching, video tracking, 3D modelling, and object recognition. The Harris corner detector [Harris 1988] is one of the popular corner detection method. It extracts the corner points in the space of image by finding extrema of a corner measure based on the second moment matrix at some fixed scale. Such a process can be seen as first of all define a searching space (*e.g.*, all points of the space of image for corner detection), then define some robust measurement upon which the decision of robust points selection is made. The measurement should satisfy many invariant or covariant properties, such as, invariant to viewpoint changes, contrast changes, scale changes, blur changes, . . . , the measurement based on the second moment matrix satisfies some of these invariances. Then the decision based on this measurement should also be made under some invariances, the choice of extrema is a appropriate one.

In the same spirit, the MSER is in accord with the principle of corner detection methods. Whereas, in the case of MSER, the searching space is the tree-based shape space given respectively by the Min-tree and Max-tree representation. This shape space is a reduced searching space compared with the number of points of the space of image. The measurement is stability function (see Section 2.6.2.1 and Eq (3.2)) defined on each region of in the space. This stability function is only based on the area variation, which satisfies most of the invariant properties mentioned above. The decision of the selection of robust regions in the space of shapes is conducted by the local minima extraction, which is similar with the extrema choice in the case of corner detection.

Both the corner detection and MSER extraction are performed by selecting robust features in the space of features being respectively the space of image and the space of shapes. However, in the case of corner detection, a multi-scale corner detection is usually required to make it robust to scale changes. A scale-adapted Harris corner detector and its extension Harris-Laplace with scale selection by finding extrema of the Laplacian of Gaussian (LoG) filter were proposed in [Mikolajczyk 2004].

The corner detection belongs to the family of scale-space based local feature detection. It is also well-known that the corner detection methods usually extract several corner points that correspond actually to a real single corner in image. Whereas, the MSER defined in the tree-based shape space is already multi-scale thanks the multi-scale property of the tree-based image representations and the invariance to scale changes of the stability function. Besides, the stable regions extracted by MSER provide a compact local path (elliptic fitting of the real regions in shape space) upon which some descriptor [Lowe 1999, Lowe 2004, Forssen 2007] is computed to establish the correspondence between a pair of images. In fact, the tree-based shape space is a multi-scale image representation, it is invariant to affine transformation of image intensities, covariant to adjacency preserving transformation, all these properties make the tree-based shape space well suited for local feature detection, which contributes to the main reason of the success of MSER. The stability function is certainly also an important reason that preserves these invariance properties of the tree-based shape space, and it contributes also to the success of MSER. However, note that due to the fact that the stability function is based on a parameter related to image contrast, the MSER is not truly contrast invariant. It is reported in [Mikolajczyk 2005] that MSER outperforms the corner detection in term of repeatability test [Mikolajczyk 2005], in the case of viewpoint changes, scale changes, and contrast changes.

A more detailed comparison between scale-space based local feature detectors and shape-space based local feature detectors will be given in Section 4.2.1.

3.3.2 Matching the tree-based shape spaces

MSER is a blob detection method, it extracts stable (usually contrasted) regions. In practical use, the centroid of the regions extracted by MSER in the shape space of an image is used as the interest points based on which the correspondence between a pair of images is established. Such correspondence matching is achieved through the point descriptors [Lowe 1999, Lowe 2004, Forssen 2007] computed upon a local patch (usually an elliptic region) around the points, the distance between the descriptors of points in two images is then used to estimate the correspondence. Such a classical process (*i.e.*, keypoints detection + descriptors + correspondence estimation) is widely used in many applications.

Unlike the corner detection methods, for which the relationship between the keypoints in the space of images is unclear, the structure in the space of image around a keypoint is trivial, and the searching space is too large, the shape space provides a reduced searching space. More interestingly, the regions extracted in the shape space by MSER obey the nested property reported in Eq (2.15), *i.e.*,

two regions of MSER are either disjoint or one is included in the other one, the structure around a MSER in the shape space (*i.e.*, a point in the shape space) is rich in terms of local information contents (*e.g.*, the number of children). These later two interesting properties are obtained thanks to the edges (tremendously reduced compared with the edges in the space of images) in the shape space that reflect the inclusion relationship of neighboring regions in the shape space. However, these two interesting properties are not utilized in the classical scheme using MSER. It might be interesting to make use of these properties by replacing the descriptors computed upon a local patch around the keypoints (centroid of MSERs) by the local structures around the MSERs in the shape space. In another word, the local structure around a MSER R_s in the shape space is served as a descriptor describing/characterizing that MSER R_s. This proposition is equivalent to match directly the underlying tree-based shape spaces of a pair of images based on some located points (*i.e.*, MSERs) of that shape space. This is somehow in the same spirit with the work of Pan *et al.* [Pan 2009b] where the authors a preferential image segmentation method based on the use of tree-based shape space given by the topographic map representation (see Section 2.6.1.4 and [Pan 2009b]) for more details.

The matching of two tree-based shape spaces seen as graphs is in general a NP-hard problem. But we only would like to match a tremendously reduced interesting nodes extracted by MSER. This might help to reduce the complexity of graph matching. Besides, there exist some efficient graph matching optimization techniques, such as the work of Torresani *et al.* [Torresani 2008]. The proposition of matching directly the tree-based shape space is still a open question that we did not exploit. We have tried the tree-based shape space matching through another interesting nodes spotting method which will be detailed in Chapter 5. It is a simplification method based on minimizing the piecewise Mumford-Shah functional [Mumford 1989] that extracts few salient shapes. A such based shape space matching is illustrated in Figure 3.6, the underlying tree representation is the topographic map.

3.3.3   Extending MSER

Using the shape space given by a tree-based image representation $\mathcal{T}$ (either a Min-tree or Max-tree), the MSER is defined as the local minima of the attribute function given as follows:

$$\mathcal{A}_q(\mathcal{N}) = \frac{(|\mathcal{N}_\Delta^+| - |\mathcal{N}_\Delta^-|)}{|\mathcal{N}|}, \tag{3.2}$$
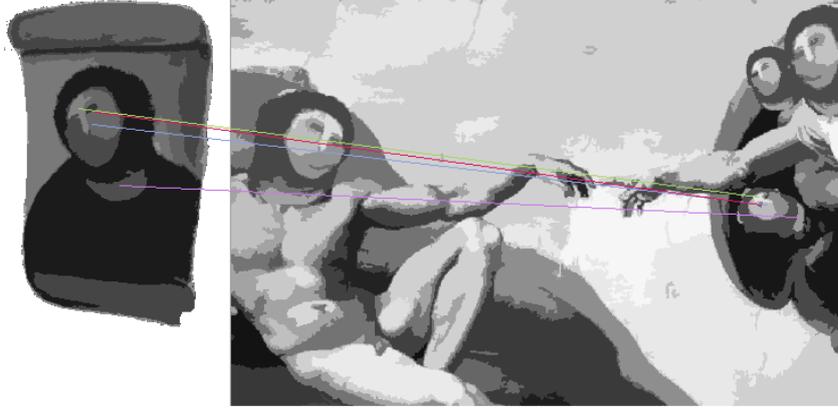
Figure 3.6: Feature correspondence via the tree-based shape space matching. The shape space is given by the topographic map representation, the keypoints (*i.e.*, interesting regions) in the shape space is extracted by a simplification method detailed in Chapter 5.

where $|\cdot|$ denotes the cardinality, $\mathcal{N}_\Delta^+$ and $\mathcal{N}_\Delta^-$ are respectively the lowest ancestor and the highest descendant such that

$$|f(\mathcal{N}_\Delta^+) - f(\mathcal{N})| \geq \Delta, \tag{3.3}$$

$$|f(\mathcal{N}) - f(\mathcal{N}_\Delta^-)| \geq \Delta, \tag{3.4}$$

where $|\cdot|$ denotes the absolute value. Note that the lowest ancestor node $\mathcal{N}_\Delta^+$ is unique, whereas the highest descendant node $\mathcal{N}_\Delta^-$ may not be unique, because a bifurcation node might be present below the node $\mathcal{N}$, in this sense the stability function in Eq (3.2) is not quite well defined. One popular choice is to chose the biggest highest descendant $\mathcal{N}_\Delta^{-*}$ that satisfies Eq (3.4), defined as follows:

$$\mathcal{N}_\Delta^{-*} = \operatorname{argmax}(|\mathcal{N}_\Delta^{-i}|), \, i = 1, \ldots, n, \tag{3.5}$$

where $\{\mathcal{N}_\Delta^{-i} \,|\, i = 1, \ldots, n\}$ is the set of disjoint descendants (*i.e.*, in different branches of the tree $\mathcal{T}$) that obey Eq (3.4). The public implementation in OpenCV is a such instance, where a small modification of Eq (3.2) is also applied, defined by:

$$\mathcal{A}_q'(\mathcal{N}) = \frac{(|\mathcal{N}| - |\mathcal{N}_\Delta^{-*}|)}{|\mathcal{N}|}, \tag{3.6}$$

An variant stability function is used in the public implementation of VLFeat [Vedaldi 2008], where the area variation is defined using only the node itself and its unique lowest ancestor which obeys Eq (3.3), it is given by:

$$\mathcal{A}_q''(\mathcal{N}) = \frac{(|\mathcal{N}_\Delta^+| - |\mathcal{N}|)}{|\mathcal{N}|}, \tag{3.7}$$

Note also that the stability function make use of a parameter that depends on the contrast of image, which make it not truly invariant to local contrast changes. The evolution of the stability function of Eq (3.2) along a branch starting from a leaf (left side) to the root (right side) of the Min-tree for a simple image is given in Figure 3.7 (a) ($\Delta = 10$) and (c) ($\Delta = 2$). Figure 3.7 (b) and (d) are respectively the zoomed curves of (a) and (c) by removing the nodes whose values (axis Y) are too large, in order to better visualize the local minima of the stability function. Note that the number of children depicted in Figure 3.7 (f) shows that almost all the nodes of that branch have only one child, so the local minima shown in (b) and (d) are actually local minima in the shape space, that correspond to MSERs. However, from Figure 3.7 (b), we can see there are actually many MSERs that are very similar, a clean up step is necessary to regroup them. In fact, in the both public implementation: OpenCV and VLFeat [Vedaldi 2008], the MSERs being too small or too large are removed, the MSERs having a big stability function values are filtered, and the MSERs that are similar in terms of position and size are regrouped. These processes can be seen as a filtering in the shape space based on the attribute function of Eq (3.2). From this point of view, the framework of shape-based morphology can be used to extend the MSER. More importantly, as discussed in Section 3.3.1, the main reason of the efficiency of MSER is that the shape space given by Min-tree and Max-tree representations have many invariant properties and the stability function (*i.e.*, a special example of the attribute function) preserves those invariant properties when it is used to extract the critical regions in an invariant/covariant way by spotting its local minima. Following this principle, any shape space given by a tree representation having those invariant properties and any attribute function that preserves those properties can be used. The minima or maxima of the attribute function gives also the interesting regions in a invariant/covariant way for a pair of images. A such example is the average of gradient's magnitude along the boundary of a region in the shape space, it is given by:

$$\mathcal{A}_\nabla(\mathcal{N}) = \big(\sum_{e \in \partial\mathcal{N}} \mathrm{grad}(e) \big)/|\partial\mathcal{N}|, \tag{3.8}$$

where $\partial\mathcal{N}$ denotes the boundary of the region represented by $\mathcal{N}$, $|\partial\mathcal{N}|$ denotes the length of the boundary, and $\mathrm{grad} : E \to \mathbb{R}$ is an image of gradient's magnitude. The local maxima of this attribute function $\mathcal{A}_\nabla$ extract the interesting regions in the shape spaces. The evolution of the attribute function $\mathcal{A}_\nabla$ for a simple image is depicted in Figure 3.7. There are also many local maxima that correspond to similar interesting regions, a filtering step in the shape space is also required. However, note that the interesting regions extracted by using the stability function of MSER in Eq (3.2) and by using the attribute function in Eq (3.8) are quite similar but

not identical. But since the centroid of the interesting regions is used to estimate the transformation between a pair of image, the extracted interesting regions from the shape space are robust to small local geometrical deformation, which is one of the many advantages of shape-space based local feature detectors. A comparison of MSER and the one using the attribute function in Eq (3.8) will be shown in Chapter 4.

The MSER extracts usually contrasted regions in the shape space, which results in a small number (but reasonable) of extracted regions. This limits the application of MSER. For instance, in spite of the accuracy and efficiency of MSER, for the applications such has image registration, image mosaicing, panorama stitching, and 3D modelling, where a high number of features is required, the corner detection is usually more preferred than MSER. Although lowering considerably MSER stability margin gives much more points that would probably be interesting. However it creates numerous local minima of the stability function, many of them represent similar regions (see Figure 3.7 (d)), merging those minima is unsound. A topology based interesting regions extraction method that we call Tree-Based Morse Regions (TBMR) will be detailed in Chapter 4. It relies on the Morse theory [Milnor 1963], and no attribute function is required, only the structure of the shape space given by Min-tree and Max-tree is needed. Numerous advantages will be presented including a much more important number of extracted features.

3.4 Filtering

Connected operators are filtering tools that act by merging the flat zones. The framework of shape-based morphology is proposed by expanding the idea of tree-based connected operators. Filtering is one direct consequence of the framework, which we will discuss in this section. Note that the filters disposed with the framework of shape-based morphology belong always to the family of connected operators thanks to the use of shape-spaces given by tree-based image representations.

3.4.1 More possibilities and flexibilities

As reviewed in Section 2.5.2, the existing tree-based connected operators treat the regions in the shape space individually. The decision of tree filtering is made without making use of the inclusion relationship between those regions. We propose to apply the connected operators in the tree-based shape space to make up that issue and to expect a more robust tree filtering decision.

In fact, as we will show in the sequel, for the filtering purpose, the framework of shape-based morphology encompasses some usual attribute filtering strategies, and

(a)

(b)

(c)

(d)

(e)

(f)
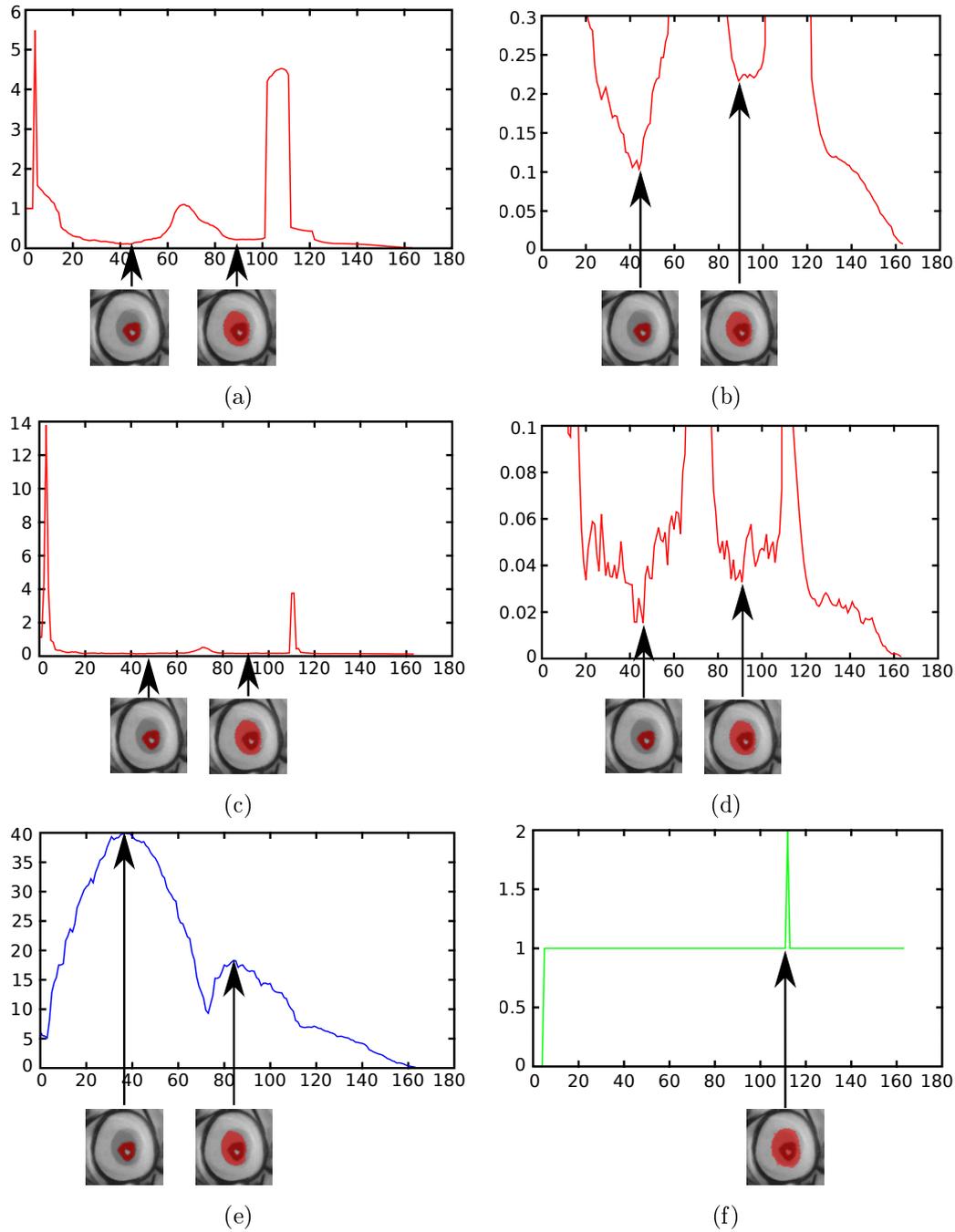
Figure 3.7: Evolution of stability function of Eq (3.2) and average of gradient's magnitude en Eq (3.8) of a branch starting from a leaf node (left side of Axis X) to the root node (right side of Axis X). Axis Y represents the value of corresponding attribute function. The images on the bottom of each figure represents the corresponding regions represented by the nodes at the corresponding locations in the shape space. (a-b) Normal and zoomed stability function with $\Delta = 10$; (c-d) Normal and zoomed stability function with $\Delta = 2$; (e) Average of gradient's magnitude along the boundaries. (f) Number of children.

two new class of connected operators based on the non-increasing shape attributes $\mathcal{A}_s$ are introduced. Indeed, as shown in Figure 3.3 and discussed in Section 3.2.4, the shape-based morphology can give some filtering results that is impossible by the existing tree filtering strategies.

3.4.2 Encompassing the classical attribute filters

In the most trivial case, the attribute $\mathcal{A}$ is increasing, and the classical connected filterers are equivalent to a pruning of the tree.

Proposition 1 *If $\mathcal{A}$ is increasing, let $\mathcal{TT}$ be the Min-tree built from the tree-based shape space seen as a node weighted graph $(G_{\mathcal{T}}, F_{\mathcal{A}})$ induced by $\mathcal{T}$ and $\mathcal{A}$, then $\mathcal{TT}$ is isomorphic to $\mathcal{T}$.*

Proof: Since $\mathcal{A}$ is increasing, so for any given node $\mathcal{N}_k$, $\mathcal{A}(\mathcal{N}_k) \leq \mathcal{A}(\mathcal{N}_a)$ holds for any ancestor node $\mathcal{N}_a$ of $\mathcal{N}_k$, which means that the leaves of $\mathcal{T}$ are regional minima on the corresponding node weighted graph $(G_{\mathcal{T}}, F_{\mathcal{A}})$. These regional minima lie also on the leaves of $\mathcal{TT}$ being the Min-tree. Furthermore, as the adjacency (graph edge) of the graph $G_{\mathcal{T}}$ represents the parenthood, for any pair of neighboring vertices $(\mathcal{N}_i, \mathcal{N}_j)$ in $G_{\mathcal{T}}$, either $\mathcal{N}_i = \mathrm{parent}(\mathcal{N}_j)$ or $\mathcal{N}_j = \mathrm{parent}(\mathcal{N}_i)$. Suppose that the former one holds, then $\mathcal{N}_j \subset \mathcal{N}_i, \mathcal{A}(\mathcal{N}_j) \leq \mathcal{A}(\mathcal{N}_i) \Leftrightarrow F_{\mathcal{A}}(\mathcal{N}_j) \leq F_{\mathcal{A}}(\mathcal{N}_i)$. Let $\mathcal{N}_k \in \mathcal{T}_{\mathcal{N}}$ be any neighboring vertex $(\neq \mathcal{N}_i)$ of $\mathcal{N}_j$, since the parent of each node on a tree structure is unique, so $\mathcal{N}_k$ is a child of $\mathcal{N}_j$, which means $F_{\mathcal{A}}(\mathcal{N}_k) \leq F_{\mathcal{A}}(\mathcal{N}_j)$. In consequence, we have $\mathcal{NN}_i = \mathrm{parent}(\mathcal{NN}_j)$, where $\mathcal{NN}$ denotes the first node in the second tree that contains the vertex $\mathcal{N}$ of the node weighted graph $(G_{\mathcal{T}}, F_{\mathcal{A}})$. So $\mathcal{TT}$ is isomorphic to $\mathcal{T}$.

Let $\mathcal{AA}$ be the current level of the second tree $\mathcal{TT}$, which means $\forall \mathcal{NN}_k \in \mathcal{TT}, \mathcal{AA}(\mathcal{NN}_k) = \mathcal{A}(\mathcal{N}_k)$. Pruning $\mathcal{TT}$ is equivalent to pruning $\mathcal{T}$. In other words, the shape-based morphology encompasses the classical filtering strategy in this case, but we do not have to test whether the attribute $\mathcal{A}$ is increasing or not.

3.4.3 Encompassing attribute thresholding strategy

A shape attribute $\mathcal{A}$ is more often non-increasing. In such a case, there exists some pair of vertices $(\mathcal{N}_i, \mathcal{N}_j)$ composing an edge in the graph $G_{\mathcal{T}}$, such that $\mathcal{N}_j = \mathrm{parent}(\mathcal{N}_i)$, and $\mathcal{A}(\mathcal{N}_j) \leq \mathcal{A}(\mathcal{N}_i) \Leftrightarrow F_{\mathcal{A}}(\mathcal{N}_j) \leq F_{\mathcal{A}}(\mathcal{N}_i)$. For example, Let $\mathcal{TT}$ be the Min-tree of the graph $(G_{\mathcal{T}}, F_{\mathcal{A}})$, then the node $\mathcal{NN}_i$ is hence an ancestor of $\mathcal{NN}_j$ on $\mathcal{TT}$. So $\mathcal{TT}$ is different from $\mathcal{T}$. Furthermore, just like the increasing attribute case, let $\mathcal{AA}$ be the current level of $\mathcal{TT}$, we have $\mathcal{AA}(\mathcal{NN}_i) = \mathcal{A}(\mathcal{N}_i)$. Pruning $\mathcal{TT}$ on the basis of $\mathcal{AA}$ is equivalent to thresholding $\mathcal{T}$. Accordingly, the

shape-based morphology encompasses the threshold-based strategies (*Direct* and *Subtractive* rules). But let us also remark that it is nevertheless impossible to retrieve the same pruning strategies as the classical ones (*Min*, *Max*, and *Viterbi*) described in Section 2.5.2.

The second attribute $\mathcal{AA}$ can be different from $\mathcal{A}$; for example, it can be any measure based on $\mathcal{A}$ or even some new attribute/measure computed from the image domain (*e.g.*, the total variation inside the context region represented by the node $\mathcal{NN}$). This is when the shape-based morphology becomes different. In general, two new classes of connected operators will be introduced. The first class is defined from the leveling family, and is named *shape-based lower/upper levelings* (see Section 3.4.4). The second class is the *self-dual morphological shapings* (see Section 3.4.5).

3.4.4 Shape-based lower/upper levelings

We will now detail the first type of novel connected operators given by the shape-based morphology, in the case of $\mathcal{T}$ being a Max-tree or Min-tree representation of the image f.

Proposition 2 *If the tree $\mathcal{T}$ is a Max-tree, the shape-based morphology gives an upper leveling, named as shape-based upper leveling $\psi_{s\uparrow}$.*

Proof: Let $\mathcal{T}$ be a Max-tree representation, then no matter what type of tree $\mathcal{TT}$ is and no matter how $\mathcal{TT}$ is pruned, the simplified tree $\mathcal{T}'$ has always a Max-tree structure in the sense that gray level for the ancestors is always lower. In the image reconstruction step, the pixels stored in some removed node $\mathcal{N}_r$ take the gray level of the first preserved ancestor $\mathcal{N}_a$ (*Direct* rule) or even lowered with the change induced by those removed ancestors (*Subtractive* rule). Anyway, $\forall x \in V, \psi_{s\uparrow}(f)(x) \leq f(x)$ always holds. By Definition 6, such an operator $\psi_{s\uparrow}$ is a upper leveling.

Proposition 3 *If the tree $\mathcal{T}$ is a Min-tree, the shape-based morphology gives a lower leveling, named as shape-based lower leveling $\psi_{s\downarrow}$.*

Proof: Let $\mathcal{T}$ be a Min-tree, the simplified tree $\mathcal{T}'$ is still a Min-tree in the sense that the gray levels of the ancestors are always higher. So, as the same as the proof in the case of $\mathcal{T}$ being a Max-tree, $\forall x \in V, \psi_{s\downarrow}(f)(x) \geq f(x)$ holds. According to Definition 5, such an operator $\psi_{s\downarrow}$ is a lower leveling.

Effectively, the fact that shape-based morphology in the case of $\mathcal{T}$ being a Max-tree (resp. Min-tree) results in a shape-based upper (resp. lower) leveling is because any anti-extensive (resp. extensive) operator is a upper (resp. lower) leveling. Let us also remark that such an operator ψ_s is not a leveling, since either $f(x) \geq \psi_s(f)(x)$

or $\psi_s(f)(x) \geq f(x)$ holds for any vertex of the graph, therefore, $\exists x, y \in V$, $f(x) \geq \psi_s(f)(x)$ and $\psi_s(f)(y) \geq f(y)$ do not hold in the same time. So the operator ψ_s is not a leveling except in the case that $\psi_s(f)$ is constant.

The classical upper (resp. lower) leveling removes the details of the regional minima (resp. maxima). In practice, it is equivalent to pruning the Min-tree (resp. Max-tree) with an increasing attribute $\mathcal{A}^\uparrow$. Nevertheless, the shape-based upper (resp. lower) leveling is based on some connected filterings on the shape space built from the Max-tree (resp. Min-tree) representation. It filters out the details of unwanted bright (resp. dark) shapes on the basis of the user defined non-increasing shape attributes.

Proposition 4 *If the second tree $\mathcal{TT}$ filtering is idempotent, then the shape-based upper leveling $\psi_{s\uparrow}$ and lower-leveling $\psi_{s\downarrow}$ are idempotent.*

Proof: Let $\mathcal{T}_0$, $\mathcal{TT}_0$, $\mathcal{TT}_0'$, and $\mathcal{T}_0'$ be the tree structures corresponding to the shape-based morphology ψ_s applied to f. $\mathcal{T}_1$, $\mathcal{TT}_1$, $\mathcal{TT}_1'$, and $\mathcal{T}_1'$ are the tree structures corresponding to ψ_s applied to $f' = \psi_s(f)$. As $\mathcal{T}_0'$ and $\mathcal{T}_1$ are a Max-tree or a Min-tree, it is trivial that $\mathcal{T}_1 = \mathcal{T}_0'$, thanks to the strict fixed order of the gray level between neighboring nodes of those trees. As the second tree filtering is based on the pruning strategy, it is equivalent to remove some blobs around minima or maxima of the graph $(G_\mathcal{T}, F_\mathcal{A})$. As a consequence, the second tree of $(G_{\mathcal{T}'}, F_\mathcal{A})$ is the same as $\mathcal{TT}'$, which means that $\mathcal{TT}_1 = \mathcal{TT}_0'$. The idempotent second tree filtering yields: $\mathcal{TT}_1' = \mathcal{TT}_0' \Rightarrow \mathcal{T}_1' = \mathcal{T}_0'$, so $\psi_s(\psi_s(f)) = \psi_s(f)$ holds.

The preservation of the blobs based on the use of extinction value is an example of an idempotent second tree filtering. The blobs of the minima or maxima are preserved, so the extinction values for the minima or maxima of $(G_{\mathcal{T}'}, F_\mathcal{A})$ are still higher than the given value. Hence the blobs will remain with any additional filtering.

3.4.5 Morphological shapings

Unlike the shape-based upper or lower leveling, which deals only with bright or dark shapes, we introduce in this section a second type of novel connected operators which process both bright and dark shapes at the same time.

Proposition 5 *If $\mathcal{T}$ is a self-dual representation, i.e., a topographic map, and $\mathcal{A}$ is a non-increasing attribute, the operator ψ_s given by the shape-based morphology is not a leveling.*

Proof: Since the shape attribute $\mathcal{A}$ is non-increasing, and the shape-based morphology yields a $\mathcal{T}'$ which is not a pruning of $\mathcal{T}$. So there could exist a pair of

neighboring vertices (x, y), such that $\mathcal{N}_x \subset \mathcal{N}_y$, and $\mathcal{N}_y$ is removed while $\mathcal{N}_x$ is preserved. The nodes $\mathcal{N}_x$ and $\mathcal{N}_y$ denote the smallest shapes that contain respectively x and y. Let $\mathcal{N}_z$ be the lowest preserved ancestor of $\mathcal{N}_y$. Then we have $\psi_s(f)(x) = f(x)$, $\psi_s(f)(y) = f(z)$. However, as the tree $\mathcal{T}$ is the topographic map, the order of the gray level of a node and any of its ancestor is not monotonous a priori. So it is possible that $f(z) < f(x)$ and $f(z) < f(y)$. In this case, we have $\psi_s(f)(x) > \psi_s(f)(y)$, $f(x) \geq \psi_s(f)(x)$ and $\psi_s(f)(y) < f(y)$. It is contradictory with the definition of leveling (see Eq (7)).

Definition 9 *If $\mathcal{T}$ is a topographic map, the connected operator defined by the shape-based morphology is called a* morphological shaping *denoted by $\mathcal{S}$.*

The name "shaping" comes from the fact that such operator acts by removing some unwanted dark and bright shapes and preserving some desired shapes.

Proposition 6 *The morphological shaping $\mathcal{S}$ is a self-dual operator.*

Proof: Since the tree $\mathcal{T}$ is self dual, which means $\mathcal{T}^+$ representing f has the same structure as $\mathcal{T}^-$ representing $-f$. And the attribute $\mathcal{A}$ is shape attribute being independent to the gray level. So $\forall, \mathcal{N}_k^+ \in \mathcal{T}^+$, $\exists \mathcal{N}_k^- \in \mathcal{T}^-$, such that $\mathcal{N}_k^+ \Leftrightarrow \mathcal{N}_k^-$ in the sense that the two nodes represent the same connected component and so $\mathcal{A}^+(\mathcal{N}_k^+) = \mathcal{A}^-(\mathcal{N}_k^-)$. In consequence, the graph $(G_{\mathcal{T}^+}, \mathcal{A}^+)$ is equal to the graph $(G_{\mathcal{T}^-}, \mathcal{A}^-)$. Then we have $\mathcal{T}\mathcal{T}^+ = \mathcal{T}\mathcal{T}^- \Rightarrow \mathcal{T}'^+ = \mathcal{T}'^-$ and $f' = -f'$, which means $\mathcal{S}(-f) = -\mathcal{S}(f)$. So the shaping $\mathcal{S}$ is a self-dual operator.

To make the self dual shapings $\mathcal{S}$ idempotent, the first condition is $\mathcal{T}_1 = \mathcal{T}_0'$. However, this requirement is not trivial due to the unfixed order of gray level of the neighboring nodes on the topographic maps $\mathcal{T}$. For any node $\mathcal{N}_k \in \mathcal{T}$, it is given by the holes filling of connected components either from upper level sets (see Eq (2.4)), or from lower level sets (see Eq (2.7)). Let us denote this nature of $\mathcal{N}_k$ respectively by $\mathcal{N}_<^k$ or $\mathcal{N}_>^k$.

Proposition 7 *Let $\mathcal{T}$ be a topographic map, for each preserved node $\mathcal{N}_k$ of $\mathcal{T}'$, if $f'(\mathcal{N}_k) < f'(parent'(\mathcal{N}_k))$ holds for $\mathcal{N}_k$ being $\mathcal{N}_<^k$, and $f'(\mathcal{N}_k) > f'(parent'(\mathcal{N}_k))$ holds for $\mathcal{N}_k$ being $\mathcal{N}_>^k$, then the tree $\mathcal{T}_1$ constructed from f' is equal to $\mathcal{T}'$.*

Proof: Thresholding the reconstructed image f' by Eq (2.4) and Eq (2.7) yields some shapes which can be also found in $\mathcal{T}'$. This correspondence is guaranteed by the condition $\forall \mathcal{N}_<^k \in \mathcal{T}', f'(\mathcal{N}_<^k) < f'(\text{parent}'(\mathcal{N}_<^k))$ and $\forall \mathcal{N}_>^k \in \mathcal{T}', f'(\mathcal{N}_>^k) > f'(\text{parent}'(\mathcal{N}_>^k))$. According to the definition of the topographic map [Monasse 2000b], the inclusion relationship between those shapes yields a unique tree structure, which is equal to $\mathcal{T}'$.

Proposition 8 *If the second tree filtering $\mathcal{T}\mathcal{T}$ is idempotent and the nature of the preserved nodes does not change, then the shaping $\mathcal{S}$ is idempotent.*

    Proof: See Proposition 7 and the proof of proposition 4.

3.5   Object detection/segmentation

3.5.1   Main idea: local minima correspond to meaningful objects

The tree-based shape space provides a tremendously reduced searching space for object detection and segmentation, and it is a multi-scale representation. All these motivate us to perform object detection and segmentation tasks using that space. The work of Vilaplana [Vilaplana 2008] shortly reviewed in Section 2.6.2.6 is a such instance.

    Suppose that for a given image, we compute a shape space given by one of the tree-based image representations $\mathcal{T}$ presented in Section 2.3, such that the interesting objects that one would like to detect are represented by some nodes in that tree $\mathcal{T}$. The question is how to retrieve those interesting objects. Similar with the filtering tasks, we could assign an attribute function $\mathcal{A}_o$ to each node. For the purpose of object detection/segmentation tasks, the attribute function $\mathcal{A}_o$ can be some measurement that characterizing the importance/meaningfulness of the region represented by a node $\mathcal{N}$, such as the meaningfulness of the boundary $\partial\mathcal{N}$ proposed in [Cao 2005] by the number of false alarms (NFA), the average of gradient's magnitude along the boundary (see Eq (3.8)), or the compactness. If the underlying tree is created by means of region merging algorithms (*e.g.*, BPT), the attribute $\mathcal{A}_o$ could also be some assessment that show the reluctancy of a merge resulting that node $\mathcal{N}$. If the form, the color, or the position of the interesting objects to be detected are are known a priori, the attribute $\mathcal{A}_o$ can also be some specific designed assessment measuring how much a node fits the a priori knowledge.

    The choice of the tree representation and the design of attribute function $\mathcal{A}_o$ as well as their computation are kind of preparation work, based on which the decision of objects detection is made. Once these information is available, the object detection/segmentation task is achieved by objects spotting in the shape space (*i.e.*, searching space). The most trivial spotting strategy is to spot the "most likely" one. It is useful if there is only one interesting object in the image. However, in most cases, the number of interesting objects is unknown, and is usually more than one. In this case, one possibility is to first of all spot the "most likely" node $\mathcal{N}_1^*$ among all the regions in the shape space, then disable all the nodes belonging to the branches that contains $\mathcal{N}_1^*$ (*i.e.*, the descendants of the node $\mathcal{N}_1^*$ as well as the
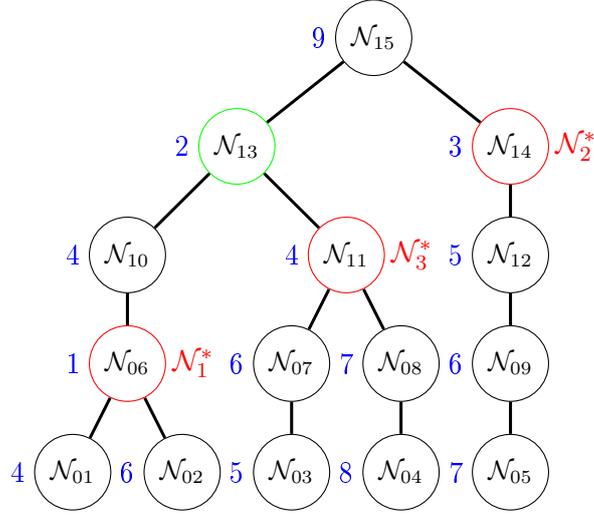
Figure 3.8: Object spotting by repeating the process of selecting the "most likely" node $\mathcal{N}_i^*$ and disabling the ancestors and descendants of node $\mathcal{N}_i^*$. Blue values are corresponding attribute value $\mathcal{A}_o$. The three nodes represented by red circles are the detected objects, where $\mathcal{N}_1^*$ is firstly spotted, then $\mathcal{N}_2^*$ and $\mathcal{N}_3^*$. Note that the green node $\mathcal{N}_{13}$ is more meaningful than $\mathcal{N}_{11}$, but it is not spotted.

ancestors of $\mathcal{N}_1$). Then retrieve a second "most likely" node $\mathcal{N}_2^*$ among the rest regions in the shape space, and disable again its relative nodes (*i.e.*, its descendants and its ancestors). This "most likely" node spotting and the descendants as well as the ancestors disabling process is repeated until all the nodes are disabled. In consequence, a set of nodes $\{\mathcal{N}_i^*, \mid i = 1, \ldots, n\}$ will be detected, where the number of detected objects is decided by the algorithm. A such objects spotting process is depicted in Figure 3.8. This spotting strategy might give interesting results in some cases, but it ignore the fact that several interesting objects may be present in a same branch of $\mathcal{T}$, which means one is included in another. For example the node $\mathcal{N}_{06}$ and node $\mathcal{N}_{13}$ in Figure 3.8.

The notion of "most likely" is usually modeled by the extremum of the attribute function $\mathcal{A}_o$. Following this idea, we propose to spot the local extrema of the attribute function $\mathcal{A}_0$ as interesting objects. For the sake of simplicity, we propose to use the local minima, if it is the local maxima of $\mathcal{A}_o$ which are interesting, then use $1/\mathcal{A}_o$ so that the local maxima become local minima. Using this strategy, the regions that are represented by nodes in the same branch can be spotted meanwhile. For instance, in Figure 3.8, the three local minima of attribute function $\mathcal{A}_o$ in the shape space are respectively $\mathcal{N}_{06}$, $\mathcal{N}_{13}$, and $\mathcal{N}_{14}$, where nodes $\mathcal{N}_{06}$ and $\mathcal{N}_{13}$ are in the same branch.
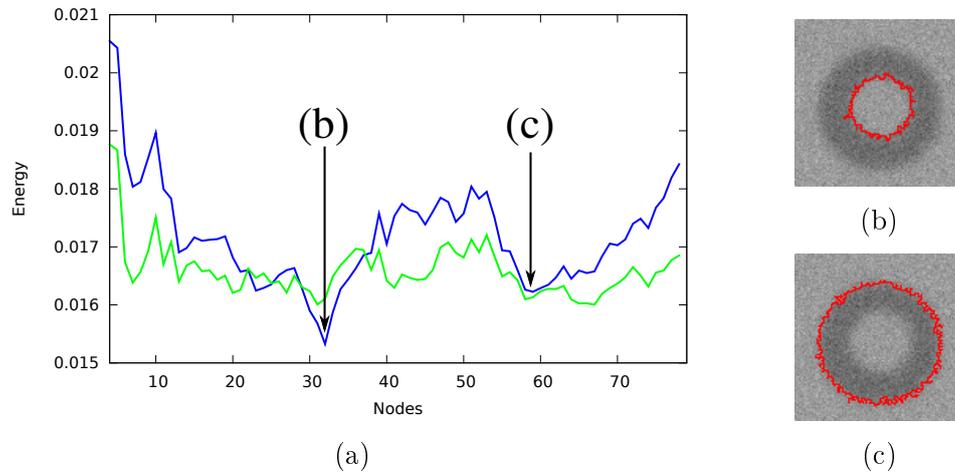
Figure 3.9: (a): Evolution of two attribute functions $\mathcal{A}_o$ along a branch starting from a leaf (left side of Axis X) to the root (right side of Axis X). These two attribute functions $\mathcal{A}_o$ are both based on the meaningfulness of the region boundaries: snake energy [Kass 1988] is depicted in green, and context-based energy estimator [Xu 2012] (see also Chapter 7) is in blue. (b) and (c): the boundaries of two regions corresponding to significant energy minima in the shape space given by the topographic map.

3.5.2 Problem: many meaningless local minima

The idea of spotting local minima in the shape space as meaningful objects is in spirit similar with the principle of MSER discussed in Section 2.6.2.1 and Section 3.3. As same as the stability function $\mathcal{A}_q$ of Eq (3.2), there are usually many local minima of the attribute function $\mathcal{A}_o$, two examples of the attribute function $\mathcal{A}_o$ based on the meaningfulness of the region boundaries are depicted in Figure 3.9. Many of the local minima correspond actually to meaningless objects (*e.g.*, the local minima between the two significant minima (b) and (c), as well as the local minima at the beginning and the end in Figure 3.9), some local minima represent some regions that are very similar, and only a representative one should be detected (*e.g.*, several local minima around (b) and (c)).

3.5.3 Connected filtering in shape spaces

It is rather difficult to design a perfect attribute function $\mathcal{A}_o$ in the sense that all its local minima correspond to meaningful objects. Usually an attribute function $\mathcal{A}_o$ having many local minima is obtained, and only those significant minima correspond to the interesting objects to be spotted. We need to filter the meaningless local
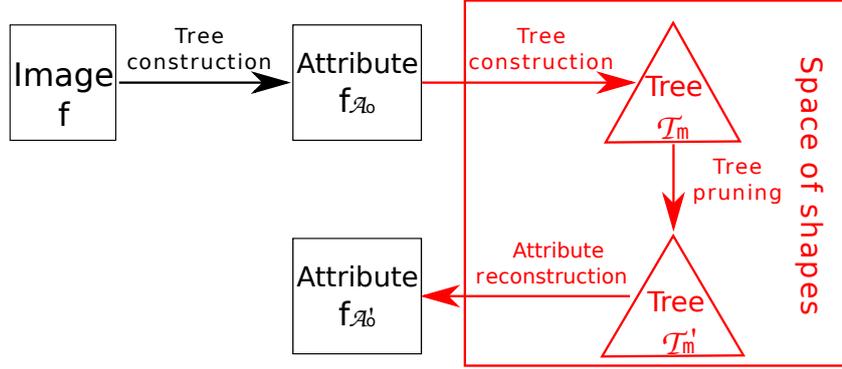
Figure 3.10: Tree pruning based connected filtering applied in the shape space. The first tree-based image representation along with the attribute function $\mathcal{A}_o$ is seen as an image $f_{\mathcal{A}_o}$ defined subordinated to the shape space. The second tree representation build from the image of attribute $f_{\mathcal{A}_o}$ is a Min-tree $\mathcal{T}_m$.

minima in the shape space. This could be achieved by applying a connected filter to the tree-based shape space seen as a nodes weighted graph. In fact the attribute function $\mathcal{A}_o$ can be seen as an image defined on the space of shapes given by a tree representation $f_{\mathcal{A}_o} : \mathcal{N} \to \mathbb{R}$. Then apply a classical tree-based connected filter on the image $f_{\mathcal{A}_o}$. The underlying tree of this connected filtering process is the Min-tree representation $\mathcal{T}_m$, a pruning of this Min-tree is well know as a local minima killer, then from the pruned Min-tree $\mathcal{T}'_m$, we can reconstruct a filtered attribute function $\mathcal{A}'_o$ of which the minima correspond to the meaningfully interesting objects. Many meaningless local minima of $\mathcal{A}_o$ are thus filtered. Note that the local minima of $\mathcal{A}'_o$ are usually flat zones of the shape space, and a flat zone of local minima might contain several local minima of $\mathcal{A}_o$ which correspond to several meaningful objects being very similar. We propose to select the region having the smallest $\mathcal{A}_o$ as the representative one for that flat zone of local minima. An example is given in Figure 3.11. This connected filtering process applied in the shape space is depicted in Figure 3.10, which is equivalent to first of all transform a given image f to an image of attribute function $f_{\mathcal{A}_o}$ (the preparation work discussed in Section 3.5.1), then followed by a classical tree pruning based connected filter. This scheme is equivalent to discard the last image restitution step and change the way of tree restitution step in the framework of shape-based morphology depicted in Figure 3.2.

By augmenting the pruning force, more and more local minima of $\mathcal{A}_o$ will be filtered or absorbed by the close local minima having a smaller attribute value (the case of flat zone of local minima in filtered attribute function $\mathcal{A}'_o$). This filtering force can be measured by the notion of extinction value (see Section 3.2.3 and [Vachier 1995])
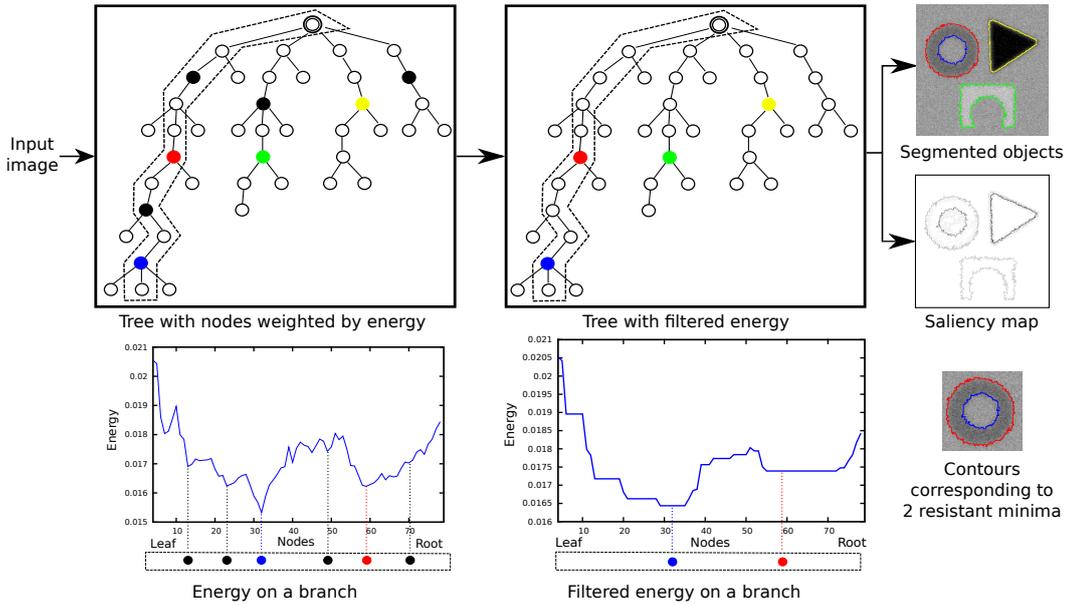
Figure 3.11: An example of the object detection scheme by spotting local minima of an attribute function $\mathcal{A}_o$ as meaningful objects. The underlying tree representation is the topographic map, and $\mathcal{A}_o$ is the context-based energy estimator (see Chapter 7). Filled circles: local minima; Colorized filled circles: resistant local minima after connected filtering in the shape space. Top right: detected meaningful objects surrounded by the colorized contour and a hierarchy of object detection result; Bottom: Evolution of attribute $\mathcal{A}_o$ and filtered attribute $\mathcal{A}'_o$ along the branch surrounded by dashed contours in the tree.

defined for those local minima. Consequently, a soft object detection result can be obtained through a hierarchy using the extinction value. This idea will be detailed in Section 3.6.

An example of such described method for object detection/segmentation is illustrated in Figure 3.11. In this example, the underlying tree representation is the topographic map. The attribute function $\mathcal{A}_o$ is the context-based energy estimator which will be detailed in Chapter 7. The filled circles represent the local minima of $\mathcal{A}_o$ in the topographic map, and the colorized ones correspond to the four meaningful objects in the image whose boundaries are colorized with the corresponding color in the output result. The local minima represented by the black filled circles will be filtered by the connected filtering in the shape space. A hierarchy of object detection represented by a saliency map is also depicted in the output of this object detection scheme.

3.6   Transformation of hierarchy

By expanding the idea of the scheme of object detection through the framework of shape-based morphology, when we increase the filtering force in the shape space, the minima disappear one by one, so less and less local minima are spotted as interesting objects. In this sense, each local minimum has a certain possibility to be identified as interesting object. This possibility reveals the meaningfulness of that object represented by a local minimum. As discussed in Section 3.2.3 and in Section 3.5.3, this meaningfulness is measured by the extinction value defined on the local minima. But how to represent the extinction values defined for those nodes being local minima in a shape space is not straightforward. In fact, this question leads to a more general question: how to better visualize and better understand a tree-based shape space. As discussed in Section 2.4.1, a tree-based shape space $S_{\mathcal{T}}$ is an equivalent image representation, it is a multi-scale representation, and an image is easier to interpret through the tree-based shape space $S_{\mathcal{T}}$. $S_{\mathcal{T}}$ can be seen as an intermediary level representation. But how can we perceive these advantages directly in the same way as one observe for example a 2D image f itself. An example is depicted in Section 9.5 through an efficient algorithm of disjoint level lines selection, from a simplified image f' reconstructed from these disjoint level lines, the main structure of topographic map (*i.e.*, tree structure yielded by all the level lines presented in image) can be easily perceived. This method can be extended to other tree representations. But still the attribute function $\mathcal{A}$ characterizing some interesting feature of the regions in the shape space $S_{\mathcal{T}}$ is not considered. Whereas, the intensity value of each point in the space of image could be easily observed through the image f itself.

In this section, we will first introduce how to better perceive a tree-based shape space as well as its associated attribute function $\mathcal{A}$. This is achieved by first of all locating a set of important nodes decided by the attribute function characterizing some interesting region feature, and by weighting the attribute function $\mathcal{A}$ instead of the intensity value to a new created image that we call attribute map $\mathcal{M}_{\mathcal{A}}$. From this attribute map, the most important information carried by the shape space can be easily perceived. Then we will detail a special case of the attribute map that represents the extinction value of the local minima of a shape space. It will be shown that it is in fact a kind of new saliency map which is equivalent to a hierarchy of image segmentation.
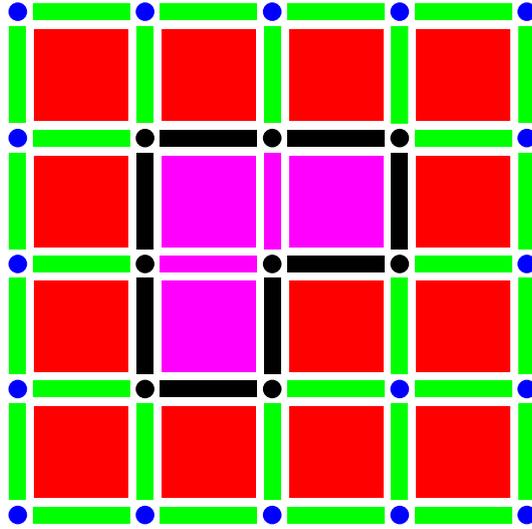
Figure 3.12: Materialization of the points in a 2D image with 0-faces (blue disks), 1-faces (green strips), and 2-faces (red squares). The original points of the image is the 2-faces, the boundaries are materialized with 0-faces and 1-faces. The contour of the purple region is composed of black 1-faces and 0-faces.

3.6.1 Attribute map

We distinguish two types of attribute map depending on weighting the attribute function $\mathcal{A}$ to the regions itself or to the region boundaries. Note that a region boundary is composed of a set of elements that lie in between points, the elements are materialized by 1-faces and 0-faces for a 2D image as depicted in Figure 3.12. For instance, the contour of the purple region having three pixels is composed of the black 1-faces and black 0-faces.

- **Attribute map $\mathcal{M}_{\mathcal{A}}^{\bullet}$ defined on regions**

The idea of attribute map $\mathcal{M}_{\mathcal{A}}^{\bullet}$ defined on regions is to weight the attribute function $\mathcal{A}$ to each point in the space of image. More specifically:

1) Initialize the attribute map $\mathcal{M}_{\mathcal{A}}^{\bullet}$ with 0, the size of $\mathcal{M}_{\mathcal{A}}^{\bullet}$ is the same as the original image.

2) For each point $p \in V$, find the set of connected components of the tree $\mathcal{T}$ that contain this point p, it is a finite set of regions starting from the first node $\mathcal{N}_p$ including p to the root node of the tree $\mathcal{T}$. $\mathcal{M}_{\mathcal{A}}^{\bullet}(p)$ is the attribute of the representative region among the set of ancestor nodes. The representative region can be obtained by some means such as, selection the most meaningful one, or the closest (*i.e.*, lowest) meaningful one.

Such defined attribute map $\mathcal{M}_\mathcal{A}^\bullet$ is useful for better visualize the tree structure as well as the attribute function. It is also useful in some real applications, such as the scale map of local scale measure for remote sensing images proposed by Luo *et al.* [Luo 2009], where the representative node is selected by the closest meaningful region subordinated to some criterions, and for instance, if the attribute function is a learned hierarchical features [Farabet 2013], the representative node could be selected using their proposed automatic node retrieval technique. The produced attribute map $\mathcal{M}_\mathcal{A}^\bullet$ can be used for final scene labeling in their method.

This attribute map $\mathcal{M}_\mathcal{A}^\bullet$ is somehow similar in spirit with the work of differential are profiles proposed by Ouzounis *et al.* [Ouzounis 2012a], where for each point p, a vector representing the differential area along the set of regions from $\mathcal{N}_p$ to the root is used instead of a scalar value being the attribute of some selected node.

- **Attribute map $\mathcal{M}_\mathcal{A}^\circ$ defined on boundaries**

The attribute map $\mathcal{M}_\mathcal{A}^\circ$ defined on boundaries is to weight the attribute function $\mathcal{A}$ to each 1-face and 0-face composing the region boundaries. More precisely:

1) Initialize the attribute map $\mathcal{M}_\mathcal{A}^\circ$ with 0, note that the size of $\mathcal{M}_\mathcal{A}^\circ$ is doubled compared with the original image f.

2) For each 1-face $e \in E$ (an element lying in between a pair of neighboring points (x, y)) in the image, find the set of connected components on the tree $\mathcal{T}$ whose boundary contains e. Note that the set of connected components are represented by one (or two) set(s) of successive nodes starting respectively from the first node $\mathcal{N}_x$, $\mathcal{N}_y$ that contains respectively x, y. if $\mathcal{N}_x \cap \mathcal{N}_y \neq \emptyset$, then either $\mathcal{N}_x \subseteq \mathcal{N}_y$ or $\mathcal{N}_y \subseteq \mathcal{N}_x$. Suppose that the former inclusion holds. Then the set of nodes are $\{\mathcal{N}_x, \ldots, \mathcal{N}_y(\text{not included})\}$. Whereas if $\mathcal{N}_x \cap \mathcal{N}_y = \emptyset$, in such a case, let $\mathcal{N}_a$ be the lowest common ancestor of $\mathcal{N}_x$ and $\mathcal{N}_y$. Then $\mathcal{N}_a$ is also the first node such that $x \in \mathcal{N}_a$ and $y \in \mathcal{N}_a$. The set of nodes are $\{\mathcal{N}_x, \ldots, \mathcal{N}_a(\text{not included})\} \cup \{\mathcal{N}_y, \ldots, \mathcal{N}_a(\text{not included})\}$. Note also that the set of connected components might be empty in the case of $\mathcal{N}_x = \mathcal{N}_y$. $\mathcal{M}_\mathcal{A}^\circ(e)$ is decided by the smallest attribute $\mathcal{A}_m$ (suppose that smaller attribute $\mathcal{A}(\mathcal{N})$ is, the region represented by $\mathcal{N}$ is more meaningful) of the set of connected components. Then $\mathcal{M}_\mathcal{A}^\circ(e)$ is given by the reverse of $\mathcal{A}_m$, such as $\mathcal{M}_\mathcal{A}^\circ(e) = 1/\mathcal{A}_m$. Besides, if $\mathcal{N}_x = \mathcal{N}_y$ which implies that the set of regions is empty, then $\mathcal{M}_\mathcal{A}^\circ(e) = 0$.

3) For each 0-face o in the image, $\mathcal{M}_\mathcal{A}^\circ(o) = \max\{\mathcal{M}_\mathcal{A}^\circ(e) \,|\, o \text{ is neighboring to } e\}$.

The attribute map $\mathcal{M}_{\mathcal{A}}^{\circ}$ defined on the region boundaries is inspired from the saliency map [Najman 1996] obtained by stacking a family of hierarchical contours. In fact, an attribute map $\mathcal{M}_{\mathcal{A}}^{\circ}$ represents a hierarchy of segmentation, each threshold of $\mathcal{M}_{\mathcal{A}}^{\circ}$ gives a partition of the image. Compared with the attribute map $\mathcal{M}_{\mathcal{A}}^{\bullet}$ defined on the regions, it is more adapted to handle the nested meaningful regions thanks to the following property:

$$\mathcal{N}_1 \subset \mathcal{N}_2 \nRightarrow \partial\mathcal{N}_1 \subset \partial\mathcal{N}_2, \tag{3.9}$$

in fact, if $\mathcal{N}_1 \subset \mathcal{N}_2$, the boundaries of their represented regions $\partial\mathcal{N}_1$ and $\partial\mathcal{N}_2$ can be totally disjoint.

The attribute map $\mathcal{M}_{\mathcal{A}}^{\circ}$ defined on region boundaries either transforms a morphological tree (*i.e.*, Min-tree, Max-tree, or topographic map) to a hierarchy of segmentation, or transforms a hierarchy of segmentation to another one. They are both guided by the attribute function $\mathcal{A}$ which is computed on a set of multi-scale regions that are not too local. Which might be more interesting than the initial one. However, as depicted in Figure 3.7 and Figure 3.9, around the meaningful nodes, many close nodes are also relative meaningful, so there are lots of close boundaries having similar meaningfulness in the attribute map $\mathcal{M}_{\mathcal{A}}^{\circ}$, which make the real meaningful ones not so visible. This effect will be illustrated in Chapter 5.

3.6.2 Saliency map using extinction value

In order to enhance the visibility of the meaningful regions, we propose a variant of the attribute map defined on region boundaries: the saliency map using extinction values (see Section 3.2.3) $\mathcal{M}_{\mathcal{E}}$. Recall that the extinction value $\mathcal{E}$ is defined on the local minima, for a given minimum m_i of an attribute function $\mathcal{A}$ in the shape space, $\mathcal{E}(m_i)$ is defined as the maximal morphological filtering force for which m_i is still a local minimum (the representative one in the case of a flat zone of local minima) of the filtered attribute function $\mathcal{A}'$. In fact, this saliency map using extinction value is inspired from the scheme of object detection, for which the basic idea is that the meaningful objects in the image correspond to local minima of some attribute function $\mathcal{A}_o$ characterizing the meaningfulness of each node. Consequently, we can discard the nodes which are not local minima of $\mathcal{A}_o$ in the shape space, because the important information is carried by the local minimal nodes. Moreover, the importance of those local minima can be measured by their extinction values $\mathcal{E}$. The saliency map based on the extinction value $\mathcal{M}_{\mathcal{E}}$ is defined by weight the extinction value $\mathcal{E}$ instead of the attribute function $\mathcal{A}$ to region boundaries. More precisely, $\mathcal{M}_{\mathcal{E}}$ is obtained by changing the step 2) in the scheme of attribute map $\mathcal{M}_{\mathcal{A}}^{\circ}$ computation:
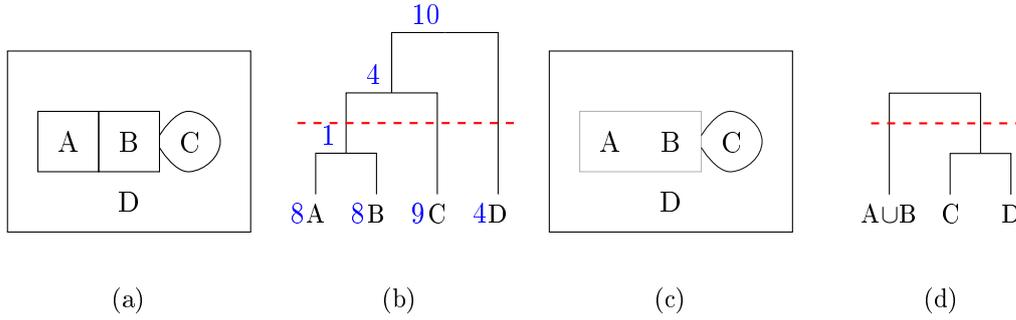
Figure 3.13: An example of hierarchy transformation through the saliency map using extinction value $\mathcal{M}_{\mathcal{E}}$. (a) Original image; (b) Original hierarchy H, blue numbers are the attribute values; (c) Saliency map using extinction value $\mathcal{M}_{\mathcal{E}}$, the boundary of region A∪B is weighted with 9, greater than the weight (being 6) of boundary of C, they are given by respectively the extinction value of the minimum A∪B and of the minimum D; (d) Dendrogram of $\mathcal{M}_{\mathcal{E}}$, and a cut of this dendrogram is given by the red dashed line.

2) The value for each element e of the saliency map using extinction value $\mathcal{M}_{\mathcal{E}}(e)$ is given by the maximal extinction value of the minima among the set(s) of nodes representing the set(s) of nested components having e as an element of its boundary. Certainly, if there is no minimum among the set of nodes, $\mathcal{M}_{\mathcal{E}}(e) = 0$.

Each threshold of this map $\mathcal{M}_{\mathcal{E}}$ represents an object detection/segmentation result. This saliency map using extinction value $\mathcal{M}_{\mathcal{E}}$ represents a hierarchy of segmentation computed from a tree-based shape space given by either a morphological tree (*i.e.*, Min-tree, Max-tree or topographic map) or a hierarchy of segmentation. The computation is guided by the attribute function $\mathcal{A}$. Note that a segmentation given by a cut of $\mathcal{M}_{\mathcal{E}}$ might be different from any cut of the original shape space $S_{\mathcal{T}}$. For instance, in Figure 3.13, the region C merge with the union of regions A and region B in the original hierarchy H depicted in Figure 3.13 (b), the union of region C and region D will never be a single region of a partition given by any cut of the original hierarchy H. However, the union of region C and region D is a possible single region of a partition given by thresholding the saliency map using extinction value $\mathcal{M}_{\mathcal{E}}$ as depicted in Figure 3.13 (c). Indeed, the partition given by the cut of $\mathcal{M}_{\mathcal{E}}$ depicted in Figure 3.13 is composed by two regions: region A∪B and region C∪D.

Some applications of the saliency map using extinction value will be depicted in Chapter 5 and Chapter 8.

3.7 Conclusion

In this chapter, we have presented the main proposition in this thesis: the concept of shape-based morphology framework. It is a versatile framework that deals with region-based image representations. Such representations are used in a large number of applications. Our framework is inspired from the connected operators. We propose to apply them on the tree-based shape space underlying any tree-based image representation. The framework of shape-based morphology makes use of the complete tree structure, *i.e.*, the inclusion relationship between neighboring regions present in the tree. Such information can be seen as a regional context whose shape is adapted to the image contents. That helps to take a more robust filtering decision. We have shown that this framework is more general than the classical existing methods. It can be used for three types of processing tasks. 1) For shape filtering, we have shown that it encompasses the classical pruning- and threshold-based strategies. Two novel classes of connected operators are introduced. Namely the shape-based lower/upper levelings, and the shapings. 2) This framework can be applied to object detection/segmentation, and a soft object detection/segmentation is obtained via a saliency map representation with no additional cost. 3) We have also shown that this framework is useful to compute a saliency map representing a hierarchical image segmentation from any tree-based image representation. All these aspects demonstrate the high potential of the proposed framework of shape-based morphology.

Part III

Applications

Tree-Based Morse Regions (TBMR)

This chapter introduces a topological approach to local invariant feature detection motivated by Morse theory. We use the critical points of the graph of the intensity image, revealing directly the topology information as initial "interest" points, and then associate to each point a local patch selected from the shape space. The critical points coincide with the leaves and nodes having bifurcation on the Max-tree and Min-tree. For each critical point, we finally extract the largest region that contains it and is topologically equivalent in its tree. We call such regions the Tree-Based Morse Regions (TBMR). They can be seen as a variant of MSER, which are contrasted regions. TBMR relies only on topological information and hence extracts the regions independently of the contrast, which makes it truly contrast invariant and quasi parameters free. TBMR extraction is fast, having the same complexity as MSER. Experimentally, TBMR achieves a repeatability on par with state-of-the-art methods, but obtains a significantly higher number of features. Its accuracy and robustness are demonstrated by applications to image registration and 3D reconstruction.

4.1 Introduction

Local invariant feature detection [Matas 2002, Tuytelaars 2004, Lindeberg 1998, Lowe 2004, Schmid 2000, Mikolajczyk 2005, Moreels 2007, Aanæs 2012] is an important step in a number of applications such as wide baseline matching, object and image retrieval, tracking, recognition, image registration and 3D reconstruction. The classical process to obtain the features consists in detecting a specific class of interest points, such as corners, together with an associated scale generally obtained from a scale-space. Typical examples of such key locations are the local extrema of the result of difference of Gaussians (DoG) applied in scale-space to a series of smoothed and resampled images. Several crucial invariance properties are required for using such points in applications, such as invariance to image translation, scaling, and rotation, to illumination changes or to local geometric distortion.

(a) Four among 76 used multi-view images.

(b) Incomplete reconstructed 3D facades using DoG.

(c) The four facades of the 3D reconstruction using TBMR.

Figure 4.1: An example of 3D reconstruction using local invariant features. Top: 4 among 76 used multi-view images. Four facades of the PMVS [Furukawa 2010] densified sparse 3D reconstruction from the SfM pipeline [Moulon 2012] using DoG (middle: almost no points on the front roof, and the back facade of the building is missing.) and the proposed TBMR (down: the whole tour is reconstructed).

In this Chapter, we propose a topological approach to extract the local invariant features. We first extract some initial "critical" points, based on ideas from the Morse theory [Milnor 1963]: minima, maxima and saddle points. More precisely, following [Caselles 2009], we propose to choose critical nodes in the two trees (called Min-tree and Max-tree [Salembier 1998]) made by the connected components of lower and upper level sets: those critical nodes are the leaves and the nodes having several children. For each critical node a scale is selected. Instead of using a scale-space, the scale comes from the tree-based shape space: we associate to a critical node $\mathcal{N}_c$ the largest region containing it and topologically equivalent in its tree. We call our method Tree-Based Morse Regions (TBMR).

TBMR has several main advantages: as it uses only topological information, it is independent on the image contrast. It is also covariant to continuous (topological) transformations such as translation, scaling or rotation. As demonstrated in this Chapter, it is also robust to local geometric distortion. Furthermore, it is essentially parameter-free: only two non-significant parameters are applied, so that we ignore regions that are either too small or too large. And last, but not the least, efficient algorithms with a quasi-linear or a linear complexity are available to compute it [Salembier 1998, Najman 2006, Nistér 2008].

In Section 4.4.1, some qualitative results compared with other state-of-the-art methods will be illustrated to show the better distribution of TBMR. Quantitative evaluation, based on the image coverage measurement in Section 4.4.2, confirms the qualitative evaluation. Tests in Section 4.4.3 demonstrate that TBMR achieves repeatability score comparable to other state-of-the-art methods with a significantly higher number of correspondences. We evaluate TBMR on two applications in which many matched features are required: image registration (Section 4.4.4) and 3D reconstruction (Section 4.4.5). For these two applications, and as illustrated in Figure 4.1, results attest that TBMR improves over the commonly used DoG [Lowe 1999].

4.2 Related work

We focus on two detector classes, those based on scale-space and those based on MSER. For a complete review of invariant feature detectors, the interested reader is referred to Tuytelaars and Mikolajczyk's survey [Tuytelaars 2008].

4.2.1 TBMR versus scale-space feature detection

There exists a variety of local invariant feature detectors having relatively good performance, as assessed by several evaluation frameworks [Schmid 2000,

Mikolajczyk 2005, Moreels 2007, Aanæs 2012].   The first type is based on scale-space. Harris corners, Hessian based detectors and the Difference of Gaussians (DoG) are such instances. The Harris corner detector [Harris 1988] finds the extrema of a corner measure based on the second moment matrix at some fixed scale. A scale-adapted Harris corner detector and its extension Harris-Laplace [Mikolajczyk 2004] with scale selection find extrema of the Laplacian of Gaussian (LoG) filter. The Hessian detector [Lindeberg 1998] extracts the extrema of a feature measure based on the Hessian matrix. Its extension Hessian-Laplace [Mikolajczyk 2004] uses the same scale selection as Harris-Laplace. The affine versions of both Harris and Hessian are based on the affine shape estimation using the second moment matrix. Harris based detectors tend to extract corner-like structures, while Hessian based detectors tend to find blobs and ridges. DoG [Lowe 1999] is similar to the Hessian detector in the sense that it approximates LoG by the trace of the Hessian matrix. DoG tends to extract points at isotropic blob structures.

In spirit, TMBR is very similar to these kinds of approaches. TBMR detects critical points (i.e., extrema and saddle points), but it does not rely on a scale-space. As described in section 4.3, it uses the shape space given by the Min-tree and Max-tree representations, as discussed in Section 2.4, this space having the main property of scale-space, namely the causality principle [Koenderink 1984].

4.2.2   TBMR versus MSER

As discussed in Section 3.3, although its original definition is quite different, the Maximally Stable Extremal Regions (MSER) [Matas 2002] is easily understand-able using Min-tree and Max-Tree. As shown in [Donoser 2006] and Section 3.3, MSER extracts the regions (nodes) that correspond to local minima of a stability function along the path to the root of the tree. The stability function of a given node is given by the difference between the area of some (grand-)parent and some (grand-)child, divided by the area of the node itself. It is defined by Eq (3.2) It is reported [Mikolajczyk 2005] that MSER achieves state-of-the-art repeatabilities and regions accuracies. It is also very efficient. However, the number of detected features are comparatively small which limits its ability for some applications like image registration and 3D reconstruction. Perdoch et al. [Perdoch 2007] propose the Stable Affine Frame (SAF) for which only local stability is required. Many more features are obtained with a comparable repeatability score. However, it is much slower than MSER.

TBMR can be seen as a variant of MSER, both relying on Min/Max-tree rep-resentations. The most fundamental difference is related to illumination change, a very common effect in natural images that is reported as an unsolved problem

in the literature [Aanæs 2012]. Indeed, the MSER stability function depends on a parameter Δ that fixes the intensity level difference of the (grand-)parent and of the (grand-)child actually used for the ratio. That prevents a true invariance of MSER to illumination change. By contrast, TBMR, being purely topological, is truly invariant to affine illumination change. A less fundamental difference concerns the number of parameters of MSER. As TBMR, MSER uses two parameters to remove too large and too small regions. But MSER also requires in the stability function, on top of the parameter Δ we just described, a threshold to remove unstable regions, and another parameter to group together detected regions that are similar in terms of position and size. Such parameters are not needed in TBMR. A last minor difference deals with the definition of MSER. Indeed, the stability function is not clearly defined in the presence of bifurcations, i.e. when a node has more than one child. That raises a difficulty in trying to reproduce some results: for example, there exist two public implementations of MSER, one from VLFeat [Vedaldi 2008], the other from OpenCV, each one of them using a similar but different stability function. The topological definition of TBMR allows for a perfect reproducibility, whatever the chosen algorithm implementation.

4.3 TBMR extraction

In this section, we describe our proposed topology-based local invariant features detector called the *Tree-Based Morse Regions (TBMR)*. The TBMRs are extracted from the shape space built from the image by the Max-tree $\mathcal{T}_M$ and Min-tree $\mathcal{T}_m$. In Section 4.3.1, we will show how to extract those "interest" regions from the shape space based on Morse theory [Milnor 1963].

4.3.1 Feature extraction based on Morse theory and shape space

Properties of the tree-based image representation make the shape space (see Section 2.4) very appropriate for local invariant features detection. MSER extracts the "stable" regions using a stability function. Doing so, it does not preserve properties such as the contrast invariance. In order to preserve all the invariance of the tree, we propose a topological approach that detects "interest" regions based on Morse theory [Milnor 1963]. Recall that a Morse function is a smooth function f whose critical points (i.e., points where $\nabla f = 0$) are isolated. Critical points are minima, maxima and saddle points of f. The topology of f is directly linked to the analysis of those critical points. However, the Morse function is not an adequate model for an image, as it prevents the existence of plateaux for example. A consequence is that we will deal with *regional* extrema and saddle points with regional extrema and
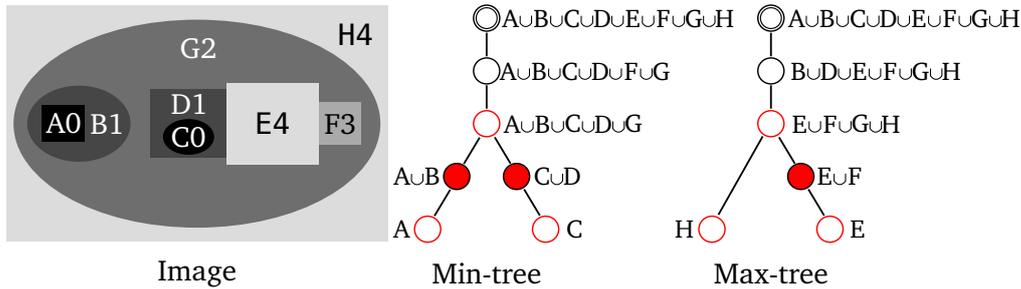
Figure 4.2: A synthetic image and the corresponding Min-tree (middle) and Max-tree (right) representation. The critical nodes are represented by red circles: (1) nodes having more than one child, and (2) leave-nodes. The filled nodes are the TBMRs.

saddle *regions* instead of isolated points.

The use of Morse theory is not new in computer vision: see, for example, the contour tree [Kweon 1994, Van Kreveld 1997] and the Reeb graph [Reeb 1946, Takahashi 1995] for shape matching. Here, we use the Max- and Min-trees to locate the critical points. To do that, we rely on the following property, that can be deduced from Caselles and Monasse [Caselles 2009, Chapter 4]:

Proposition 9 *The critical points of a Morse function are the extrema of f, corresponding to the leaves of the Max-tree $\mathcal{T}_M$ and of the Min-tree $\mathcal{T}_m$ of f, and the saddle points of f, corresponding to nodes of these trees with several children.*

We thus call *critical nodes* the leaves and the nodes of $\mathcal{T}_M$ and $\mathcal{T}_m$ with more than one child. Having critical nodes rather than critical points allows us to deal with any function, even if it is not a Morse one. In Fig. 4.2, critical nodes are highlighted with a red circle.

The next stage is to associate a scale to each critical node. A critical node corresponds to a change of topology in its tree: either an apparition (leaves corresponding to extrema) or a merge (nodes with several children). Thus, on a branch of the tree between two critical nodes, there is no topological change in the tree. In other words, a node that is not critical is topologically equivalent to the first critical node we encounter going from the node to the leaves of the tree. Conversely, and as a node corresponds to a region of the image, a critical region/node $\mathcal{N}_c$ is topologically equivalent to any region/node that contains $\mathcal{N}_c$ but no other disjoint critical region/node. A good "scale" choice for representing a critical region/node is thus the largest region/node to which it is topologically equivalent. We call such a region a *Tree-based Morse Region* (TBMR). A reason to take the *largest* topologi-
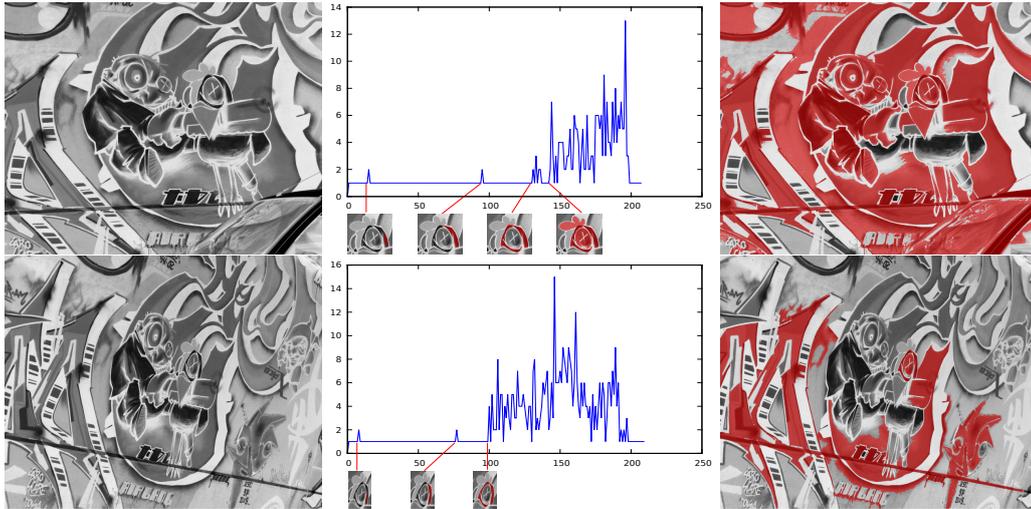
Figure 4.3: An example of TBMR extraction. Left: Front view (top) and 30° view (down) of "Graffiti" scene [Mikolajczyk 2005]. Middle: Evolution of number of children starting from a leaf to the root and some extracted TBMRs along the branch; Right: first TBMR containing the TBMRs illustrated in middle column (though they are actually too large).

cally equivalent node in the tree is that we want as much context as reasonable to encode the region.

In practice, we do not consider TBMRs that are either too small or too big. This small regions discarding is performed before analyzing the tree structure, which means the small regions do not contribute to the topological changes of the tree structure. Discarding them eliminates also some noise without modifying other components. Also, regions that meet the image border are considered truncated and we ignore them. In Fig. 4.2, TBMRs are drawn with a red disk. The evolution of the number of children of a node, starting from a leaf to the root is illustrated in Figure 4.3.

As in most shape-space based methods, we compute the centroids of the selected regions as the final feature points. The ellipse with the same first and second moments as the detected region is then used as the local patch upon which a descriptor is computed.

4.4 Results

Qualitative and quantitative comparison of the distribution of TBMR with other popular local feature detectors are illustrated in Section 4.4.1 and Section 4.4.2. In

section 4.4.3, the repeatability assessment of the TBMR is evaluated using the framework of Mikolajczyk et al. [Mikolajczyk 2005]. Two applications using local invariant features are presented in sections 4.4.4 and 4.4.5 to compare the TBMR with other widely used detectors. The application to image registration in Section 4.4.4 highlights the accuracy and the robustness of the TBMR. The experiments are conducted on the Stanford Mobile Visual Search (SMVS) Data Set [Chandrasekhar 2011]. In Section 4.4.5, the application to 3D reconstruction using structure from motion is first tested on the dataset of Strecha et al. [Strecha 2008], providing the ground truth of the camera positions. The baseline error and angular error measurements reveal the accuracy of the TBMR. Then the 3D reconstruction experiments are conducted on some real images taken in a sunny day around some structure. The structure from motion succeeds in reconstructing a complete 3D model using the TBMR, whereas only part of the scenes are reconstructed for the 3D model by using other detectors. In all experiments, the parameters of the corresponding method are set with the recommended values.

4.4.1   Qualitative features comparison

We first compare the TBMR with the state-of-the-art local feature detectors by visualize the distribution of the keypoints obtained with each method. The methods that we tested are Harris-Affine, Hessian-Affine, DoG, DoG octave-1, MSER. For the shape-space based methods: MSER and TBMR, the centroid of the extracted regions is considered as the detected keypoints. The qualitative comparison are conducted on two images taken against the sunlight in a sunny day. The distribution of the keypoints extracted with different methods on the two tested images are illustrated respectively in Figure 4.4 and in Figure 4.5. From these two figures, the same assessment is observed: MSER detects few points, which explicates the failure of a 3D reconstruction using MSER in the example shown in Figure 4.1; Harris-Affine, Hessian-Affine, and DoG all extract a reasonable number of points, but few points on the real object of the scene, which make them fail to reconstruct the 3D structure correctly in Figure 4.1. By using the option octave-1 for DoG, a significantly higher number of points than DoG without this option are detected, but the additional points are mostly distributed where there were already many points, not on the real object. However, the proposed TBMR has a reasonable number of keypoints, and they are distributed more uniformly over the image, which contributes the main reason of its success of having a correct 3D reconstruction in Figure 4.1.

(a) (a) Harris-Affine.

(b) (b) Hessian-Affine.

(c) (c) DoG.

(d) (d) DoG octave-1.

(e) MSER.

(f) TBMR.

Figure 4.4: Qualitative comparison of TBMR with other widely used local feature detectors applied on an image taken against the sunlight (used in Figure 4.1). Yellow points in the image are the detected keypoints.

(a) (a) Harris-Affine.

(b) (b) Hessian-Affine.
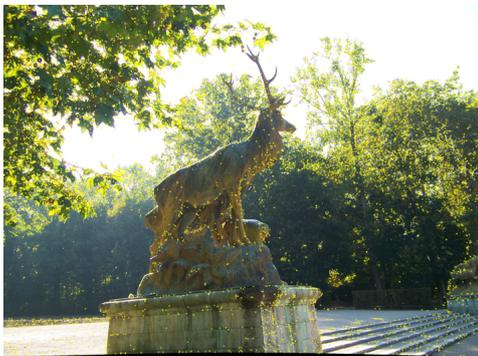
(c) (c) DoG.

(d) (d) DoG octave-1.

(e) MSER.

(f) TBMR.

Figure 4.5: Qualitative comparison of TBMR with other widely used local feature detectors applied on an image taken against the sunlight. Yellow points in the image are the detected keypoints.

4.4.2   Image coverage evaluation

In order to assess the uniformity of those keypoint distributions obtained with different methods, we first measure the distribution of keypoints along the two image dimensions, as well as the number of extracted points for a set of images taken around some scene objects. In Fig. 4.6, we show the distribution of keypoints position along the horizontal dimension for the images taken around the objects of scene presented respectively in Fig. 4.5, Fig. 4.17, and Fig. 4.18. These images are taken to make the object of interest presented in the middle (horizontally) of the scene. The distributions shown in Fig. 4.6 are smoothed by taking the average inside a horizontal window (size is set to 21). As shown in this figure, MSER extracts few points; TBMR has a number of points comparable with Harris-Affine, Hessian-Affine, and DoG without option octave -1; DoG with the option octave -1 has many more points. However, TBMR has the largest part of keypoints that cover the objects of interest in those scenes, which contributes to its success in Fig. 4.17 and 4.18.

We also evaluate how well the keypoints cover the image. First, we propose to dilate the extracted keypoints by a 2D window centered at each point with a certain size (e.g., 31). Then we compute the rate of area covered by the dilated region. Note that for two close keypoints, their dilated regions may have a large part in common, but the common regions count only once. As shown in Fig. 4.7, MSER covers a small part of the image because of a few extracted points. TBMR covers the image better than the others having a comparable or much larger number of detected points, which confirms the qualitative observation in Section 4.4.1.

4.4.3   Repeatability evaluation

To assess the performance of the proposed TBMR, we compare it with other affine detectors: Harris-Affine and Hessian-Affine, defined on the scale-space, and MSER, defined on the shape space.

We repeat the tests of Mikolajczyk et al. [Mikolajczyk 2005]. They evaluate the *repeatability score* based on the *overlap error* ε:

$$\varepsilon(R_{E_1}, R_{E_2}) = 1 - \frac{R_{E_1} \cap R_{H_{21}^T E_2 H_{21}}}{R_{E_1} \cup R_{H_{21}^T E_2 H_{21}}}, \tag{4.1}$$

where R_E represents the elliptic region (i.e., local patch of each extracted feature) defined by $x^T E x \leq 1$, and H_{21} is the ground truth homography between the test and reference image. The repeatability score for a pair of images is then defined as the ratio between the number of region-to-region correspondences established under a certain overlap error (e.g., 40%) and the smaller number of regions in the compared images. Another evaluated measurement is the absolute number of
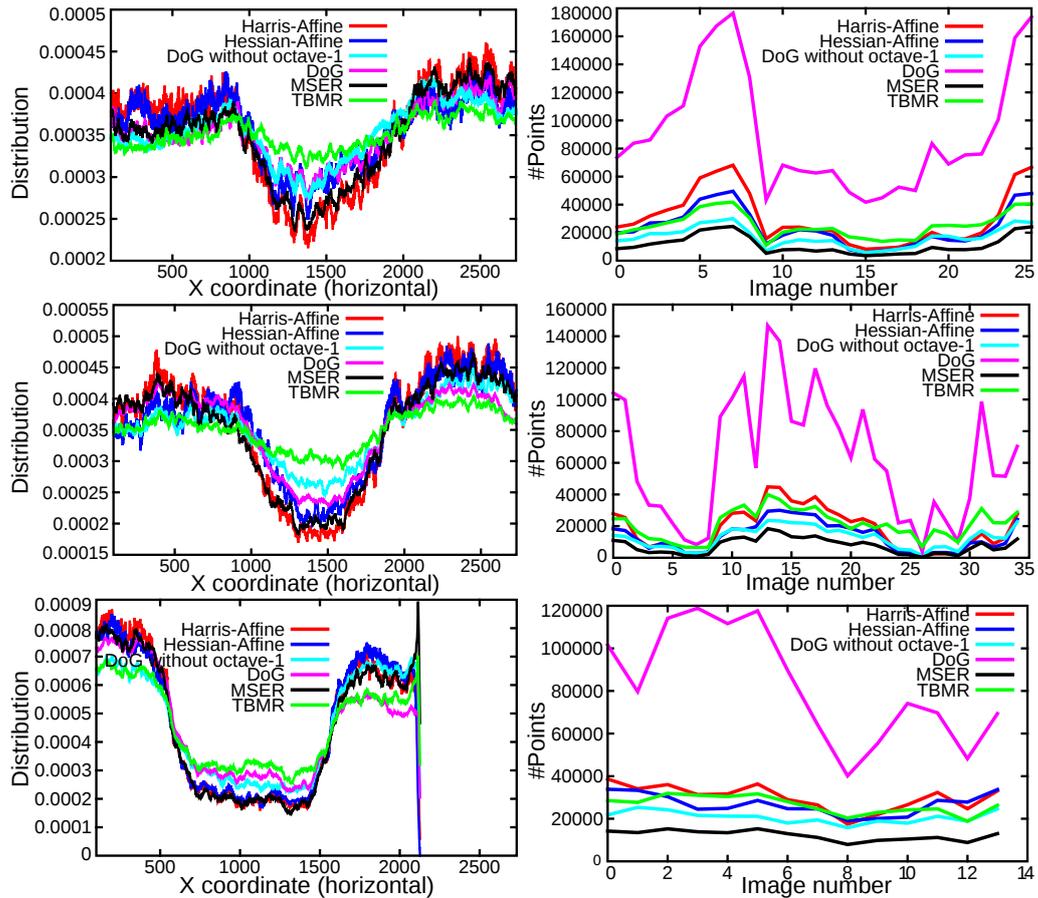
Figure 4.6: Horizontal distribution of the keypoints (left) and number of extracted keypoints (right), for the multi-view images taken around the objects of scene presented in respectively Fig. 4.5, Fig. 4.17, and Fig. 4.18 (top to bottom).
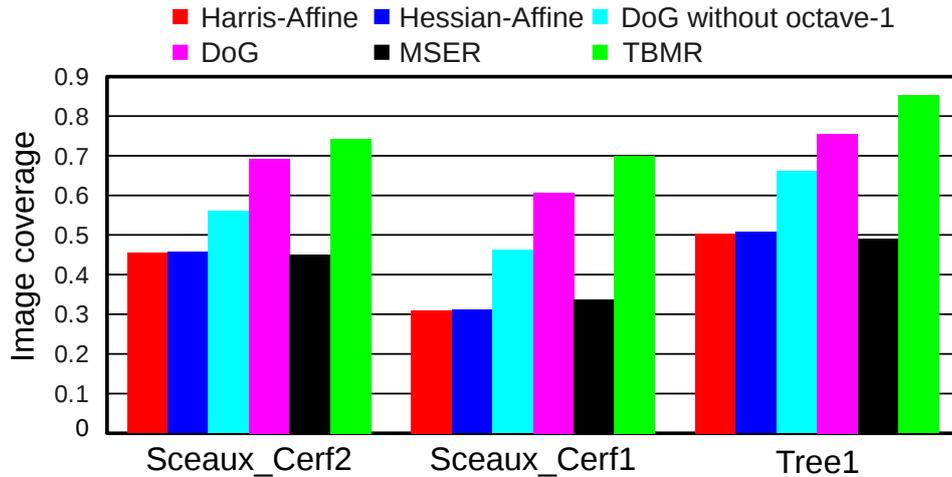
Figure 4.7: Image coverage evaluation by dilating a keypoint with a 2D window having size 31×31; Vertical axis represents the average rate of area covered by the dilated region for multi-view images presented in Fig. 4.5, Fig. 4.17, and Fig. 4.18.

correspondences. A high repeatability score and a large number of correspondences are normally desired.

Some results applied to the sequences *Wall* (viewpoint change), *Bark* (scale change), *Trees* (blur), *Leuven* (light change) [Mikolajczyk 2005] are illustrated in Figure 4.8. Compared to the scale-space based approaches (i.e., Harris-Affine and Hessian-Affine), the TBMR achieves a competitive repeatability score and a significantly higher number of correspondences, except for the blur sequence *Trees*. The explanation is that the topology of the image is damaged by the blur. For the same reason, the performance is poor on UBC sequence (not shown here), testing robustness to strong JPEG compression artifacts. Such defects (blur, JPEG artifacts) are better handled by the scale-space methods. Compared with the MSER which is also based on the shape space, the TBMR has a comparable repeatability score, but a significantly higher number of correspondences thanks to the contrast independent property of TBMR.

Experiments on other datasets (such as the dataset of DTU) that contains more images will be added.

4.4.4 Image registration

Image registration methods use the local features to establish a correspondence between a number of interest points (e.g., the centroids of the detected elliptical regions) in images. These one-to-one correspondences are then used to estimate
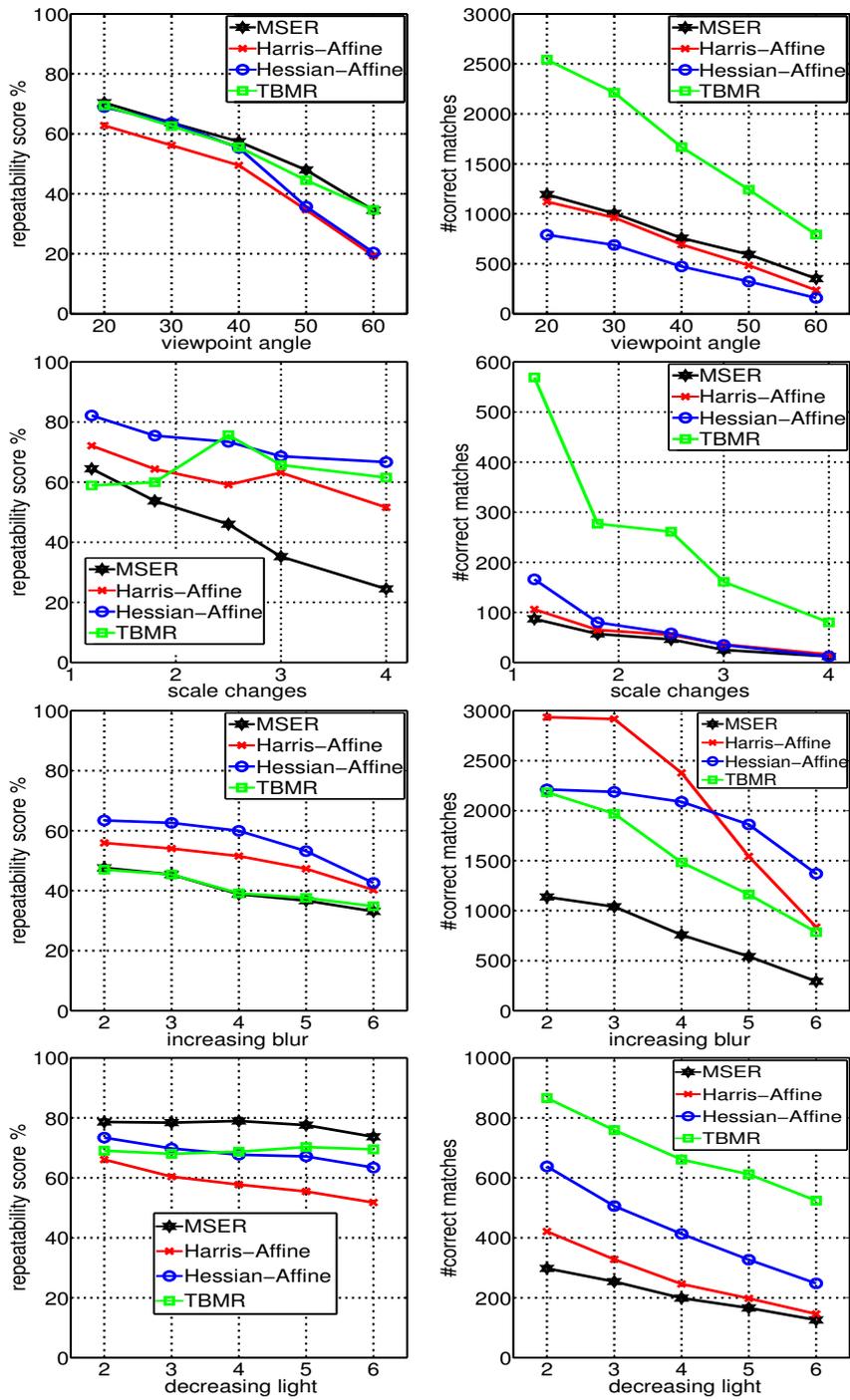
Figure 4.8: Repeatability score (left) and number of correspondences (right) for the sequences *Wall, Bark, Trees, Leuven* (top to down).

the transformation, thereby establishing point-by-point correspondence between the reference image and the target image. Hence, the accuracy and robustness of the local features is crucial to the quality of image registration result. There should also be enough pairs of matched points so that the estimation of the parameters of the transformation between images is possible.

For these experiments, we use the work of Moisan et al. [Moisan 2012]. It is based on the Optimized Random Sampling Algorithm (ORSA) proposed by Moisan and Stival [Moisan 2004], a variant of RANSAC algorithm introducing an *a contrario* criterion [Desolneux 2000] to avoid the fixed thresholds for inlier/outlier discrimination. ORSA is used to estimate the homography registering both images. When the homography is assessed, a panorama is built by stitching the images in the coordinate frame of the second image. The experiments are conducted on the CD-covers and paintings of the Stanford Mobile Visual Search(SMVS) Data Set [Chandrasekhar 2011]. As both the CD-covers and the paintings are planar scenes, they are amenable to homography registration. We experimented Harris-Affine, Hessian-Affine, MSER, DoG, and the proposed TBMR to compute point correspondences between images using the SIFT descriptors of Lowe [Lowe 2004]. The point correspondences is the input of ORSA.

For the images of Figure 4.9, Figure 4.11, and Figure 4.13 Harris-Affine, Hessian-Affine and MSER all fail in estimating the homography due to insufficient number of correspondences. And DoG also fails for images in Figure 4.11 and Figure 4.13. In Figure 4.9, although DoG achieves a homography, the registration result is inaccurate at the top left corner, whereas the TBMR results in a meaningful homography in all cases. A chessboard mix of the two registered images for Figure 4.9, Figure 4.11, and Figure 4.13 are given respectively in Figure 4.10, in Figure 4.12, and in Figure 4.14, from which the qualitative results can be better visualized. Note that for these image registration examples, there is no ground truth for these images and finding a relevant metric for such poor quality images is a challenge in itself, so only visual inspection is left to the reader's appreciation.

4.4.5   3D reconstruction

Structure from Motion (SfM) is a popular process of estimating a three-dimensional structures from a sequence of two-dimensional images. The SfM algorithms take multi-view stereo images along with the internal camera calibration information as input and yield a sparse 3D point cloud, camera orientations and poses in a common 3D coordinate system. The feature points are used in the phase of model estimations, including homography, fundamental and essential matrices, and camera poses. These estimations are very crucial to the quality of 3D reconstruction. Therefore,

DoG registration   TBMR registration

Figure 4.9: Homographic registration of a pair of images. Top: reference image (left) and target image (right); Down: registration results using respectively DoG (left) and TBMR (right). The result given by the use of DoG with standard parameters is not as accurate as the one based on TBMR (see the zoomed top left corner). Harris-Affine, Hessian-Affine, MSER with standard parameters all fail in registering the pair of images.

(a) Chessboard mix of the two registered images using DoG
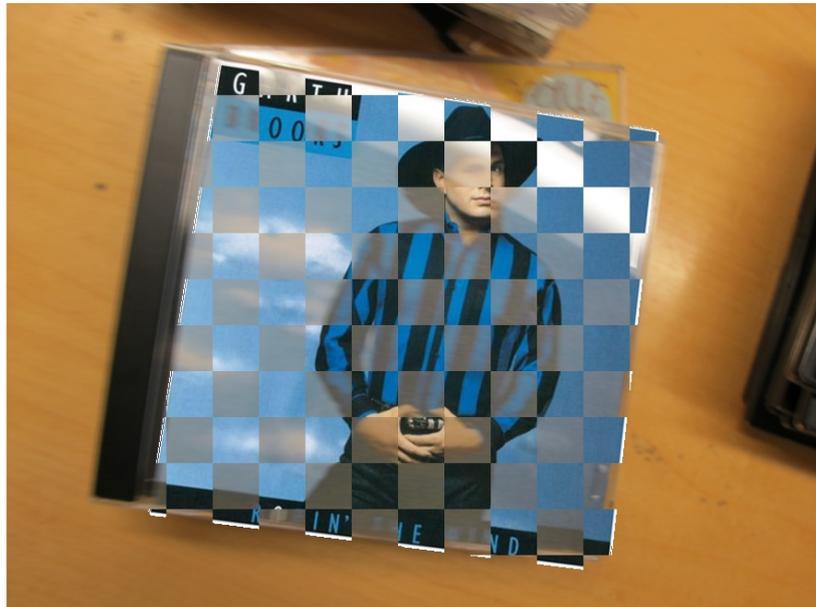
(b) Chessboard mix of the two registered images using TBMR

Figure 4.10: Chessboard mix of the two registered images using respectively DoG for (a) and TBMR for (b).

Figure 4.11: Homographic registration of a pair of images. Left: reference image. Middle: target image. Right: registration result using TBMR. All the other tested detectors using standard parameters do not provide a correct registration.
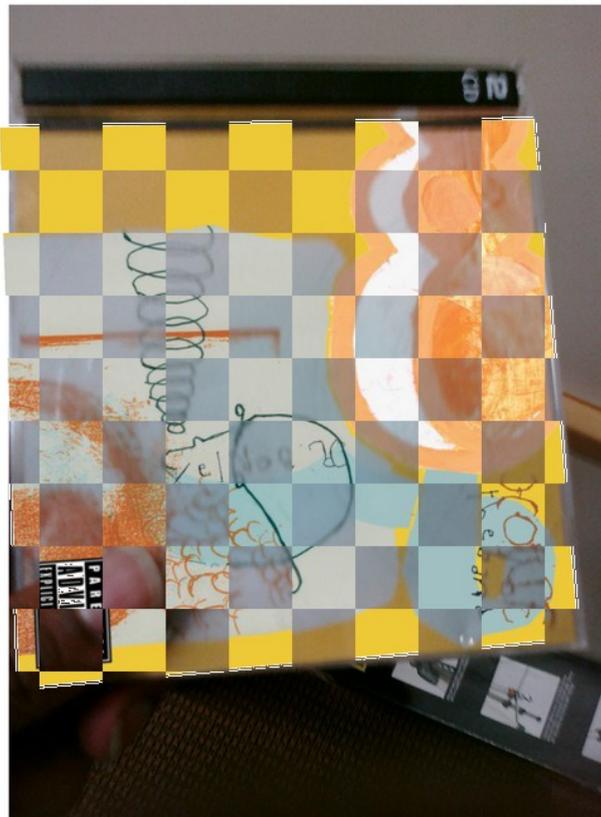
Figure 4.12: Chessboard mix of the two registered images using TBMR

Figure 4.13: Homographic registration of a pair of images. Left: reference image; Middle: target image; Right: registration result using TBMR. All the other tested detectors using standard parameters do not provide a correct registration.

Figure 4.14: Chessboard mix of the two registered images using TBMR

the accuracy and robustness of the local invariant features detectors is a defining aspect of the 3D reconstruction result.

To assess the performance of the feature detectors in this application, we use the software of Moulon et al. [Moulon 2012] which makes use of the *a contrario* criterion [Desolneux 2000] instead of globally-fixed thresholds for model estimation in SfM pipelines. In other respects, their pipeline is similar to the one of the popular Bundler software [Snavely 2006]: some initial pair of images are selected and two-view 3D reconstruction performed; other images are then sequentially added, each time refining the 3D scene with bundle adjustment, an iterative optimization method based on Levenberg-Marquardt algorithm. It is shown in their work that the adaptive SfM outperforms the state-of-the-art methods with significant precision improvements. The pipeline may stop prematurely at some point if not enough point correspondences are found to add further images.

4.4.5.1   Quantitative benchmark

We first benchmark the TBMR with Harris-Affine, Hessian-Affine, MSER and DoG on the public dataset of Strecha et al. [Strecha 2008], where the ground truth of the camera orientations and poses are also available. The SIFT descriptors of Lowe [Lowe 2004] is again used to establish one-to-one correspondences between the feature points for each detector. The quality of the 3D reconstruction is tested in terms of the precision of estimated camera orientations and poses. The baseline errors and angular errors compared to the ground truth are illustrated in Figure 4.15. For each sequence, the absence of the curve corresponding to some detectors means that it fails to calibrate all the cameras, or the baseline and angular errors are too high compared with others. Harris-Affine fails or the measurements are too high for all the sequences. Hessian-Affine fails or the measurements are too high for five sequences, while MSER works for three sequences. DoG and TBMR succeed for all the sequences. Compared to Harris-affine, Hessian-affine, MSER, the TBMR is more robust and behaves better in terms of baseline errors and angular errors. Compared to DoG, in most cases (especially for the sequences of *Herz-Jesus-P8* and *Castle-P19*), the TBMR performs better based on the baseline and angular errors. In Figure 4.16, we show also the recall rate of good tracks, the ratio between the amount of final maintained tracks, and the number of input tracks found between the feature points. The absolute number of final maintained tracks is presented as well. The highest recall rate and the largest number of 3D points obtained with the TBMR also reveal its robustness.

(a) (a) Herz-Jesus-P8.

(b) (b) Herz-Jesus-P25.

(c) (c) Entry-P10.

(d) (d) Fountain-P11.

(e) Castle-P19.

(f) Castle-P30.
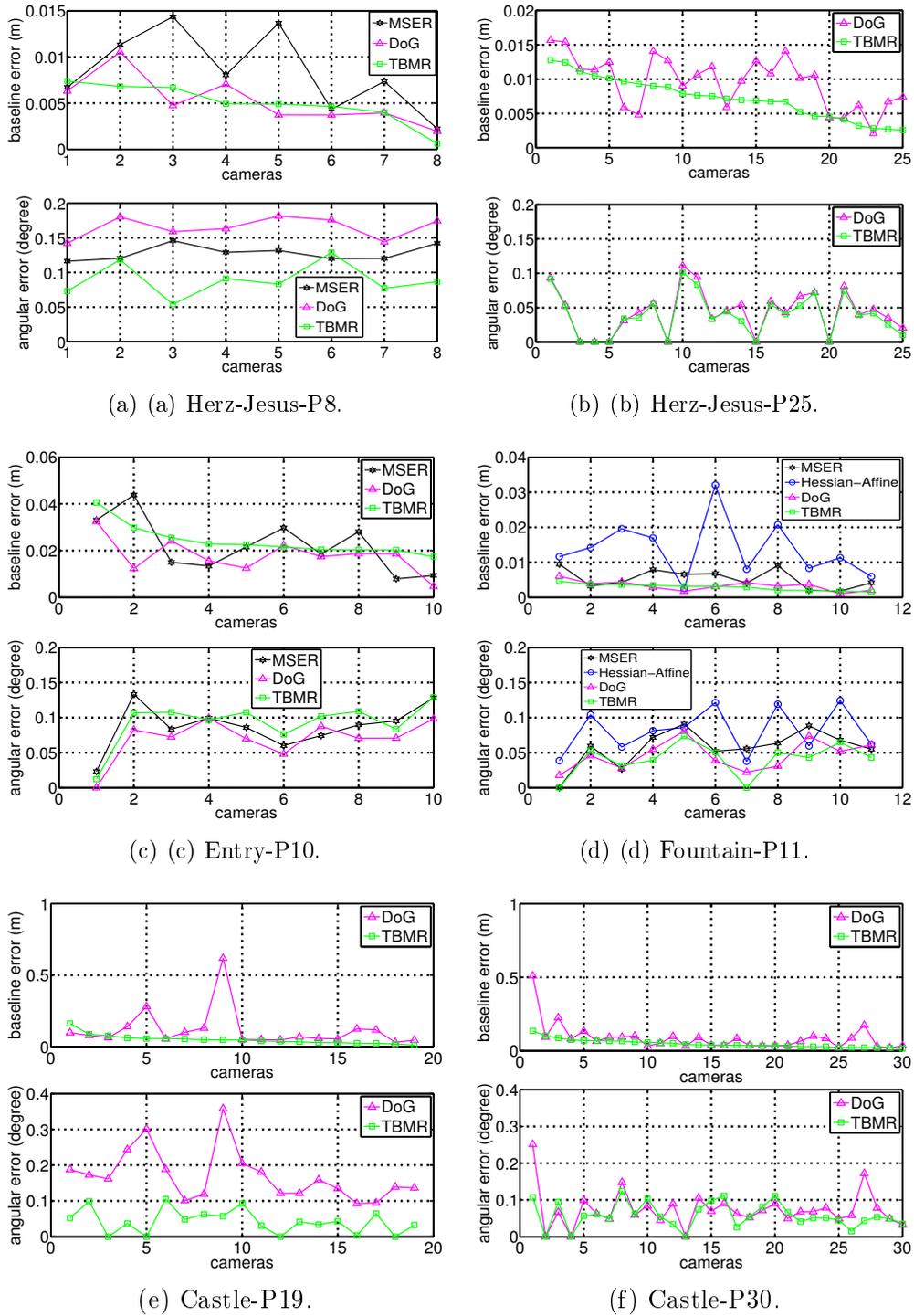
Figure 4.15: Evaluation of camera calibration based on baseline error and angular error applied on the dataset in [Strecha 2008].
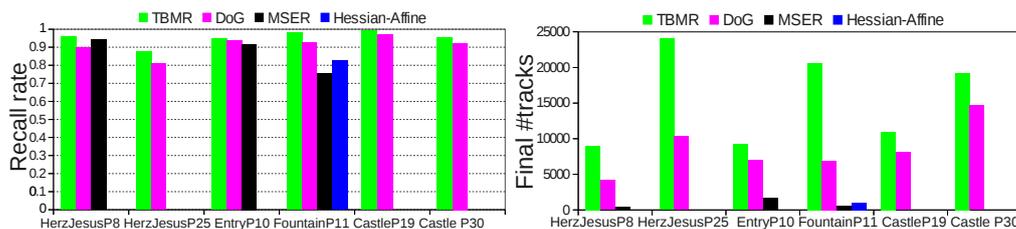
Figure 4.16: Recall rate (left) of the tracks and the absolute number (right) of final maintained tracks used to yield the sparse 3D points.

4.4.5.2   Qualitative results

We also test the described SfM with different detectors on some other images taken in a sunny day. Since the SfM produces a sparse 3D points cloud, not a dense 3D reconstruction, the PMVS software [Furukawa 2010] is used to densify the 3D points. Note that PMVS uses interest points of DoG and Harris corners to reconstruct the 3D structures based on the estimated model. Consequently, this may result in a lack of 3D points in a region (e.g., the surface near the bottom of the 3D structures on the right side of Figure 4.18 using the TBMR). By integrating the TBMR in PMVS, we would expect the reconstructed 3D structures to be denser thanks to its contrast independent property. In Figure 4.17, the DoG detector fails to reconstruct half of the scene where shadows are present. In Figure 4.18, DoG detector behaves similarly, whereas TBMR works well in all cases: the complete 3D structures are reconstructed. Note that Harris-Affine, Hessian-Affine and MSER perform even worse than DoG, so the corresponding results are not presented in Figures 4.17, 4.18. These examples also confirm the importance of the invariance to illumination changes, as pointed out by Aanæs et al. [Aanæs 2012].

4.5   Conclusion

We introduced a topological approach to local feature detection motivated by Morse theory. We made use of the critical points (i.e., minima, maxima, saddle points) in images and the shape space given by Max-tree and Min-tree built from the image. More precisely, we use the critical regions that are the leaves and nodes with bifurcation in the Max- and Min-trees. To each critical region, we proposed to associate the largest region from the shape space that contains it but does not contain any disjoint critical region. We have shown that the proposed method called TBMR is truly contrast independent, and quasi parameter free. Besides, the TBMR is fast to compute with a linear or quasi-linear complexity.

(a) 4 among 35 used multi-view images
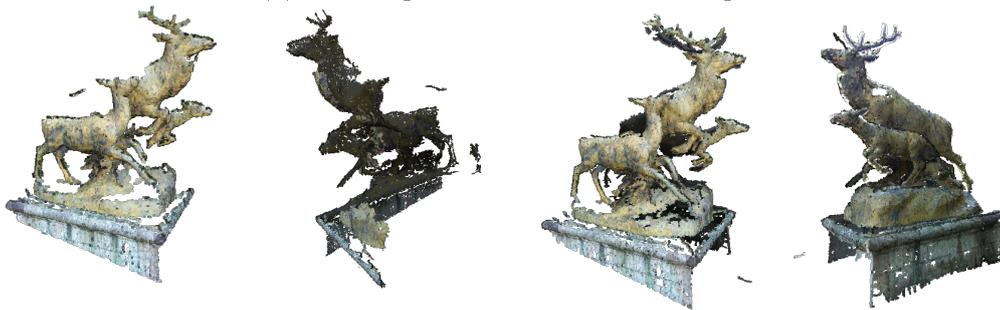
(b) 3D result using DoG   (c) 3D result using TBMR

Figure 4.17: Densified 3D reconstruction. (a) Some used images. (b) and (c) are respectively the left and right side of the reconstructed 3D structures using respectively DoG and TBMR. The use of DoG misses the right side of the scene; the use of TBMR results in a correct 3D reconstruction; other detectors behave worse than DoG.

(a) 4 among 14 used multi-view images.

      (b) 3D result using DoG.   (c) 3D result using TBMR.

Figure 4.18: Densified 3D reconstruction. (a) Some used images. (b) and (c) are respectively the left and right side of the reconstructed 3D structures using respectively DoG and TBMR. The use of DoG misses the right side of the scene; the use of TBMR results in a correct 3D reconstruction; other detectors behave worse than DoG.

Experimentally, we showed on standard data [Mikolajczyk 2005] that the developed TBMR achieves a reasonable repeatability and a significantly higher number of features in images. We conducted also some experiments on two applications relying on local features using public data sets. The homographic registration results and 3D reconstruction results demonstrated the accuracy and robustness of TBMR compared to other state-of-the-art detectors.

In the future, we plan to explore the choice of the associated region from the shape space for each critical point, which means instead of extracting the largest region in the definition of TBMR, we can imagine selecting the most meaningful region based on some significance measure (e.g., average of gradient's magnitude). But this significance should be designed invariant to illumination changes and affine transformations.

Hierarchical image simplification

Many methods relying on the morphological notion of *shapes*, (i.e., connected components of level sets of a tree-based shape space) have been proved to be very useful for pattern analysis and recognition. Selecting meaningful level lines (boundaries of regions in the shape space given by the topographic map representation) yields to simplify images while preserving salient structures. On the other hand, many image simplification and/or segmentation methods are driven by the optimization of an energy functional, for instance the Mumford-Shah functional. In this chapter, we propose an efficient self-dual morphological shaping that very quickly compute to a locally (subordinated to the topographic map) optimal solution of the piecewise-constant Mumford-Shah functional, which selects a set of salient level lines, and yields a simplified image from these level lines. Moreover, using the same principle of the piecewise-constant Mumford-Shah functional minimization subordinated to the topographic map, an attribute function $\mathcal{A}_\nu$ can be assigned to each level line, that characterizes its resistance under the energy minimization. Then by applying the framework of shape-based morphology as well as the attribute function $\mathcal{A}_\nu$, a saliency map representing a hierarchical image simplification will be obtained using the scheme presented in Section 3.6.2. Experimental results demonstrate the efficiency, usefulness, and robustness of our method, when applied to image simplification, color image pre-segmentation, and autophagosome counting in cellular images.

5.1 Introduction

In natural images, meaningful contours are usually smooth and well-contrasted. Recently, many authors claim that significant contours of objects in images coincide with segments of the image level lines [Caselles 1999]. Each connected level line is the contour of a region in the shape space given by a topographic map. Image simplification or segmentation can then be defined by selecting meaningful level lines in that tree. That subject has been investigated in the past ten years by [Pardo 2002, Cao 2005, Cardelino 2006]. In [Lu 2007] Lu et al. propose also a tree simplification method for image simplification purpose using the binary partition tree and a knee

function.

Following the seminal work of Mumford and Shah [Mumford 1989], finding relevant contours is often tackled thanks to an energy-based approach, as a compromise between some image-driven force (image contrast along contours, data fidelity, etc.) and the regularity of contours. Minimizing the Mumford-Shah functional tends to find a simplified or segmented image into regions. Curve evolution methods [Sethian 1999, Chan 2001] are usually used to solve such an energy minimization problem. They have solid theoretical foundations, yet they are often computational expensive.

In this chapter we propose to formalize the piecewise-constant Mumford-shah functional on an image, subordinated to the shape space given by its topographic map representation. The selection of the salient level lines corresponds to a meaningful locally optimal solution of the energy minimization problem. The main contribution is the proposition of an efficient greedy algorithm which takes into account the meaningfulness of the set of level lines. Simply put, a level line is easier to remove when it has a low degree of meaningfulness and when it favors a great decreasing of energy. Our algorithm drives very fast to a relevant local optimum in the sense that no more level lines can be removed while deceasing energy. The reason why we claim that we reach a relevant optimum is that meaningful level lines are hard to be removed during the proposed process. Note that our method actually belongs to the class of morphological shapings described in Section 3.4.5. Besides, a variant of this greedy algorithm that computes an attribute function $\mathcal{A}_\nu$ characterizing the resistance of being removed under the energy minimization will be proposed, along with the scheme of hierarchy transformation (see Section 3.6) provided by the framework of shape-based morphology, we can produce a saliency map to represent a hierarchical image simplification.

In [Guigues 2006], the authors proposed an efficient greedy algorithm to minimize the Mumford-Shah functional on a certain hierarchy, which leads to a global optimal segmentation on that hierarchy. In [Salembier 2009], the authors gave a detailed review of the tree filtering strategies (see also Section 2.5.2). The works in [Pan 2009a] and [Ballester 2007] are closest to what we propose here. They both select meaningful level lines for image simplification and segmentation purpose using the piecewise-constant Mumford-Shah functional. In [Pan 2009a] the whole image domain is initially considered as a single region; level lines of the topographic map browsed from root to leaves and are successively removed until the functional cannot decrease anymore. However, this top-down decision is based upon a non-significant energy variation since it is computed from the very few pixels lying between a shape and their immediate sub-shapes. Actually, our work is related to the one

described in [Ballester 2007], where at each removal step, the level line which decreases the most the functional is selected. As a consequence, the iterative process of [Ballester 2007] requires not only to compute a lot of information to be able to update the functional value after each level line suppression, but also to find at each step, among all remaining level lines, the one candidate to the next removal. Hence [Ballester 2007] is computationally expensive, while what we propose here is fast.

The rest of this chapter is organized as follows. Some background related work is reviewed in Section 5.2. Our proposed method is detailed in Section 5.3. In Section 5.4, we will show how to make this method hierarchical. In Section 5.5, we present some experimental results. We then conclude and give some perspectives in Section 5.6.

5.2   Related work

Connected operators are filtering tools that act by merging flat zones, the good contour preservation properties make them very useful for image simplification and segmentation. One popular implementation of the connected operators relies on the tree-based shape space (see Section 2.5). Many image simplification and segmentation methods relies on the relevant shapes extraction from the tree-based shape space are reviewed in Section 2.6.1.3. Here, we focus particularly on the relevant shapes (i.e., salient level lines) selection via variational models by minimizing some functional (e.g. Mumford-Shah functional in Section 5.2.1). We distinguish the functional on shape space into functional on hierarchy of segmentation (see Section 5.2.2) and functional on morphological trees, particularly the topographic map(see Section 5.2.3).

5.2.1   The Mumford-Shah Functional

According to the Mumford-Shah model [Mumford 1989], an image $f : \Omega \to \mathbb{R}$ or $\mathbb{Z}$ is modeled as a piecewise-smooth function. A segmentation of f is defined as a pair $(R, \tilde{f})$, where $R = R_1 \sqcup \cdots \sqcup R_n$, each region R_i is a connected component in the image domain and $\forall i \neq j, R_i \cap R_j = \emptyset$, $\tilde{f}$ is a regular function within each R_i. Let ∂R be the union of boundaries of the set of regions $\{R_i\}$:

$$\partial R = \bigcup_{i=1,\dots,n} \partial R_i, \tag{5.1}$$

then $R \sqcup \partial R$ is the whole image domain Ω. The segmentation is given by the minimization of the Mumford-Shah functional $E(f, \partial R)$ defined by:

$$E(f, \partial R) = \iint_R (\tilde{f} - f)^2 dx dy \, + \, \mu \iint_R ||\nabla \tilde{f}|| \, dx dy \, + \, \nu \, |\partial R|, \qquad (5.2)$$

where $|\partial R|$ represents the total length of the boundaries ∂R, and μ and ν are two positive parameters. The first term in (5.2) is the data fidelity; the second term penalizes the lack of smoothness of the model within each region; and the last term penalizes the total boundary length.

A special case of Eq (5.2) is obtained by restricting the segmented image $\tilde{f}$ to piecewise-constant functions. For each region R_i, it is given by:

$$\tilde{f} = \tilde{f}_i = \frac{1}{|R_i|} \iint f \, dx dy, \qquad (5.3)$$

where $|R_i|$ denotes the area of the region R_i. Then, the simplified Mumford-Shah functional is given by:

$$E(f, \partial R) = \iint_R (\tilde{f}_i - f)^2 \, dx dy + \nu \, |\partial R|. \qquad (5.4)$$

Since it is usually difficult to minimize the energy functional of Eq (5.2), the simplified Mumford-Shah functional given by Eq (5.4) is used instead.

Curve evolution methods [Sethian 1999] are usually used to solve such an energy minimization problem. Indeed, the optimization of the Mumford-Shah functional is predominated by active contour methods [Chan 2001, Vese 2002, Cremers 2002]. They have solid theoretical foundations, yet they are often computational expensive. Besides, these active contour based optimization methods evolves via gradient descent, which is known for its overdependence on initialization and its tendency to achieve undesirable local minima. Recently, in order to reduce these problems, some methods using combinatorial optimization techniques [Grady 2009, El-Zehiry 2011, El-Zehiry 2013] are proposed to minimize the Mumford-Shah functional in Eq (5.2) and in Eq (5.4). It has been show that the combinatorial optimization may achieve a lower energy compared with the gradient descent methods.

5.2.2   Functional on the hierarchy of segmentation

Tree-based shape spaces are a powerful tool for image simplification and segmentation, they produce a multi-scale representation which allows to design robust and fast algorithms. One class of the tree-based shape space, the hierarchies of segmentation H provides a tremendously reduced space of partitions (i.e., the family of all possible cuts of H denoted as $\mathbb{C}(\mathbb{H})$), compared with the space of all possible partitions of the image. We shall detail the minimization of the simplified Mumford-Shah

subordinated to the space of partitions given by a hierarchy of segmentation using the theory of scale-sets image analysis proposed by Guigues et al. [Guigues 2006]. Let $\mathcal{P}$ denotes a partition of image domain Ω, and let $\mathcal{P}(\Omega)$ and $\mathcal{P}art(\Omega)$ denote respectively the family of subsets of Ω and partitions of image domain Ω.

Definition 10 *An energy function E^λ: $\mathcal{P}art(\Omega) \to \mathbb{R}^+$ is an affine energy on $\mathcal{P}art(\Omega)$ if there exists two functions C, D: $\mathcal{P}art(\Omega) \to \mathbb{R}^+$ and a parameter $\lambda \in \mathbb{R}^+$ such that*

$$E^\lambda(\mathcal{P}) = \lambda C + D \quad \forall \mathcal{P} \in \mathcal{P}art(\Omega). \tag{5.5}$$

In practice, D is usually a goodness-of-fit term (e.g., L2 norm data fidelity), and C is a regularization term (e.g., length of boundaries). The piecewise-constant Mumford-Shah functional in Eq (5.4) is a such example denoted as E^ν.

Definition 11 *An energy function E: $\mathcal{P}art(\Omega) \to \mathbb{R}^+$ is a separable energy if*

$$E(\mathcal{P}) = \sum_{R_i \in \mathcal{P}} E(R_i) \quad \forall \mathcal{P} \in \mathcal{P}art(\Omega). \tag{5.6}$$

The two terms of the piecewise-constant Mumford-Shah functional are examples of separable energy function via the function E: $\mathcal{P}(\Omega) \to \mathbb{R}^+$:

$$E(R_i) = \iint_{R_i} (\tilde{f}_i - f)^2 \, dxdy + \frac{\nu}{2} |\partial R_i| \quad \forall R_i \in \mathcal{P}(\Omega), \tag{5.7}$$

where $|\partial R_i|$ denotes the length of the contour of region R_i.

Definition 12 *An energy function E: $\mathcal{P}(\Omega) \to \mathbb{R}^+$ is a subadditive energy if*

$$E(R_i \cup R_j) \leq E(R_i) + E(R_j) \quad \forall R_i, R_j \in \mathcal{P}(\Omega) \text{ such that } R_i \cap R_j = \emptyset. \tag{5.8}$$

The regularization term of the piecewise-constant Mumford-Shah in Eq (5.4) (i.e., length of boundaries) is an example of subadditive energy function.

Definition 13 *Let $E^\lambda = \lambda C + D$ be an affine energy. Then E^λ is said to be a multi-scale energy if C and D are separable and C is subadditive. The value λ is called the* scale parameter *of the energy.*

Recall that a parameter λ behaves as a *scale parameter* if it obeys the causality principle (see Section 2.3.7 and [Koenderink 1984, Morel 1995, Guigues 2006]), which says for any couple of scales $\lambda_2 > \lambda_1$, the "structure" found at scale λ_2 must

find some "cause" at scale λ_1. In the case of image partitions, this principle is defined through the presence of boundaries, the boundaries of partition $\mathcal{P}_{\lambda_2}$ are in one-to-one mapping of a subset of boundaries of partition $\mathcal{P}_{\lambda_1}$, which means partition $\mathcal{P}_{\lambda_1}$ is finer than partition $\mathcal{P}_{\lambda_2}$: $\mathcal{P}_{\lambda_1} \sqsubseteq \mathcal{P}_{\lambda_2}$.

The piecewise-constant Mumford-Shah functional in Eq (5.4) denoted as E^ν is a multi-scale energy. Let us review the algorithm proposed by Guigues et al. in [Guigues 2006] that deals with a multi-scale energy E^λ subordinated to a hierarchy of segmentation H. For any $\lambda \in \mathbb{R}^+$, let $C^*_\lambda(H) \in \mathbb{C}(H)$ be the cut of the hierarchy H that minimizes E^λ. A region $R_i \in H$ which belongs to this cut $C^*_\lambda(H)$ is called λ-optimal. For each region $R_i \in H$, let $H(R_i)$ be the partial hierarchy rooted at R_i given by:

$$H(R_i) = \{R_j \,|\, R_j \in H \text{ such that } R_j \subseteq R_i\}, \tag{5.9}$$

and let $\mathbb{C}\big(H(R_i)\big)$ be the family of all possible cuts of the partial hierarchy $H(R_i)$. As proved in [Guigues 2006], we have the following proposition:

Proposition 10 *A region $R_i \in H$ is λ-optimal for an multi-scale energy E^λ: $\mathbb{C}(H) \to \mathbb{R}^+$ if and only if the following two properties hold:*

(i) R_i is partially λ-optimal, i.e., $\forall C(H(R_i)) \in \mathbb{C}(H(R_i)), E^\lambda(\{R_i\}) \leq E^\lambda(C(H(R_i)))$.

(ii) R_i is maximal in H for the property (i), which means there is no region $R_j \in H$ such that $R_i \subset R_j$ is also partially λ-optimal.

In what follows, let $P^*_\lambda(H)$ be the set of all partially λ−optimal nodes of H, that is the set of nodes of H which verify the property (i) of Proposition 10 for the energy E^λ.

For each region $R_i \in H$, let

$$\Lambda^*(R_i) \triangleq \{\lambda \,|\, R_i \in C^*_\lambda(H)\} \tag{5.10}$$

be the set of scales such that the region R_i is in the cut of H minimizing E^λ. Also let

$$\Lambda^*_\uparrow(R_i) \triangleq \{\lambda \,|\, R_i \in P^*_\lambda(H)\} \tag{5.11}$$

be the set of scales for which R_i is partially λ-optimal. As shown in [Guigues 2006], when E^λ is a multi-scale energy, if R_i is partially λ-optimal, then it is also partially λ'-optimal for any $\lambda' > \lambda$. Thus $\Lambda^*_\uparrow(R_i)$ is an interval of type $[a, +\infty)$. Following the maximal principle of condition (ii) in Proposition 10, we define also

$$\Lambda^*_\downarrow(R_i) \triangleq \bigcup_{R_j \in H, R_i \subset R_j} \Lambda^*_\uparrow(R_j) \tag{5.12}$$

which represents a set of scales for which R_i is no longer maximal for the property (i) in Proposition 10. Note that the union of intervals of type $[a, +\infty)$ is still an interval of that type. The two conditions of a region R_i being λ-optimal in Proposition 10 can be summarized as follows:

$$\Lambda^*(R_i) = \Lambda^*_\uparrow(R_i) \backslash \Lambda^*_\downarrow(R_i), \tag{5.13}$$

which leads to the following proposition:

Proposition 11 *Given a multi-scale energy E^λ subordinated to a hierarchy of segmentation H, we have $\forall R_i \in H$, the set $\Lambda^*(R_i)$ is an interval of the type $\Lambda^*(R_i) = [\lambda^+(R_i), \lambda^-(R_i)$, where*

$$\lambda^+(R_i) = \inf\Lambda^*_\uparrow(R_i), \tag{5.14}$$
$$\lambda^-(R_i) = \inf\Lambda^*_\downarrow(R_i). \tag{5.15}$$

Thus we have

$$C^*_\lambda(H) = \{R_i \,|\, R_i \in H \text{ such that } \lambda^+(R_i) \leq \lambda < \lambda^-(R_i)\}. \tag{5.16}$$

The interval $\Lambda^*(R_i)$ for a region $R \in H$ is called the persistence of the region R_i. The notion of persistent hierarchy H^* obtained from a hierarchy and a multi-scale energy E^λ is defined as follows:

$$H^* \triangleq \{R_i \,|\, R_i \in H \text{ such that } \Lambda^*(R_i) \neq \emptyset\}. \tag{5.17}$$

On this persistent H^*, we have $\lambda^-(R_i) = \lambda^+(R_i^p)$, where R_i^p denotes the parent of R_i in H^*. Thus knowing $\lambda^+(R_i)$ for each node of H^*, λ^- can be easily computed. We shall now explain how λ^+ can be efficiently computed via dynamic programming method [Guigues 2006].

For each $R_i \in H, \lambda \in \mathbb{R}^+$, we define

$$E(\lambda, R_i) = \lambda C(R_i) + D(R_i). \tag{5.18}$$

We define the partial energy of the node $R_i \in H$ as the energy of the optimal cut of $H(R_i)$ with respect to E^λ, and we denote it by $E^*(\lambda, R_i)$:

$$E^*(\lambda, R_i) = E^\lambda(C^*_\lambda(H(R_i))). \tag{5.19}$$

Note that for any leave node $R_l \in H$, we have $E^*(\lambda, R_l) = E(\lambda, R_l)$ holds for any $\lambda \in \mathbb{R}^+$. As shown in [Guigues 2006], we have the following propositions:

Proposition 12 *For each node $R_i \in H$, the partial energy $E^*(\lambda, R_i)$ is related by the dynamic programming equation:*

$$E^*(\lambda, R_i) = \inf \left\{ E(\lambda, R_i), \sum_{S \in \mathcal{F}(R_i)} E^*(\lambda, S) \right\}, \qquad (5.20)$$

where $\mathcal{F}(R_i)$ is the family of children of R_i.

Proposition 13 *Let $E^\lambda = \lambda C + D$ a multi-scale energy on the hierarchy of image segmentation H, then $\forall R_i \in H$, we have*

(i) $E^*(\lambda, R_i)$ is a piecewise affine, non decreasing, continuous and concave function of λ.

(ii) $\forall \lambda \in \mathbb{R}^+$,

$$E^*(\lambda, R_i) = \begin{cases} \sum\limits_{s \in \mathcal{F}(R_i)} E^*(\lambda, S) & \text{if } \lambda < \lambda^+(R_i) \\ E(\lambda, R_i) & \text{otherwise} \end{cases} \qquad (5.21)$$

(iii) If C is strictly subadditive, which means $\forall R_j \in H, C(R_j) < \sum\limits_{S \in \mathcal{F}(R_j)} C(S)$ (e.g., the boundary length of the piecewise-constant Mumford-Shah functional in Eq (5.4)), then $\lambda^+(R_i)$ is the unique solution of

$$E(\lambda, R_i) = \sum_{S \in \mathcal{F}(R_i)} E^*(\lambda, S). \qquad (5.22)$$

Combining the results in Propositions 11, 12, 13, we are able to compute the partition given by the optimal cut $C^*_\lambda(H)$ minimizing the piecewise-constant Mumford-Shah functional in Eq (5.4) subordinated to a hierarchy of segmentation H for any $\lambda \in \mathbb{R}^+$. Note that this $C^*_\lambda(H)$ obtained with the work of Guigues et al. [Guigues 2006] is a global optimal solution among all the possible cuts of the hierarchy of segmentation.

5.2.3 Functional on the topographic map

The works in [Pan 2009a] and [Ballester 2007] are closest to our simplification method in this chapter. The basic idea is to minimize the simplified Mumford-Shah functional in Eq. (5.4) subordinated to the topographic maps. The segmentation is restricted to the regions whose boundaries are level lines. Although this is not the general context for a segmentation, since the meaningful boundaries of objects

may not coincide with full level lines, but segments of them [Caselles 1999]. However, the main edges of the image are included in them, and the level lines are contrast invariant and robust. So still this is very useful as an image simplification or pre-segmentation tool.

More specifically, for a given topographic map $\mathcal{T}$ composed of a set of shapes $\{\tau_i\}$, any two successive shapes of $\mathcal{T}$ are related by an edge reflecting the inclusion relationship, also known as the parenthood between nodes of the tree. Let

$$\partial\mathcal{T} = \{\partial\tau \mid \tau \in \mathcal{T}\} \tag{5.23}$$

be the union of boundaries of all the shapes of $\mathcal{T}$. The minimization of the energy functional of Eq (5.4) restricted to $\mathcal{T}$ is given by:

$$\min_{\mathcal{T}'} E(f, \partial\mathcal{T}'), \tag{5.24}$$

where $\mathcal{T}'$ is a simplified version of $\mathcal{T}$ by removing some shapes from $\mathcal{T}$ and by updating the parenthood relationship between the shapes of $\mathcal{T}'$. The parenthood relationship between those preserved shapes is updated by taking the lowest preserved ancestor as the parent. The inclusion relationship between any two successive shapes of $\mathcal{T}'$ still holds. So $\mathcal{T}'$ has also a tree structure.

The basic operation of the energy minimization problem of Eq (5.24) is the merging of the element "regions" formed by the level lines $\partial\mathcal{T}$. For each given shape τ, the induced element "region" R_τ is defined as:

$$R_\tau = \{p \mid p \in \tau, p \notin C(\tau)\}, \tag{5.25}$$

where $C(\tau)$ denotes all the children of the shape τ. Note that R_τ is not always a spatial connected component due to the fact that two successive level lines may have some part in common, this may cause a spatial separation of those pixels within R_τ. But these pixels are indeed in the same connected component of τ and any other ancestor of τ. An element "region" R_τ is merged with the element "region" R_{τ_p} induced by its parent τ_p. This merging process can also be seen as a level line suppression, which will cause the update of the "region" for its parent:

$$R'_{\tau_p} = R_{\tau_p} \cup R_\tau. \tag{5.26}$$

The parenthood relationship for its children $\tau_{c1}, \ldots, \tau_{ck}$ should also be updated to τ_p. Fig. 5.1 shows an example of a such merging operation.

Observe that the minimization problem of Eq. (5.24) is a combinatorial optimization. The computation of the optimum has an exponential complexity. Hence the greedy algorithm is usually applied to compute a local optimum instead of a
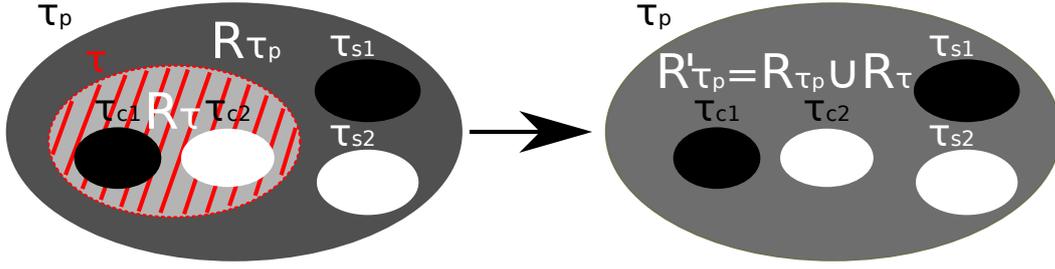
Figure 5.1: Suppressing the node τ makes the "region" R_τ (covered with red oblique lines) merge with R_{τ_p}; the result (depicted in the right image) is a simplified image.

global optimum. It iteratively removes those level lines, while removed, the functional decreases. The greedy algorithm stops if no more level line can be removed that favors a decrease of the energy functional. The removability of a level line $\partial\tau$ is decided by the sign of the simplified functional of Eq. (5.4) variation ΔE_τ while τ is suppressed. Let $S(f, R_i)$ be the sum of value of all the pixels inside region R_i. Then the functional variation ΔE_τ is given by:

$$\Delta E_\tau = \frac{S^2(f, R_\tau)}{|R_\tau|} + \frac{S^2(f, R_{\tau_p})}{|R_{\tau_p}|} - \frac{S^2(f, R'_{\tau_p})}{|R'_{\tau_p}|} - \nu|\partial\tau|. \qquad (5.27)$$

If ΔE_τ is negative, which means the suppression of τ decreases the functional, then remove τ. According to Eq. (5.27), the removability of a shape τ depends only on R_τ and R_{τ_p}. As a shape τ suppression triggers the update of R_{τ_p}, the removal of τ impacts also the removability of its relatives, i.e., its parent, its children and siblings. So the order of level line removal is critical. Different choice of suppression order drives to different local optimum solution of (5.24). In [Pan 2009a], Pan propose to remove the level lines for top to down (i.e., root node to leaves node). In the work of Ballester et al. In [Ballester 2007], the authors propose to remove the level line that decreases the most the functional at each remove step.

5.3   Efficient salient level lines selection

In this section, we will detail the proposed method for image simplification based on the minimization of piecewise-constant Mumford-Shah functional subordinated to the topographic maps. It is an efficient alternative to the method of Ballester et al [Ballester 2007] by introducing an efficient level lines ordering heuristics. We first discuss the importance of the level lines ordering in Section 5.3.1 for the greedy algorithm minimizing the energy functional. Then we will present the proposed algorithm in Section 5.3.2, where we will also show that this simplification method is one

of the many variant of self-dual morphological shapings introduced in Section 3.4.5.

5.3.1 Level lines sorting

As discussed in Section 5.2.3, the order of level lines removal is critical for the greedy algorithm minimizing the energy functional in Eq (5.4) subordinated to the topographic map. The order used in the work of Pan [Pan 2009a] is a top-down (i.e., from root node to the leaves node) decision. Such order is not quite reliable, because the removal decision for a level line $\partial\tau$ under scrutiny is based on the sign of the Eq (5.27) in which R_τ has very few pixels lying between the level line $\partial\tau$ itself and its direct children, whereas R_{τ_p} may have many more pixels. In the work of Ballester et al. [Ballester 2007], the order is decided by the greedy algorithm itself. At each step, the algorithm selects the shape among the non removed ones that decreases the most the energy functional. It bears a heavy computational update of ΔE for the nodes that are relatives of the suppressed level line. In addition, after each update, it also requires to find the new level line with the highest negative functional variation ΔE. We propose to fix that issue by introducing a reasonable ordering of level lines based on their meaningfulness.

In the framework of classical connected operators (see Chapter 2), the attribute function $\mathcal{A}$ characterizing each node plays a very important role in connected filtering, which have been shown be interesting for image simplification/segmentation in many works [Breen 1996, Salembier 1998, Salembier 2000, Wilkinson 2001, Ballester 2007, Lu 2007]. For example, let attribute function $\mathcal{A}$ that measures the meaningfulness of each shape (i.e., a level line), then the classical connected filtering is usually performed by removing the nodes whose attribute is lower than a given threshold. We propose to combine this idea of classical connected operators with the energy minimization problem of Eq (5.24). Simply put, for each shape $\tau \in \mathcal{T}$, we assign an attribute function $\mathcal{A}$, and we sort the set of shapes $\{\tau \,|\, \tau \in \mathcal{T}\}$ in increasing order of the attribute function $\mathcal{A}$, then following this order, apply the level lines removal process according to the sign of Eq (5.27).

Observe that the minimization of energy functional in Eq (5.4) favors the removal of level lines having small contrast (by data fidelity term) or being complex, which is in accord with the fact that meaningful contours in natural images are usually well-contrasted and smooth. Thus the average of gradient's magnitude along the level line defined in Eq (3.8) and the compactness defined in Section 2.5.2 might be useful for the level lines sorting. Besides, the average of curvature along the level lines defined as bellow might also be interesting.

$$\mathcal{A}_\kappa(\mathcal{N}) = \big(\sum_{e \in \partial\mathcal{N}} \mathrm{curv}(e) \big)/|\partial\mathcal{N}|, \tag{5.28}$$

where $|\partial \mathcal{N}|$ denotes the length of the boundary, and curv $: E \to \mathbb{R}$ is an image of curvature.

In the point of view of classical connected filtering, the level lines removal guided by the energy minimization of Eq (5.4) make the filtering decision more robust. In fact, for the classical connected filtering, the attribute thresholding strategy is equivalent to localize a node $\mathcal{N}_c$ among the sorted level lines, and filter all the nodes from the beginning till this node $\mathcal{N}_c$ in of the sorted level lines. Whereas, the level lines removal decision made by the energy minimization filters many level lines in the beginning of the sorted level lines, and fewer and fewer level lines at the end of the sorted level lines.

In the point of view of energy minimization of Eq (5.24) the shapes having small (resp. great) attribute $\mathcal{A}$ (i.e., meaningfulness) is easier (resp. more difficult) to filter under the energy minimization process. The level lines sorting provides a reasonable order to perform the level lines suppression that make the energy functional decrease.

5.3.2   Efficient greedy algorithm

For a given image f and its associated topographic map representation $\mathcal{T}$, the output is a simplified tree $\mathcal{T}'$ from which a simplified image f' is reconstructed. Our proposed algorithm is described in the following:

Initialization: The output tree, $\mathcal{T}'$, will be incrementally simplified. First, it is set to $\mathcal{T}' = \mathcal{T}$.

Step 1: Sort the set of shapes $\{\tau \,|\, \tau \in \mathcal{T}\}$ in the increasing order $\mathcal{O}$ of shape meaningfulness indicated by an attribute function $\mathcal{A}$.

Step 2: Propagate the shapes in the order $\mathcal{O}$, for each shape $\tau \in \mathcal{T}'$, compute ΔE_τ. If it is negative, remove τ from $\mathcal{T}'$, update R'_{τ_p} and update the parenthood relationship for its children on $\mathcal{T}'$.

As a level line removal process during the minimization process affects the removability of its parent and children, so it is possible that for a shape τ that has been verified to be preserved, the preservation decision of τ may be altered by the later removal of its parent or some child. Consequently, step 2 may need to be iterated until no shape can be removed. In practice, 2 or 3 propagations are enough, and most of the level lines are removed in the first propagation. In consequence, the complexity of the minimization problem of Eq. (5.24) is linear w.r.t. the number of shapes of $\mathcal{T}$.

Note that the image reconstructed f' from the last tree $\mathcal{T}'$ is a locally optimal solution of Eq. (5.24) in the sense that any more level line removal will increase the simplified Mumford-Shah functional.

As the proposed greedy algorithm removes the level lines everywhere in the tree structure $\mathcal{T}$ no matter the attribute function $\mathcal{A}$ used for sorting the level lines is increasing or not, and as the underlying tree $\mathcal{T}$ is the topographic map, so this simplification method is not a leveling. It belongs actually to the family of self-dual morphological shapings introduced in Section 3.4.5. However, for this greedy algorithm, no explicit shape attribute function $\mathcal{A}_s$ is utilized to apply shaping, the shaping process is performed by the energy minimization of Eq (5.4).

5.4 Making the method hierarchical

In this section, we will show how to expand the principle of image simplification described in Section 5.3 to produce a hierarchical image simplification. It is achieved by using the framework of shape-based morphology described in Chapter 3. First of all, we will discuss the causality principle for hierarchical image simplification in Section 5.4.1, then in Section 5.4.2, we will show how to deduce an explicit attribute function $\mathcal{A}_\nu$ following the idea of salient level lines selection by minimizing the simplified Mumford-Shah functional. Finally, a saliency map representing a hierarchical image simplification will be detailed in Section 5.4.3 relying on the framework of shape-based morphology.

5.4.1 Causality principle

In the case of the piecewise-constant Mumford-Shah functional minimization applied on a hierarchy of segmentation (see Section 5.2.2), the parameter ν in Eq (5.4) is a scale parameter satisfying the causality principle, which says for $\nu_2 > \nu_1$, a contour presented at a larger scale ν_2 must also present at the scale ν_1. We propose to adapt the causality principle to the proposed image simplification in Section 5.3 by hierarchical salient level lines selection.

Definition 14 *For two parameters $\nu_2 > \nu_1$, if the level lines selected by minimizing the energy functional in Eq (5.4) with ν_2 are also preserved by minimizing the energy functional with ν_1, then the method of salient level lines selection is causal and hierarchical.*

The greedy algorithm proposed in Section 5.3 is a fast alternative of the algorithm of Ballester et al. [Ballester 2007], but it is not hierarchical according to Definition 14. Because for some level line τ, it may be removed with a parameter ν_1, but preserved for a bigger $\nu_2 > \nu_1$. A such example is given in Figure 5.2, the shape on top middle of the image (c) is preserved for $\nu_2 = 500$, while it is re-

<div align="center">(a)   (b)   (c)</div>
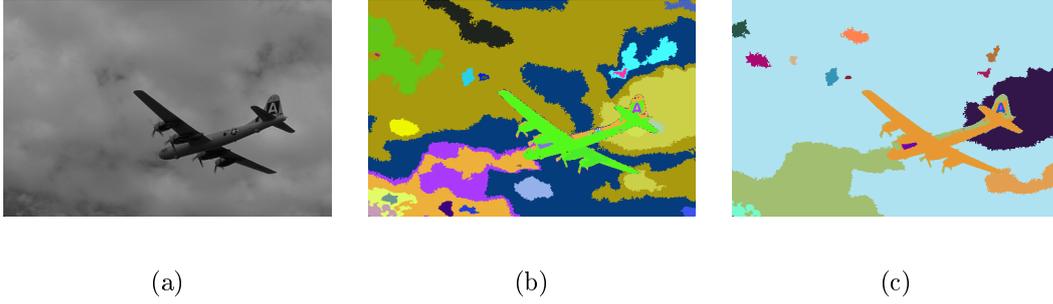
Figure 5.2: Illustration of causality principle violation. (a): input image; (b): randomly colorized simplified image with $\nu = 100$; (c): randomly colorized simplified image with $\nu = 500$.

moved for $\nu_1 = 100$ in image (b). Note that the original algorithm of Ballester et al. [Ballester 2007] is not hierarchical neither.

5.4.2   Attribute $\mathcal{A}_\nu$ based on the functional

In order to make the simplification method described in Section 5.3 hierarchical, we propose to use the framework of shape-based morphology, where an attribute function $\mathcal{A}$ characterizing each node is required. Thus instead of fixing the parameter ν in Eq (5.4), we compute an individual ν for each shape of the tree following the same principle of the energy minimization. Let us denote this as $\mathcal{A}_\nu$.

In fact, for a given ν, the removability of a shape τ is based on the sign of ΔE_τ in Eq (5.27). The energy variation ΔE_τ is a increasing linear function with respect to ν, when ν is bigger enough than some value ν_{min}, ΔE_τ will be negative, which implies the removal of this shape decreases the energy functional, and it will be removed. Thus the ν_{min} is a value of transition for the removal decision of the underlying shape. Let us this value of transition as the attribute function $\mathcal{A}_\nu$ given by

$$\mathcal{A}_\nu(\tau) = \frac{\frac{S^2(f, R_\tau)}{|R_\tau|} + \frac{S^2(f, R_{\tau_p})}{|R_{\tau_p}|} - \frac{S^2(f, R'_{\tau_p})}{|R'_{\tau_p}|}}{|\partial \tau|}. \tag{5.29}$$

Note that for a given shape τ, the attribute function $\mathcal{A}_\nu(\tau)$ defined in Eq (5.29) depends on its element "region" R_τ and the element "region" of its parent R_{τ_p}, which means $\mathcal{A}_\nu(\tau)$ is decided by the shape τ itself, its parent, its siblings, and its children. Besides, since the attribute function $\mathcal{A}_\nu$ is computed under the hypothesis that the shape τ under scrutiny is suppressed, in the same way as the algorithm described in Section 5.3.2, we need to update the element "region" for its parent by Eq (5.26), and update the parenthood relationship for its children to its parent. These update

operations will affect the computation of $\mathcal{A}_\nu$ for the parent, children and siblings of τ. So the computation order is again important. We propose three attribute function $\mathcal{A}_\nu$ computations based on different propagation orders.

1 Shape meaningfulness increasing

step 1: Compute ν for each shape $\tau \in \mathcal{T}$ according to 5.29 as the initial $\mathcal{A}_\nu$ and sort the set of shapes $\{\tau \,|\, \tau \in \mathcal{T}\}$ on basis of the shape meaningfulness increasing order as described in Section 5.3.1.
step 2: Propagated the sorted shapes in increasing order, and remove the shape ν one by one. Update the value $\mathcal{A}_\nu$ for its not suppressed parent, children, and siblings only if they are bigger than the old values. Update also the parenthood relationship.

2 Shape meaningfulness and ν increasing

step 1: The same as the first step of the shape meaningfulness increasing order.
step 2: Propagate the sorted shapes in increasing order, remove the first not suppressed shape τ and all the shapes whose current $\mathcal{A}_\nu$ is less than $\mathcal{A}_\nu(\tau)$. Do the same update operations for those related shapes of the removed one/ones.

3 ν increasing

step 1: Compute ν for each shape $\tau \in \mathcal{T}$ as the initial value of $\mathcal{A}_\nu$.
step 2: Remove the shape with the smallest $\mathcal{A}_\nu$ and do the update operations. Repeat this step till all shapes are suppressed.

Among these three possibilities of computing $\mathcal{A}_\nu$, the first one based on the shape meaningfulness is the most fast one. For each node under scrutiny, we do not have to verify if other shapes can also be removed, contrary to the second and third one.

This attribute function $\mathcal{A}_\nu$ is related to the minimization of the piecewise-constant Mumford-Shah functional. It measures the resistance of a shape to be removed under the minimization problem of Eq (5.24). A bigger $\mathcal{A}_\nu(\tau)$ means that it is more difficult to remove the shape τ. Thus the attribute function $\mathcal{A}_\nu$ is also some kind of meaningfulness measurement deduced from the piecewise-constant Mumford-Shah functional.

5.4.3   Saliency map representing hierarchical selection

Following the idea of Section 3.6, we can compute first of all an attribute map $\mathcal{M}^\circ_{\mathcal{A}_\nu}$ using $\mathcal{A}_\nu$ described in Section 5.4.2 to illustrate the meaningfulness of the level lines in image. An example is illustrated in Figure 5.3 (b), where we can observe that the

visual salient level lines have great value $\mathcal{A}_\nu$ values, which signifies that they resist to be removed under by the minimization in Eq (5.24). However, in this attribute map, there are many close level lines are relative salient, but only one of them is desired. A salient map $\mathcal{M}_\mathcal{E}$ using the extinction value defined on the local minima of the following attribute function is used to fix that issue.

$$\mathcal{A}_\nu^\downarrow(\tau) \,=\, \max_{\tau'\in\mathcal{T}}\big(\mathcal{A}_\nu(\tau')\big) - \mathcal{A}_\nu(\tau). \tag{5.30}$$

As shown in Figure 5.3 (c), only one meaningful level line is presented to represent a salient level line in the image. This saliency map represents a hierarchical level lines selection. Each thresholding of this map $\mathcal{M}_\mathcal{E}$ selects salient (of certain degree) level lines from which a simplified image simplification is reconstructed as depicted in Figure 5.3 (d), (e), and (f) with respectively a small, median, and great thresholding value. Note that the level lines of the simplified images in (d), (e), and (f) obey the causality principle given by Definition 14.

5.5   Results

We first illustrate the importance of the level lines removal order in Section 5.5.1 through a comparison of the increasing order Eq (3.8) and decreasing order of Eq (5.28), and a random order. For the rest of the experiments, the shape meaningfulness we use to sort the level lines is the average of gradient magnitude along the level lines in Eq (3.8). In Section 5.5.2, we compare the proposed image simplification described in Section 5.3 with the method of Ballester et al. [Ballester 2007], which shows that our method achieves a similar simplification result, but with a significantly reduced time. Then the robustness of the proposed simplification method with respect to increasing Gaussian noise is illustrated in Section 5.5.3. Some hierarchical image simplification applied on the images of BSDS500 [Arbelaez 2011] are shown in Section 5.5.4. Then in Section 5.5.5, we apply the simplification method to color images in BSDS500. As the topographic map for a color image is still under development, we simply use the union of the selected salient level lines in the red, green, and blue channel. This process yields a pre-segmentation result having very few salient regions, and the boundaries between them remain intact. Finding an actual segmentation becomes a lot easier with such pre-segmentation. Finally, in Section 5.5.6, we apply the simplification method to count the autophagosomes which are usually presented as small and round structures with different contrast in cellular images. Quantitative assessment shows that this scheme achieves a comparable results compared with the number of autophagosomes counted by experts on a sequence of images.

(a)   (b)

(c)   (d)

(e)   (f)
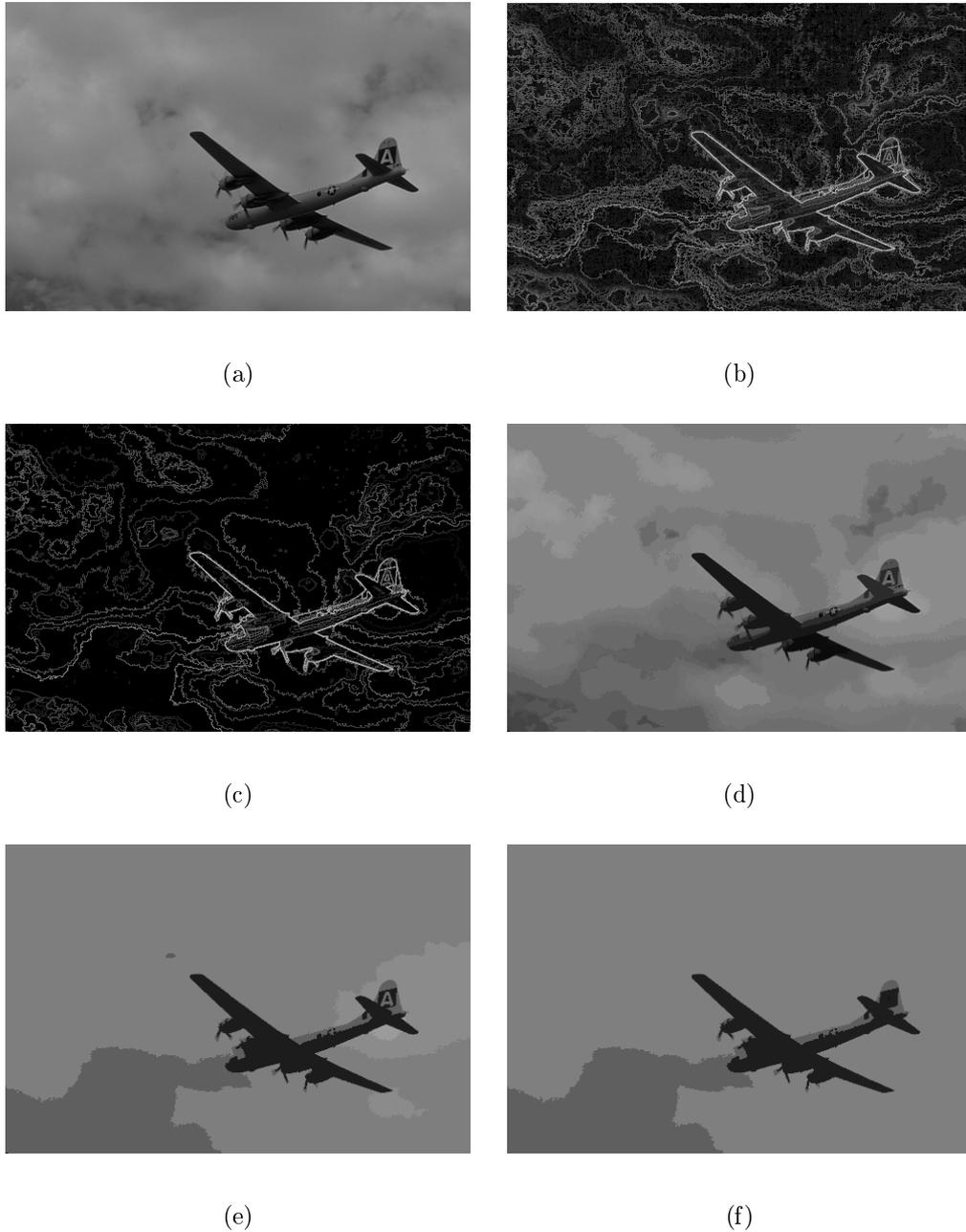
Figure 5.3: Illustration of saliency map representing the hierarchical salient level lines selection using shape meaningfulness increasing order. (a): input image; (b): attribute map defined on boundaries $\mathcal{M}^{\circ}_{\mathcal{A}_{\nu}}$; (c): saliency map $\mathcal{M}_{\mathcal{E}}$ representing the hierarchical salient level lines selection. (d-f): a simplified image reconstructed from the level lines obtained by thresholding the saliency map $\mathcal{M}_{\mathcal{E}}$ with respectively small, median, and great threshold value.

5.5.1   Importance of the pre-computed ordering

As discussed in Section 5.3, the level lines removal order is critical to the result of the greedy algorithm that computes a local optimum of Eq (5.24). To assess the importance of the pre-computed ordering, we begin with a random order of the level lines for the greedy algorithm described in Section 5.3.2. Then we compare the simplification result given by the random order with the ones obtained using respectively the decreasing, increasing order of Eq (3.8) (i.e., average of gradient's magnitude), and decreasing order of Eq (5.28) (i.e., average of curvature).

An example of such comparison applied on image shown in Figure 5.3 is depicted in Figure 5.4. For the random order used in (a), the meaningful level lines corresponding to the head of the plane is removed. In (b), although many level lines are selected using the decreasing order of average of gradient magnitude, many salient ones are removed. In (c) and (d), few level lines are selected, but the simplification results are more reasonable in the sense that they contain almost all the salient ones in image. This comparison confirms that the level lines removal order is very important, and the shape meaningfulness increasing order is a reasonable choice.

5.5.2   Comparison with the method of Ballester et al.

In Figure 5.5, we compare our proposed method with the one of Ballester et al. on a classic image "house" (256×256) having originally 23578 level lines. Qualitatively, the image (b) and (c) obtained by the two methods using the same parameter $\nu = 1000$ are very close. The result (Figure 5.5 (b)) given by the method of Ballester retains 30 level lines, and 27 level lines are selected by our approach. In spite of their strong simplifications, the salient structures are preserved in the simplified images. The distinct final functional minimized by the two approaches confirms that they fall into different local minimum. The CPU time for the Ballester method that we implemented and our approach are respectively 4s and 0.2s on a regular PC station with an Intel Core i7 3.40GHz and 8 Gbytes of memory.

5.5.3   Robustness to noise

Fig. 5.6 shows a quantitative evolution of our method w.r.t. increasing noise. The image "house" is corrupted with an additive Gaussian noise of variance $\delta^2 = 5, 10, 15, 20, 25$. All the results are obtained using the same parameter $\nu = 400$. Qualitatively, the simplification results from (a) to (f) are very close. They indeed select almost the same amount of level lines (around 32), and most of them coincide with the same salient structures in the image. Furthermore, the similar PSNR of each result also confirms the robustness of the proposed method w.r.t. noise.

(a) Random order (16 $\partial\tau$)   (b) Decreasing order of Eq (3.8) (290 $\partial\tau$)

(c) Increasing order of Eq (3.8) (8 $\partial\tau$)   (d) Decreasing order of Eq (5.28) (12 $\partial\tau$)
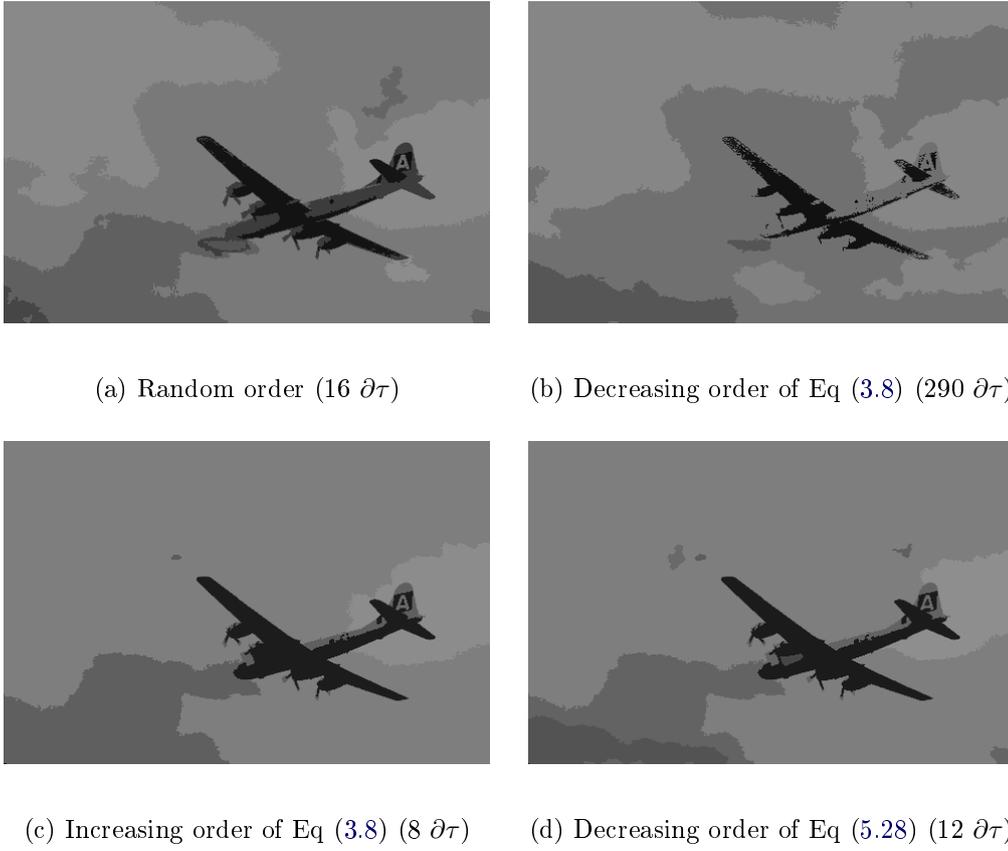
Figure 5.4: Comparison of the proposed greedy algorithm using different orders of the level lines. The parameter ν is set to $\nu = 1000$.

(a) Input image   (b) Ballester, $\nu = 1k$ (c) Our, $\nu = 1k$
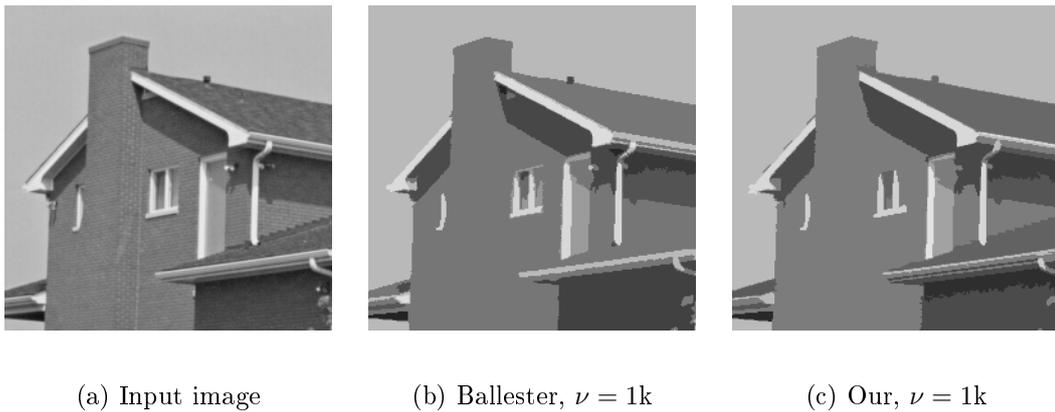
Figure 5.5: Comparison of our approach with the one of Ballester et al.. (a): $E = 1.2433e + 08$; (b): $E = 1.28113e + 07$, PSNR $= 27.3$; (c): $E = 1.15285e + 07$, PSNR $= 28.7$.

(a) Original image   (b) $\delta^2 = 5$ (c) $\delta^2 = 10$

(d) PSNR = 29.1   (e) PSNR = 28.8   (f) PSNR = 28.2

(g) $\delta^2 = 15$ (h) $\delta^2 = 20$ (i) $\delta^2 = 25$

(j) PSNR = 27.5   (k) PSNR = 26.7   (l) PSNR = 26.3.
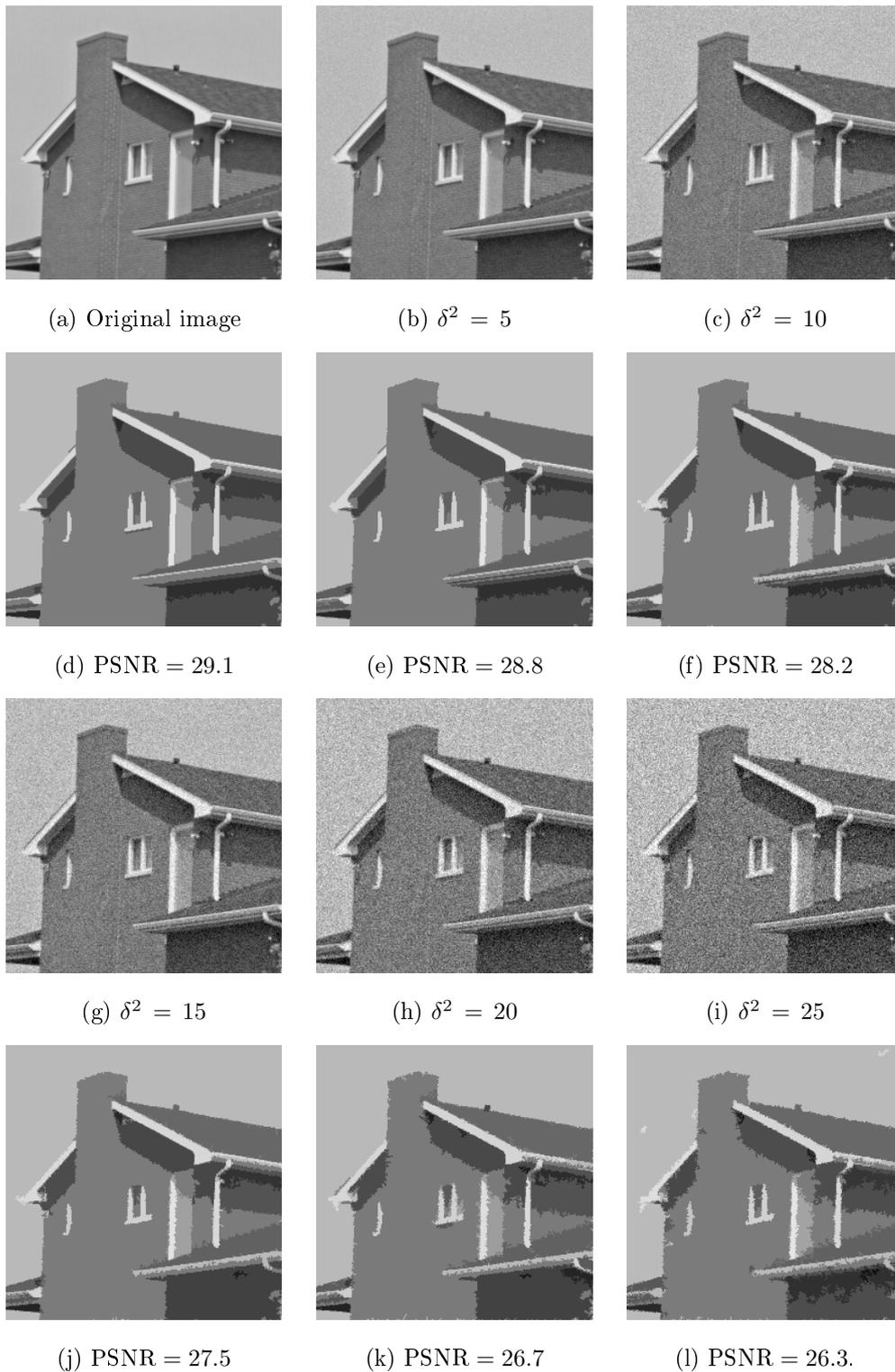
Figure 5.6: Qualitative behavior of our method w.r.t. increasing noise. (d-f, j-l):
Corresponding result applied to images of (a-c, g-i) with additive Gaussian noise of
variance going from 0 (b) to 25 (f) with a step of 5.

5.5.4 Hierarchical image simplification

In Section 5.4.2, we propose three manners to compute an attribute function $\mathcal{A}_\nu$ bearing the same idea of the image simplification method by solving Eq (5.24). Based on this attribute function $\mathcal{A}_\nu$, we have shown in Section 5.4 that we are able to compute a saliency map $\mathcal{M}_\mathcal{E}$ representing a hierarchical salient level lines selection. In Figure 5.7, we illustrate an example of the saliency map $\mathcal{M}_\mathcal{E}$ using the extinction values defined on the local minima of the three possible attribute function $\mathcal{A}_\nu$. For the image shown in Figure 5.7 (a), the corresponding saliency map $\mathcal{M}_\mathcal{E}$ is depicted respectively in (b) for the shape meaningfulness increasing order, in (d) for the shape meaningfulness and ν increasing order, and in (f) for the ν increasing order. Qualitatively, these three saliency maps are very similar. A simplified image reconstructed from the level lines obtained by thresholding the corresponding saliency map with the same threshold value is depicted on the right side. These simplified images are indeed quite close. Another example of saliency map representing a hierarchical image simplification is shown in Figure 5.7 (h-j) using the $\mathcal{A}_\nu$ computed with shape meaningfulness increasing order.

5.5.5 Color image pre-segmentation

In Fig. 5.8, we test our proposed method to color images from the Berkeley Segmentation Dataset BSDS500 [Arbelaez 2011], an extension of the BSDS300 [Martin 2001]. The strategy is to apply individually the proposed method to the red f_r, green f_g, and blue f_b channels. In order to obtain a pre-segmentation, a high parameter value $\nu = 2500$ is used, and a grain filter [Monasse 2000b] is applied to get rid of too tiny shapes. Less than 50 level lines are selected for each channel of each image, which results in a ratio of level lines selection around 2300. The regions formed by the union of those selected salient level lines from the 3 channels are considered as the pre-segmented regions. The images shown in Fig. 5.8 are obtained by taking the average color inside each region, where the boundaries between salient regions remain intact. Finding an actual segmentation becomes a lot easier with such a pre-segmentation. In fact, for a color image f_c, the salient level lines extracted from the three channels f_r, f_g, and f_b may have some segments that are very close but not identical. These close segments may yield many small regions around the meaningful boundaries in the pre-segmentation result of f_c. By merging these small regions to its most similar neighboring region, it results in a over-segmentation result which is almost the same as a pre-segmentation for visual perception. For those pre-segmentations in Figure 5.8, this last merging process yields an over-segmentation having around 100 regions.

(a) Input image   (b) Saliency map $\mathcal{M}_{\mathcal{E}}$ (c) PSNR = 25.28

(d) Saliency map $\mathcal{M}_{\mathcal{E}}$ (e) PSNR = 26.17

(f) Saliency map $\mathcal{M}_{\mathcal{E}}$ (g) PSNR = 26.37

(h) Input image   (i) Saliency map $\mathcal{M}_{\mathcal{E}}$ (j) PSNR = 20.10
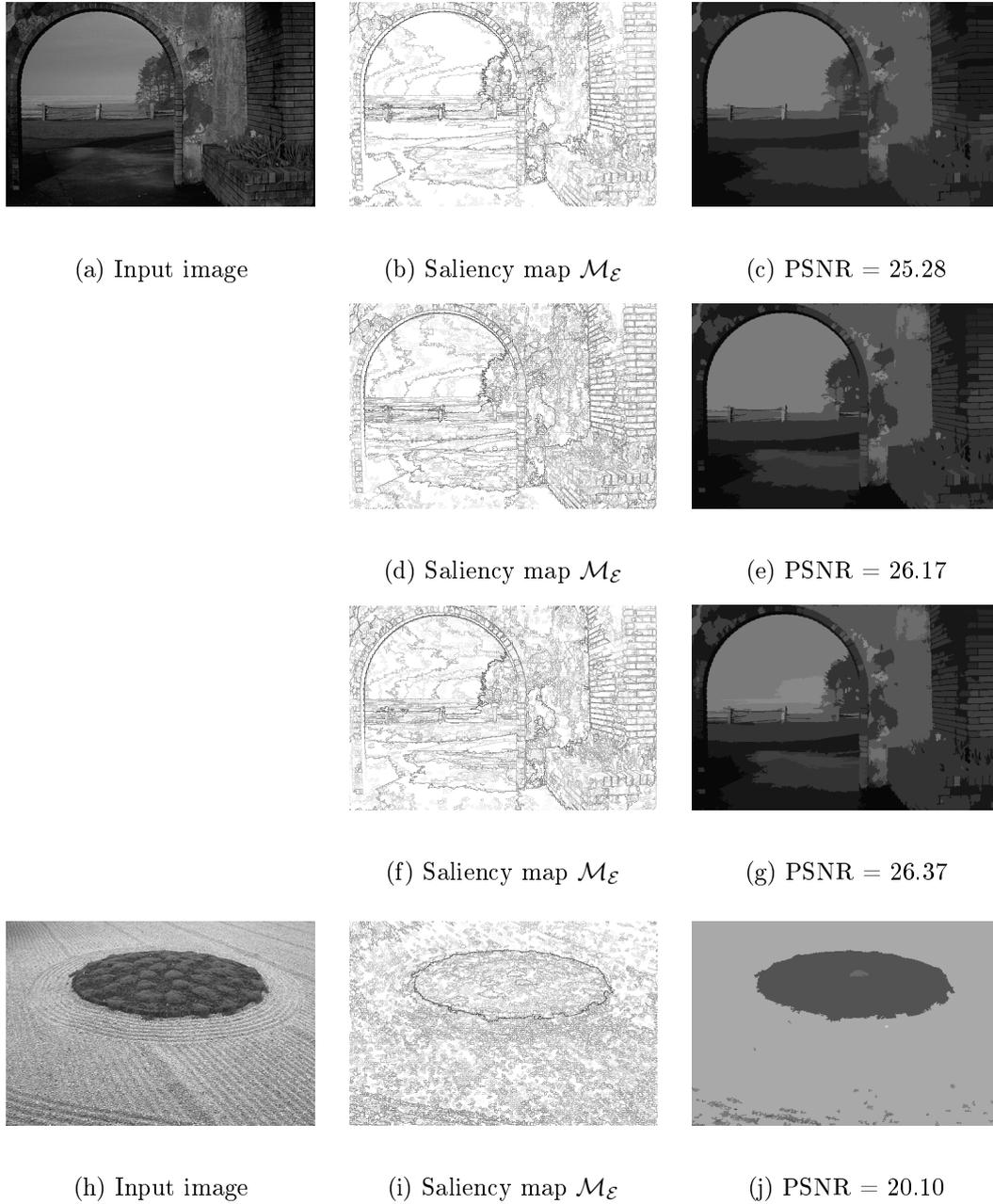
Figure 5.7: Illustration of hierarchical image simplification with the 3 proposed $\mathcal{A}_{\nu}$ computations: shape meaningfulness increasing in (b); shape meaningfulness increasing and ν increasing in (d); ν increasing in (f). Right column: corresponding simplified image reconstructed from the selected level lines given by thresholding the saliency map in middle column with same value. (h-j): another example of hierarchical image simplification using the shape meaningfulness increasing order.
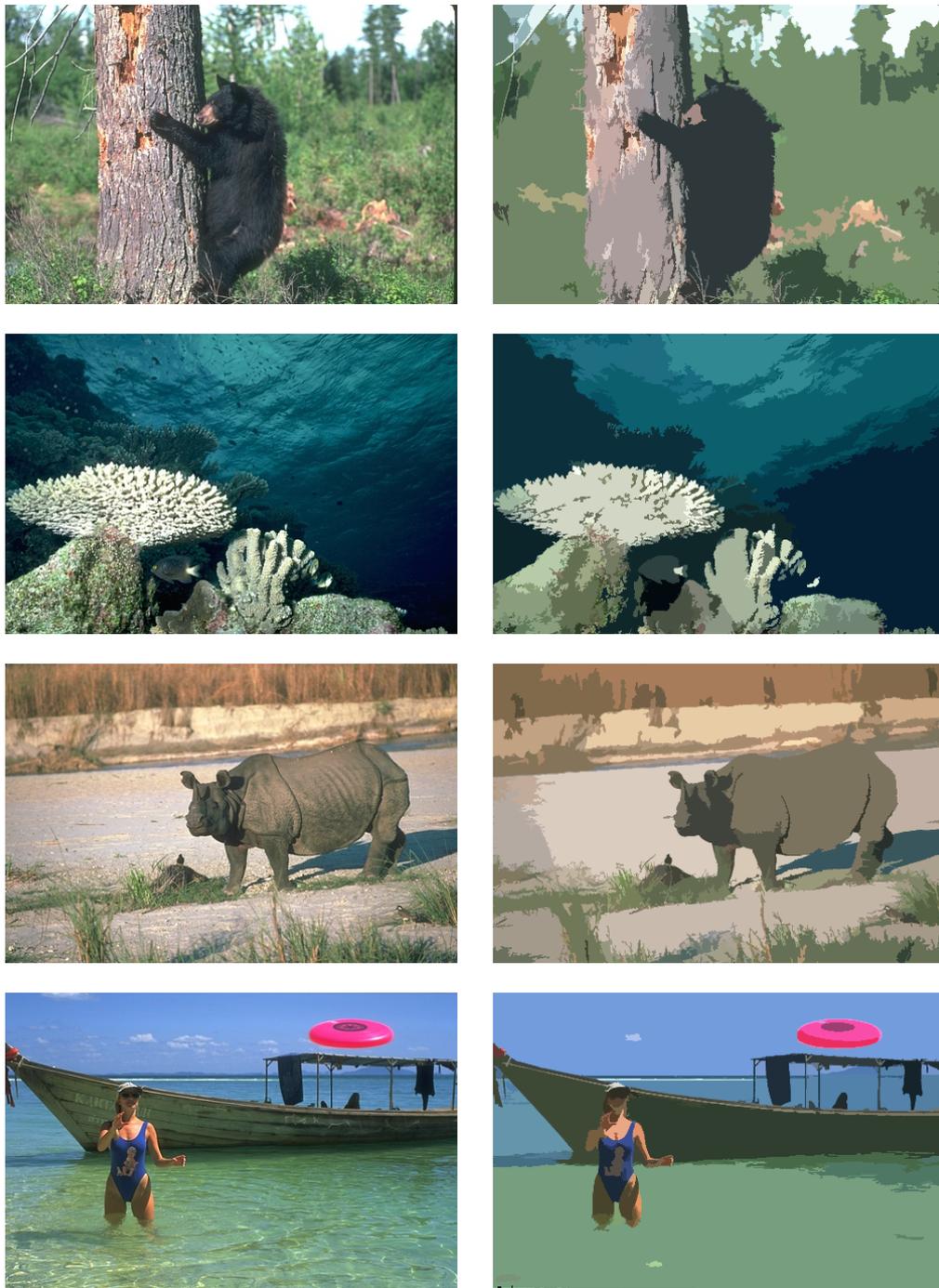
Figure 5.8: Some pre-segmentation results obtained with our proposed method on the Berkeley Segmentation Dataset BSDS500 [Arbelaez 2011]. Left column: input image; Right column: pre-segmentation obtained with the simplification method.
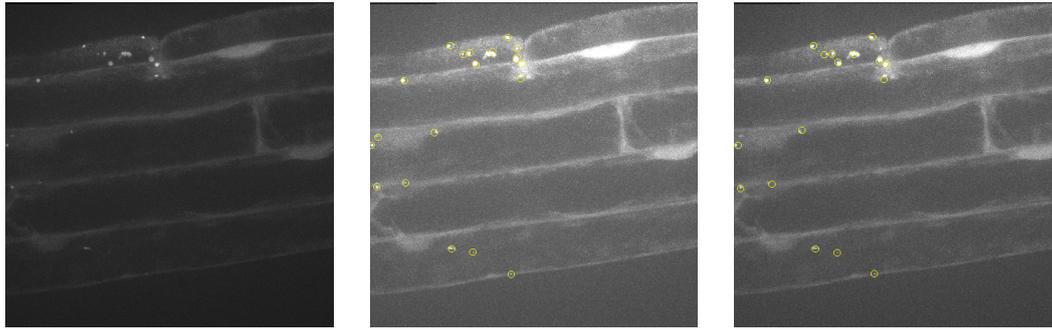
| (a) Original image | (b) Less stringent counting | (c) More stringent counting |

Figure 5.9: An example of autophagosomes counting achieved by an expert for the cellular image in (a). The autophagosomes are surrounded with yellow circles. There are 19 autophagosomes for the less stringent counting in (b), and 17 autophagosomes for the more stringent counting in (c).

5.5.6   Application to autophagosome counting

The autophagosome is a cellular compartment whose formation is tightly regulated during development and stress responses in all eukaryotic cells (from yeast to animals and plants). Its formation allows to embark portion of the cytosol (cytosol is the cell content eluding the nucleus and other compartments, it is found often in the images as "long and thin objects", e.g., the bright long and thin structures in Figure 5.9 (a)) for bulk degradation and recycling of resources, a process known as Autophagy (self-eating). The level of autophagy (e.g. in response to stress or drugs) can be evaluated by counting autophagosomes (small and round "bright" objects in Figure 5.9 (a)) present in a cell, Several proteins are involved in this process, the ATG proteins. The protein ATG8 is present on the membranes of the autophagosomes at all stages of its life, so that ATG8 proteins made fluorescent by genetic engineering are widely used as reliable "markers" to identify and study autophagosomes by fluorescent microscopy. Counting of autophagosomes is often problematic and time-consuming, due to the fact that an important fraction of the ATG8 protein is also present diffusely in the cytsol and in the nucleus. A counting example is given in Figure 5.9, where the autophagosomes are surrounded with yellow circles. The Figure 5.9 (b) and Figure 5.9 (c) correspond to respectively a less stringent and a more stringent counting. This work of the application to autophagosome counting is joint with Michele Bianchi.

We propose to use the simplification method described in Section 5.3 to automatically count the autophagosomes in a cellular image f. Observe that in a cellular

image (see Figure 5.10 (a)) the autophagosomes are presented as "bright" small and round objects, and the bright, long and thin structures presented also in image f are the unwanted objects. Thus we first apply a shape-based upper leveling defined in Section 3.4.4 to remove those bright, long and thin structures (see Figure 5.10 (b)). The attribute function for each node $\mathcal{N}$ used in this shape-based upper leveling is an elongation based measurement defined as bellow:

$$\mathcal{A}_e(\mathcal{N}) = \frac{|\mathcal{N}|}{\pi \times l_{\max}^2(\mathcal{N})}, \tag{5.31}$$

where $|\mathcal{N}|$ denotes the area of the region represented by the node $\mathcal{N}$, and $l_{\max}$ denotes the major axes of the best fitting ellipse having the same moments as the underlying region. The attribute $\mathcal{A}_e$ is low for the long and thin structures. For the shape-based upper leveling, we simply filter those nodes whose attribute $\mathcal{A}_e$ is under a certain threshold value. A filtered image f_s (see Figure 5.10 (b)) is reconstructed using the subtractive rule (see Section 2.5.3). Then we apply the method of salient level lines selection described in Section 5.3 to the filtered image f_s, which yields a simplified image f_s' (see Figure 5.10 (d)). In this simplified image f_s', only few level lines are selected, but note that there may have several level lines that are very close and represent actually a same object. As in a cellular image f, two different autophagosomes are always disjoint, we group those close level lines and also remove those dark level lines, which leads us to a final counting as shown in Figure 5.10 (f) for the image in Figure 5.10 (a). The Figure 5.10 illustrates an example of such scheme of autophagosome counting, where the final counting result is very similar the less and more stringent counting in Figure 5.9 (b) and (c), in fact it counts 18 autophagosomes in this image compared with 19 autophagosomes for less stringent counting, and 17 for more stringent counting.

Some results for a sequence of cellular images are shown in Figure 5.12. Qualitatively, the counting result obtained with the scheme relying on the proposed simplification method is quite close to the less stringent and more stringent counting achieved by an expert. The number of autophagosomes counted with this scheme is also between the interval of more stringent counting and less stringent counting given by an expert.

A quantitative evolution of the proposed scheme relying on the proposed simplification method is illustrated in Figure 5.12. It is applied on a sequence of images for which a less stringent and more stringent counting by an expert are available. The evolution in Figure 5.12 shows that this scheme achieves a result comparable to the ground truth, which demonstrate the usefulness of the proposed simplification method.

(a) Original image f (b) Shape filtering f_s (c) Zoomed details of f_s

(d) Simplified image f_s' (e) Zoomed details of f_s' (f) Final counting
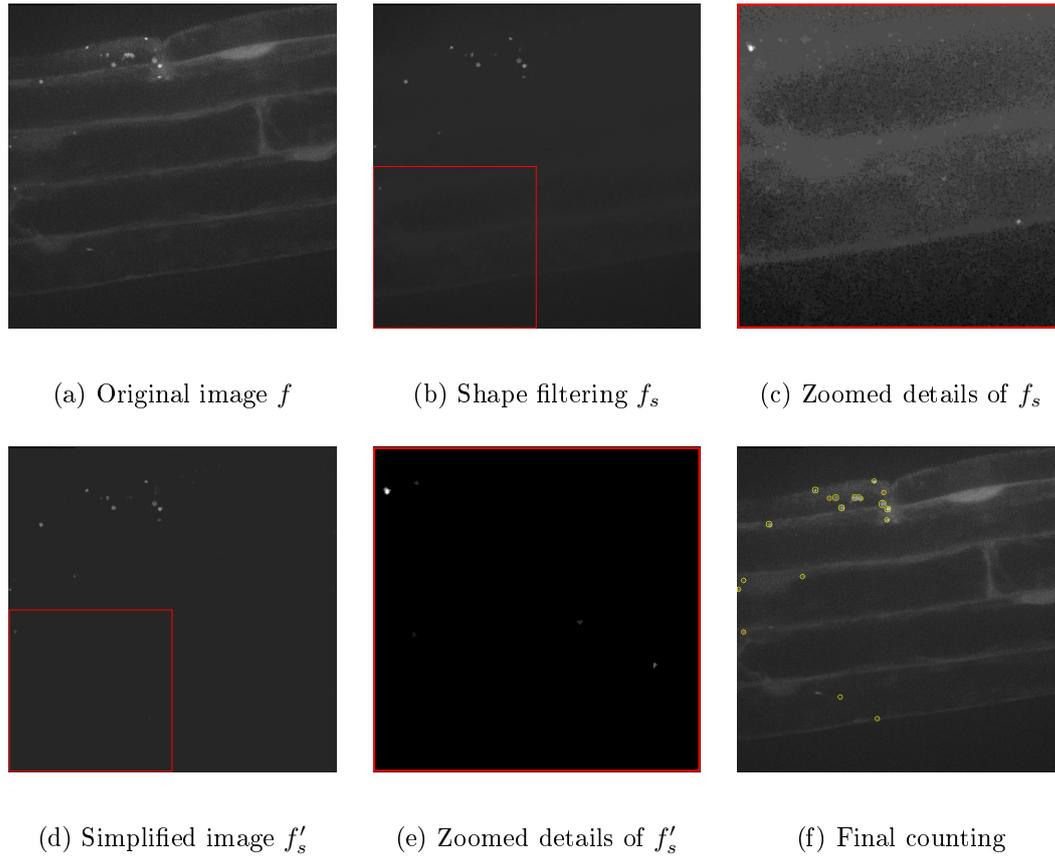
Figure 5.10: Illustration of autophagosome counting. In the output of the shape-based upper leveling (b) using the attribute $\mathcal{A}_e$ in Eq (5.31), there are still many level lines as shown in (c) being the zoomed and normalized details of the red block in (b). Whereas, in the simplified image f_s', the background is uniform as shown in (e). There are 18 autophagosomes in the final counting in (f).

(a) 9 autophagosomes · (b) 7 autophagosomes · (c) 8 autophagosomes

(d) 11 autophagosomes · (e) 6 autophagosomes · (f) 6 autophagosomes

(g) 35 autophagosomes · (h) 26 autophagosomes · (i) 26 autophagosomes
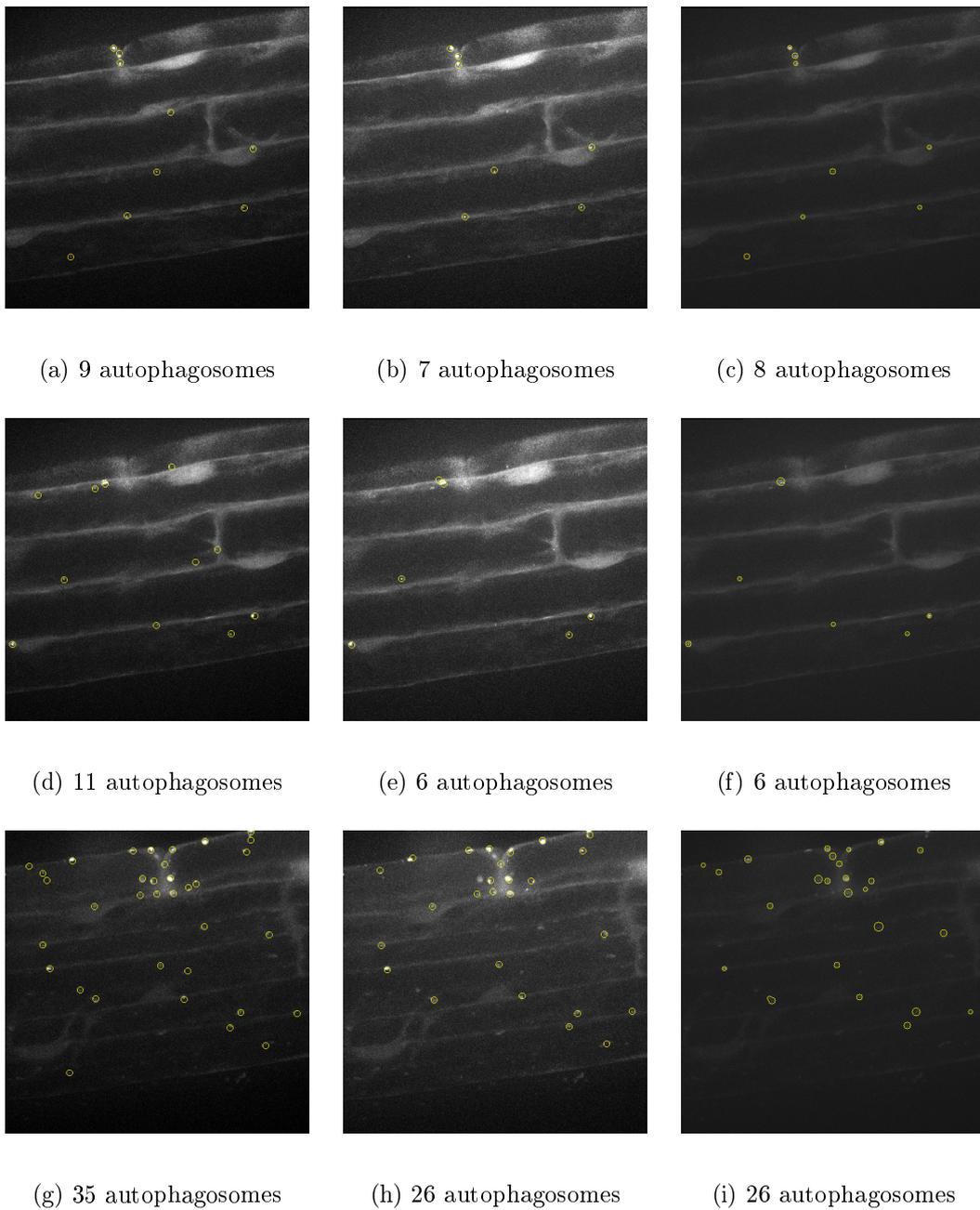
Figure 5.11: Some results of the autophagosome counting using the proposed simplification method. Left column: Less stringent counting by an expert; Middle column: More stringent counting by an expert; Right column: Result of our counting.
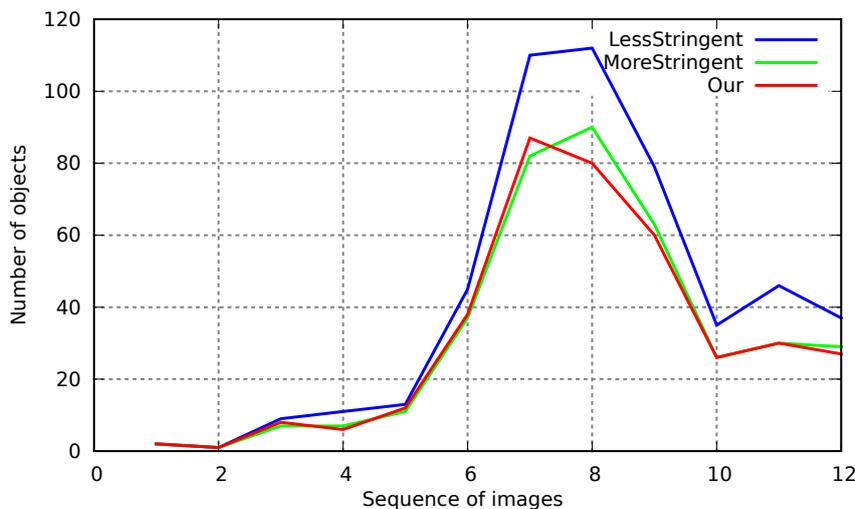
Figure 5.12: A quantitative evolution of the scheme relying on the proposed simplification method for automatic autophagosomes counting on a sequence of images. Our result represented by the red curve is close to the less stringent counting in blue and more stringent counting in red.

5.6 Conclusion

In this chapter, we presented an efficient morphological shaping to salient level lines selection, based on the minimization of the piecewise-constant Mumford-Shah functional. Our major contribution is to rely on a *meaningful* ordering of level lines in order to minimize this energy functional on the tree of shapes. As a consequence, the proposed greedy algorithm converges to a relevant local optimum very quickly compared with the similar work of Ballester et al.. We have shown that the proposed method allows for strongly simplifying images while preserving their salient structures. We have seen that a strong property of our proposal is its robustness to noise. Furthermore simplification results can be used as pre-segmentations that are suitable for object recognition, scene analysis, or practical shape matching [Lisani 2003]. The application to autophagosome counting in cellular images demonstrates also its usefulness and robustness. We have also proposed a hierarchical version of this simplification method via the saliency map obtained with the transformation of hierarchy in the shape-based morphology.

Shape filtering

Connected filters are well-known for their good contour-preservation property. A popular implementation strategy relies on tree-based image representations detailed in Section 2.5. The filtering is performed by thresholding the tree based on some attribute function $\mathcal{A}$ characterizing the connected component represented by each node. Rather than being satisfied with a mere thresholding, we propose the shape-based morphology described in Chapter 3 by expanding this idea and applying connected filters on tree-based shape space. For the filtering purpose, it is a generalization of the existing tree-based connected operators. Indeed, it has been show in Chapter 3 that the framework includes the classical existing connected operators by attributes, and two classes of novel connected operators have also been introduced in the framework of shape-based morphology. They are respectively shape-based lower/upper levelings and morphological shapings. In this chapter, we will show several applications of these shape-based filters including retinal image analysis, and illustration of morphological shapings. Quantitative evaluations demonstrate that some simple shape-based filters, as compared to more evolved processings, achieves state-of-the-art results.

6.1 Introduction

Mathematical morphology, as originally developed by Matheron and Serra [Serra 1982], proposes a set of morphological operators based on structuring elements. Later, Salembier and Serra [Salembier 1995], followed by Breen and Jones [Breen 1996], proposed morphological operators based on attributes, rather than on elements. Such operators, also known as attribute filters or connected filters, have been popularized by Salembier, Wilkinson, and Ouzounis [Salembier 2009, Westenberg 2007, Wilkinson 2008, Ouzounis 2007, Ouzounis 2011b]. One popular implementation of such operators relies on transforming the image into an equivalent representation, generally a tree of components (*e.g.*, level sets) of the image; such trees are equivalent to the original image in the sense that the image can be reconstructed from its associated tree. Filtering then involves the design of a shape attribute that weights how much a node of the tree fits a given shape.

Two different approaches for filtering the tree (and hence the image) have been proposed: the more evolved approach consists in pruning the tree by removing some entire branches of the tree, and is easy to apply if the attribute is increasing on the tree. However, most shape attributes are not increasing. When the attribute is not increasing, three pruning strategies have been proposed (Min, Max, Viterbi; see Section 2.5.2 and [Salembier 2009] for more details). Another type of strategy is to simply remove the nodes of the tree for which the attribute is lower than a given threshold.

The framework of shape-based morphology described in Chapter 3 shares the same idea as the thresholding based strategy of classical tree-based connected operators. But we propose to apply connected filters on the tree-based shape space, instead of directly on the space of image (see the black+red path of Figure 3.2). It has been show that a straightforward of this surprising and simple idea is that it gives more flexibilities and possibilities to deal with the non-increasing attribute, which is usually the case for shape attributes. Indeed, It is shown that this framework encompasses some usual attribute filtering operators, and a novel connected filters based on non-increasing criterion is introduced from the family of levelings called shape-based lower or upper levelings. Another class of novel connected filters named morphological shapings is also introduced in the framework of shape-based morphology.

In this chapter, we focus on filtering aspect of the framework of shape-based morphology. Some applications of those simple shape-based filters are studied in this chapter. First of all, we apply the shape-based lower/upper levelings to retinal image analysis, including blood vessels segmentation and optic nerve head (ONH) segmentation. The benchmark of different approaches of blood vessels segmentation is based on two databases, DRIVE [dri 2013b], [Staal 2004] and STARE [sta 2013], [Hoover 2000]. The results of ONH segmentation are compared with some other more evolved approaches using the DRI-ONS [dri 2013a], [Carmona 2008] database. In both cases, quantitative evaluations demonstrate that shape-based filtering achieves some state-of-the-art results. And last, we illustrate an application of the morphological shapings.

The rest of this chapter is organized as follows. In Section 6.2, we briefly review a number of dedicated methods for retinal image analysis, including blood vessels segmentation and optic nerve head (ONH) segmentation. In section 6.3, we will detail how to apply the framework of shape-based morphology to blood vessels segmentation and optic nerve head segmentation in retinal images. Section 6.4 gives a set of experimental results, including applications of the proposed shape-based lower/upper levelings to both blood vessels segmentation and ONH segmentation in

retinal images, and a comparison of morphological shapings with classical threshold-based approaches. Finally we conclude in Section 6.5.

6.2 Related work

In this section, we focus on some related work about the applications of the two classes of novel connected operators given by the framework of shape-based morphology. In Section 6.2.1, some dedicated methods for the retinal image analysis, such as blood vessels segmentation, optic nerve head segmentation are firstly reviewed respectively in Section 6.2.1.1 and Section 6.2.1.2.

6.2.1 Retinal image analysis

The retinal microvasculature is the very unique region of the human body that can be directly visualized non-invasively *in vivo*. The retina can be photographed relatively straightforwardly with a fundus camera. Several eye fundus images features are commonly evaluated in computer aided-diagnosis of several retinopathologies, such as diabetic retinopathy, age-related macular degeneration and retinopathy of prematurity [Patton 2006]. The automated diagnostic systems based on some interesting eye fundus images features have been much developed over the past 20 years, which provides the potential for significant resource savings in large-scale screening programs, as well as being free from observer bias and fatigue. We will focus on two types of features: the retinal blood vessels and the optic nerve head (ONH).

6.2.1.1 Blood vessels segmentation

The retinal vascular segmentation can be useful, for example, to measure the narrowing, length, tortuosity, and branching pattern of retinal vessels. The segmentation of blood vessels is important for the detection of non-proliferative diabetic retinopathies, such as venous beading [Bade 2007, Vallabha 2004], and neovascularizations [Ciulla 2003, Kauppi 2007]. Moreover, Blood vessels bifurcations are also useful providing points in the retinal image registration and comparison process [Laliberté 2003, Markaki 2009]. An example of an eye fundus image and its retinal blood vessels segmentation is shown in 6.1.

There are many methods for segmentation of retinal blood vessels in the literature. We only review those methods that we will compare with, including six pixel processing-based methods, a tracking-based method, and a method based on active contour model. The pixel processing-based methods usually feature a common process consisting of two steps. The first step is an enhancement procedure (usually a
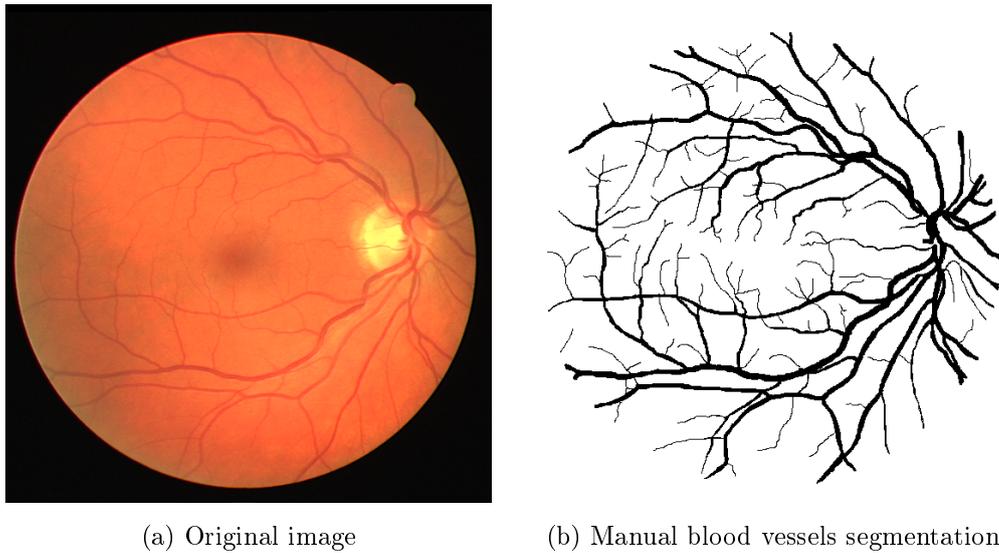
(a) Original image   (b) Manual blood vessels segmentation

Figure 6.1: An eye fundus image (a) of DRIVE [dri 2013b] database and one manual segmentation of retinal blood vessels (b).

convolution operator) used to select an initial set of pixels to be further validated as vessels in the second step. The first method of this type is the matched filter proposed in [Chaudhuri 1989], where the authors use 12 rotated versions of a 2-D Gaussian shaped template for searching vessel segments along all possible directions. For each pixel, the maximum response to these kernels is retained. As the mean value is subtracted from each convolution kernel, the output of this method is ideally zero in the background, and a blood vessels segmentation result can be obtained by thresholding this output image. In [Hoover 2000], Hoover et al. proposed also a matched filtering approach using local thresholding strategy to segment the retinal vessels. Marinez-Pérez et al. proposed in [Marinez-Pérez 1999] a method based on a two-stage region growing procedure using the features derived from image derivatives obtained at multiple scales. It segments progressively the retinal vessels. Then in [Zana 2001], the authors proposed an algorithm that combines morphological filters and cross-curvature evaluation to segment vessel-like structures. Jiang et al. proposed in [Jiang 2003] a general framework of adaptive local thresholding using a multi-threshold scheme, combined with a classification procedure incorporating some relevant information related with retinal vessels to verify each resulting binary object. In [Niemeijer 2004], Niemeijer et al. proposed a supervised method based on a pixel classification using a simple feature vector. They use a K-nearest neighbor (kNN) classifier using the features extracted form the green channel of the color eye fundus images. It outputs a soft classification that indicates the proba-

(a) Original image   (b) Manual ONH segmentations
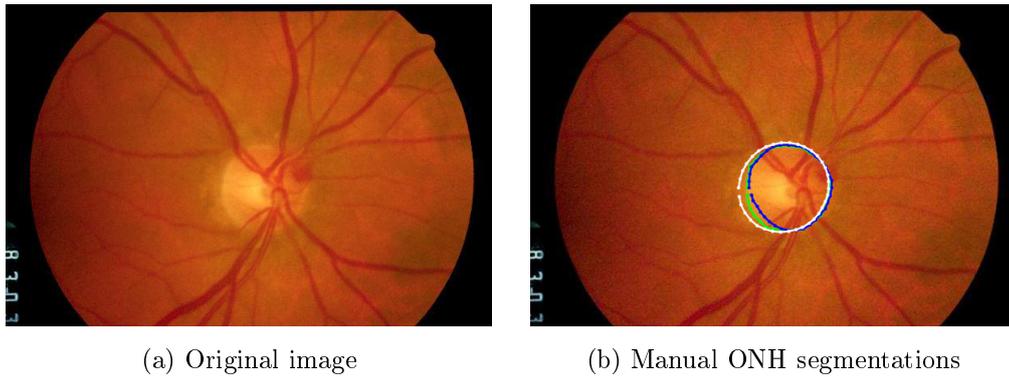
Figure 6.2: An eye fundus image (a) of DRIONS [dri 2013a] database and two manual ONH segmentations given by blue and green boundaries in (b).

bility of each point to be vessel point. Another supervised approach is proposed in [Staal 2004]. In [Mendonca 2006], the authors proposed a tracking-based method that starts with the extraction of centerlines, which are used as guidelines for the subsequent filling phase. The final retinal vessel segmentation is obtained using an iterative region growing method that integrates the contents of several binary images resulting from vessel width dependent morphological filters. Lastly, Al-Diri and Steel proposed an active contour model based method for segmenting the retinal blood vessels [Al-Diri 2009].

6.2.1.2 Optic nerve head (ONH) segmentation

The optic nerve head (ONH), also called optic disk or papilla is oval-shaped and is located in the area where all the retina nerve fibers come together to form the start of the optic nerve that leaves the black of the eyeball. The localization and segmentation of the ONH is of critical importance, which can be used qualify retina lesions, like the exudate lesions that characterize the diabetic macular edema (DME) [Ciulla 2003]. Moreover, it is possible to diagnose glaucoma [Hoffmann 2007] based on the size and shape of the ONH outer boundaries. An example of the segmentation of ONH in eye fundus image is illustrated in Figure 6.2.

Many methods are proposed for ONH segmentation, such as mathematical morphology based approach [Welfer 2010], Gabor filters based method [Rangayyan 2010], and Hough transform based approach [Zhu 2010]. But we only give a short review of several methods that make use of the elliptical model of ONH. The first one is the approach proposed by Lowell et al. in [Lowell 2004], where the authors use a specialized template matching to localize the ONH, and segmentation of ONH is achieved by a deformable contour model using a global elliptical

model and a local deformable model with variable edge-strength dependent stiffness. In [Carmona 2008], the authors proposed a method to identify the ONH contour with genetic algorithms in two steps. First of all, they extract a set of hypothesis interest points (IPs) exhibiting geometric properties and intensity levels similar to the ONH contour pixels. Then a genetic algorithm was used to find an ellipse containing the maximum number of hypothesis points in an offset of its perimeter. This elliptical region is finally considered as the segmented ONH. Molina and Carmona proposed in [Molina 2011] a variant of the method in [Carmona 2008]. The main difference between them lies in how to obtain the set of IPs. In [Carmona 2008], the set of IPs are obtained using a domain-knowledge-based method. Whereas, in [Molina 2011], they used a Laplacian pyramid of the eye fundus image to obtain the set of IPs.

6.3   Shape filtering for retinal image analysis

The retinal blood vessels are usually long and thin structures in an eye fundus image, it can be characterized by some shape attribute (e.g., the elongation defined in Eq (5.31)). Besides, the optic nerve head can be usually approximated by an elliptical region, which can also be described by a shape attribute. Moreover, unlike classical linear filters, connected operators preserve the location and the shape of the contours. All these provide us a motivation for using the shape-based filters to blood vessels segmentation (see Section 6.3.1) and ONH segmentation (see Section 6.3.2) in retinal image analysis.

6.3.1   Blood vessels segmentation

Many existing methods of retinal blood vessels segmentation work on the green channel of the color retinal image. To improve the visibility of the blood vessels, for each color retinal image f_c, a black top-hat transform is applied to the green channel f_g. When a mask of eye fundus is available, we combine it with the black top-hat f_t. We thus obtain an image f_i in which the blood vessels are visible: indeed, the main structures of the blood vessels are present in the Max-tree $\mathcal{T}$ representation of f_i.

   For each connected component represented by some $\mathcal{N}$ of the Max-tree $\mathcal{T}$, we compute a shape attribute $\mathcal{A}$ characterizing the blood vessels, which are usually long and thin structures. The attribute used here is the elongation $\mathcal{A}_e$ defined in Eq (5.31). Since the blood vessels are long and thin, nodes having a low attribute $\mathcal{A}_e$ correspond to the blood vessels.

   The core of this application is the filtering of the Max-tree $\mathcal{T}$. A mere thresholding of the elongation $\mathcal{A}_e$ is not sufficient, often giving unwanted objects (noise).

(a) Input color image f_c (b) Green channel f_g (c) Inversed black top-hat f_i

(d) Inversed filtering result   (e) Segmented blood vessels   (f) Manual segmentation
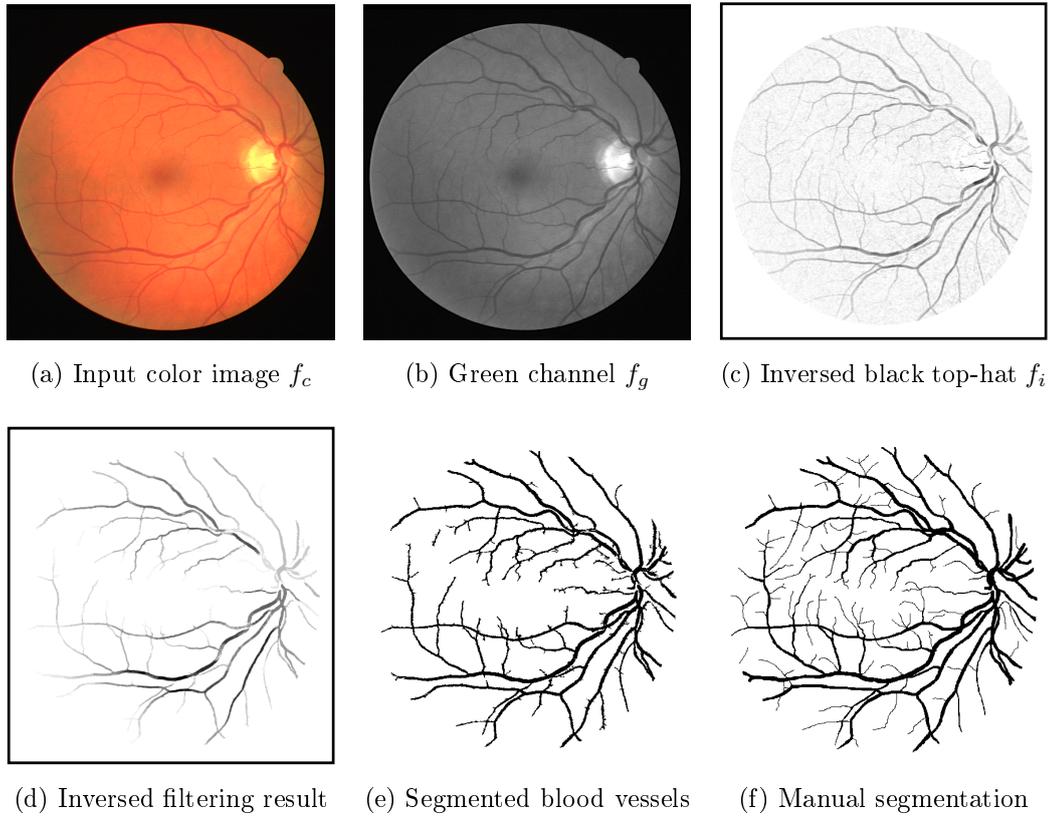
Figure 6.3: Illustration of the complete process of blood vessels segmentation in retinal image using elongation-based upper leveling (a-e); (f): Manual segmentation.

However, a very low thresholding value t_{min} on $\mathcal{A}_e$ ensures that thresholded nodes are blood vessels. These initial extracted nodes are used as seeds in the sequel. We then apply a morphological filtering with a depth criterion: using the Min-tree $\mathcal{TT}$ of the node weighted graph $(G_{\mathcal{T}}, \mathcal{A}_e)$, we only preserve the nodes that have a certain depth d_0 in $\mathcal{TT}$ and that furthermore contain the seeds. The connected components represented by the preserved nodes are considered as segmented blood vessels. The whole process is one of the many variant of shape-based upper levelings.

An example of such a blood vessels segmentation process is given in Figure 6.3. Compared with the manually segmented blood vessels segmentation (Figure 6.3 (f)), the elongation-based upper leveling (Figure 6.3 (e)) correctly segments most of the blood vessels.

6.3.2   Optic nerve head (ONH) segmentation

In retinal image analysis, another critical task is the optic nerve head (also called optic disk or papilla) localization and segmentation. The segmented blood vessels can be used to locate the optic disk [Welfer 2010]. Besides, the optic disk has an elliptical shape [Lowell 2004], [Carmona 2008], [Molina 2011], which motivates us to use the shape-based morphology.

The optic disk is usually visible, but separated into several parts by the blood vessels in the red channel f_r of the color eye fundus image f_c. Unlike the work of Welfer et al. [Welfer 2010], we make use of the segmented blood vessels as pre-processing. More specifically, the method presented in Section 6.3.1 using the proposed elongation-based upper leveling provides the segmented blood vessels f_b. For those pixels belonging to the blood vessels in f_r, they take the value of the closest non blood vessel pixel. We thus obtain a modified image f_i in which the optic disk presents as a single object. In fact, the main structure of the optic disk is present as a shape in the topographic map representation $\mathcal{T}$ representation of f_i.

Once again, we can compute a shape attribute $\mathcal{A}$ to characterize the connected components represented by the nodes of the tree of shapes $\mathcal{T}$. As the optic disk can be modeled by an elliptical shape, we thus use a shape attribute $\mathcal{A}_{ell}$ measuring how much a connected component fits an elliptical shape. For each node $\mathcal{N}_k$ on the tree of shapes $\mathcal{T}$, $\mathcal{A}_{ell}$ is given by

$$\mathcal{A}_{ell}(\mathcal{N}_k) = |\mathcal{N}_k|/(\pi \times l_1 \times l_2) \tag{6.1}$$

Where $|.|$ denotes the cardinality, l_1 and $_2$ denotes the length of two principal axis of the best fitting ellipse for the connected component represented by $\mathcal{N}_k$. The nodes having a high value of such attribute $\mathcal{A}_{ell}$ are good candidates to be the optic disk. However there may exist many shapes on the tree $\mathcal{T}$ which fit very well the elliptical model. The optic disk corresponds to the one being bright and well contrast. Besides, the size of the optic disk is normally in a fixed range. It can not be extremely small nor extremely large. And each eye fundus image is taken so that the optic disk is near to the center of scene. Bearing in mind these prior features, the specific attribute $\mathcal{A}_{onh}$ used for this task is based on the combination of these information and the shape attribute $\mathcal{A}_{ell}$ given by Eq (6.1) using the fuzzy theory [Bloch 1996].

This application can be seen as the best node selection on the tree $\mathcal{T}$. The node $\mathcal{N}_h$ having the highest value of attribute $\mathcal{A}_{onh}$ is identified as the best candidate. And its best fitting ellipse is considered as the ONH. In this application, the shape-based upper leveling helps to make the best candidate of optic disk present on the tree structure. However, in order not to make the decision immediately, a morphological

(a) Input color image   (b) Green channel   (c) Obtained vessels   (d) Red channel

(e) Before filtering   (f) Filtered image   (g) Detected CC   (h) ONH contours
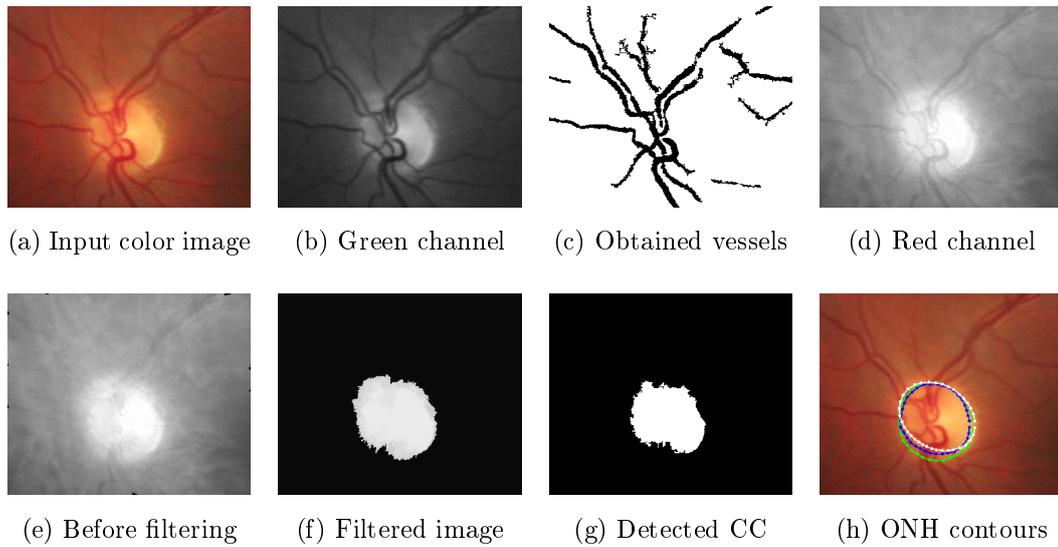
Figure 6.4: Illustration of the complete process of the proposed ONH segmentation (a-e); (f): Our segmented ONH contour (best fitting ellipse of f) in white compared with the manually segmented ONH contours in green and blue traced by two experts.

filtering using the Max-tree $\mathcal{TT}$ of the node weighted graph $(G_{\mathcal{T}}, \mathcal{A}_{onh})$ may give interesting result f_o for further processing. The strategy is to only preserve the nodes that have a certain depth in $\mathcal{TT}$ and that furthermore contain the node $\mathcal{N}_h$.

An example of such an optic nerve head segmentation is shown in Figure 6.4. The ONH segmented by our approach (white contour in Figure 6.4 (h)) is very close to the two manual segmentations (red and green contours in Figure 6.4 (h)).

6.4 Results

In this section, we will show some results of applications using those shape-based filters introduced in the framework of shape-based morphology (see Section 3.4). Firstly, in Section 6.4.1, we will illustrate some qualitative results of the retinal blood vessels segmentation described in Section 6.3.1, which shows that the segmented retinal vessels are quite close to the manual segmentations. Furthermore, quantitative evaluation on two public datasets of eye fundus images shows that this "simple" filtering step achieves a accuracy on par with state-of-the-methods. Then some qualitative ONH segmentation results will be illustrated in Section 6.4.2, which shows that the segmented ONH boundaries are very close to the manual segmentations. The quantitative evaluation using the discrepancy measurement on a public eye fundus images dataset also confirms the good performance of the ONH segmen-

tation method described in Section 6.3.2. Finally, we will present an example of morphological shapings $\mathcal{S}$ compared with the classical threshold-based strategy in Section 6.4.3.

6.4.1   Blood vessels segmentation in retinal images

We tested the described elongation-based upper leveling in Section 6.3.1 on the database of Digital Retinal Images for Vessel Extraction (DRIVE) [dri 2013b], [Staal 2004] and on the database of STructured Analysis of the Retina (STARE) [sta 2013], [Hoover 2000]. DRIVE is a database assembled in the Netherlands from a diabetic retinopathy screening program. It includes 40 color fundus images of 584×565 pixels, captured using a $45°$ field-of-view fundus camera. The 40 color images are divided into a training and a test set, both containing 20 images. For the training images, a single manual segmentation of the vasculature is available. For the test cases, two manual segmentations are available; one is used as gold standard, the other one can be used to compare computer generated segmentations with those of an independent human observer. The complete database contains seven pathological cases (four on the test set and three on the training set). The STARE database contains 20 images among which ten images are abnormal. These retinal images were captured using a TopCon TRV-50 fundus camera at $35°$ FOV, and digitized to 700×605 pixels, 8 bits per RGB channel. A manual segmentation is available for each image of the database. Masks of the eye fundus, derived from the matched spatial filter [Hoover 2000], are also available.

Figure 6.5 and 6.6 show four segmentation results respectively from DRIVE database and STARE database. Qualitatively, most of the blood vessels are correctly extracted, although some noise at the end of the vessels is also extracted, and some very thin blood vessels are missed.

Quantitative assessment is based on three performance measurements named respectively *sensitivity*, *specificity* and *accuracy* [Staal 2004]. Sensitivity measures the true positive rate (TPR), specificity measures the true negative rate (TNR), accuracy measures the rate of pixels correctly classified. Another performance measurement named *kappa* values are also used in [Staal 2004] for the DRIVE database. It is a statistical measure of inter-rater agreement, which measures the agreement between two raters who each classify N items into C mutually exclusive categories.
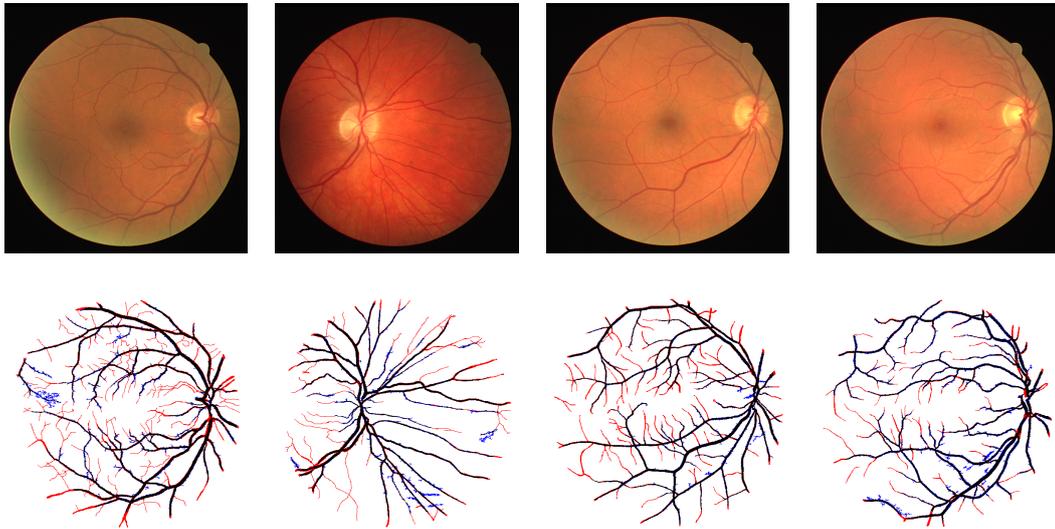
Figure 6.5: Illustration of four segmented blood vessels from the 20 test retinal images of DRIVE database. Top: Input color retinal images, Bottom: Corresponding segmented results, Black pixels: true positive; White pixels: true negative; Blue pixels: false positive; Red pixels: false negative.
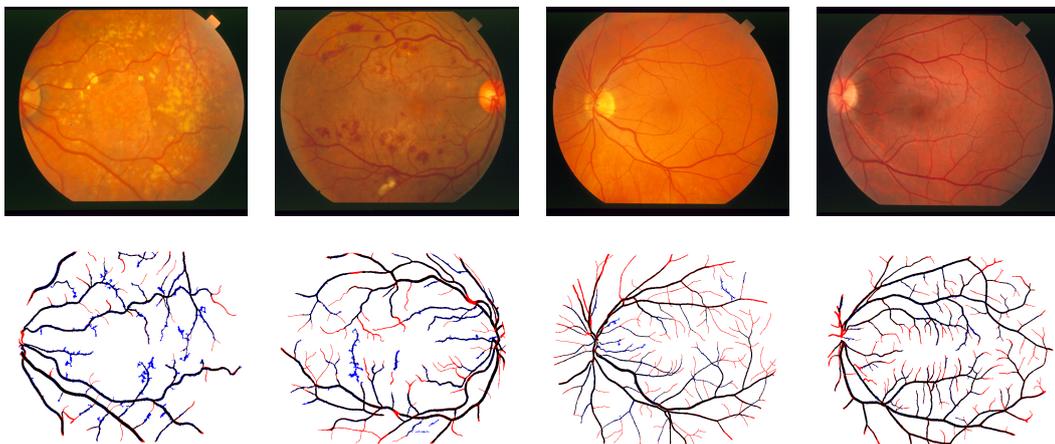
Figure 6.6: Illustration of four segmented blood vessels from the 20 test retinal images of STARE database. Top: Input color retinal images, Bottom: Corresponding segmented results, Black pixels: true positive; White pixels: true negative; Blue pixels: false positive; Red pixels: false negative.

These measurements are defined below as:

$$
\begin{aligned}
sensitivity &= TPR = \frac{TP}{P} = \frac{TP}{TP + FN} & (6.2)\\[2mm]
specificity &= TNR = \frac{TN}{N} = \frac{TN}{TN + FP} & (6.3)\\[2mm]
accuracy &= \frac{TP + TN}{TP + TN + FP + FN} & (6.4)\\[2mm]
kappa &= \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}. & (6.5)
\end{aligned}
$$

where TP stands for true positive, FP for false positive, TN for true negative and FN for false negative. $\Pr(a)$ is the relative observed agreement among raters, and $\Pr(e)$ is the hypothetical probability of chance agreement. If the raters are in complete agreement then $kappa = 1$. If there is no agreement among the raters other than what would be expected by chance (as defined by $\Pr(e)$), $kappa = 0$.

A benchmark of different approaches (including ours) is provided in Table 6.1 and Table 6.2. It shows the good performance of our proposed elongation-based upper leveling works for both DRIVE database and STARE database. In the case of DRIVE database, our result is slightly under the best results given by the method of Mendonça [Mendonca 2006]. Note also that the approaches of Staal [Staal 2004] and Niemeijer [Niemeijer 2004] are supervised approaches. On the STARE database, our proposed method performs also very well. It is better than the method of Mendonça [Mendonca 2006]. Both methods are very close to the second human observer. Table 6.3 shows that our method is more robust than others, in the sense that it performs equivalently on both abnormal and normal images. And to facilitate the comparison of our results with those presented by Hoover et al. [Hoover 2000], Jiang *et al.* [Jiang 2003] and mendonça [Mendonca 2006] in their original papers, we compute the average sensitivity for the false positive fraction (1-specificity) of the second hand-labeled image set, and by considering all image pixels (without FOV). The Table 6.4 shows the average sensitivity corresponds to a false positive fraction of 0.044, which was the value reported by Jiant et al [Jiang 2003].

Note that our proposed elongation-based upper leveling is only a "simple" filtering step, whereas other approaches are more complicated. Besides, our process is not complete: further post-processing can improve the results. Observe that the main missing part of retinal blood vessels is the ones which are very thin, a path opening [] applied on the input image and following a skeleton extraction may help to retrieve those missing vessels being very thin, so improve the quantitative evaluation.

Method	TPR	TNR	Accuracy	Kappa
2nd Expert	0.7761	0.9725	0.9473 (0.0048)	0.7589
Mendonça [Mendonca 2006]	**0.7344**	**0.9764**	**0.9452 (0.0062)**	-
Staal [Staal 2004]	0.7193	0.9773	0.9441 (0.0057)	0.7345
Niemeijer [Niemeijer 2004]	0.6793	0.9801	0.9416 (0.0065)	0.7145
Our	0.6924	0.9779	0.9413 (0.0078)	0.7166
Zana [Zana 2001]	0.6696	0.9769	0.9377 (0.0078)	0.6971
Al-Diri [Al-Diri 2009] -	-	-	0.9258 (0.0126)	0.6716
Jiang [Jiang 2003]	0.6478	0.9625	0.9222 (0.0070)	0.6399
Perez [Marinez-Pérez 1999]	0.7086	0.9496	0.9181 (0.0240)	0.6389

Table 6.1: Benchmark of different blood vessels segmentation approaches on DRIVE database.

Method	TPR	TNR	Accuracy
2nd Expert	0.8949	0.9390	0.9354 (0.0171)
Jiang [Jiang 2003]	-	-	**0.9513**
Our	0.7149	0.9749	0.9471 (0.0114)
Mendonça [Mendonca 2006]	0.6996	0.9730	0.9440 (0.0142)
Hoover [Hoover 2000]	0.6751	0.9567	0.9267 (0.0099)

Table 6.2: Benchmark of different blood vessels segmentation approaches on STARE database.

Method	Sensitivity	Specificity	Accuracy
Normal cases			
2nd human observer	0.9646	0.9236	0.9283 (0.0100)
Shape upper-leveling	**0.7178**	**0.9802**	**0.9493 (0.117)**
Mendonça [Mendonca 2006]	0.7258	0.9791	0.9492 (0.0122)
Hoover [Hoover 2000]	0.6766	0.9662	0.9324 (0.0072)
Abnormal cases			
2nd human observer	0.0.8252	0.9544	0.9425
Shape upper-leveling	**0.7120**	**0.9696**	**0.9447 (0.0106)**
Mendonça [Mendonca 2006]	0.6733	0.9669	0.9388 (0.0150)
Hoover [Hoover 2000]	0.6736	0.9472	0.9211 (0.0091)

Table 6.3: Benchmark of vessel segmentation methods (STARE images - Normal versus abnormal cases).

Method	Sensitivity
2nd human observer	0.895
Shape upper-leveling	0.830
Mendonça [Mendonca 2006]	0.828
Hoover [Hoover 2000]	0.75
Jiang [Jiang 2003]	**0.835**

Table 6.4: Benchmark of vessel segmentation methods (STARE images; FPF = 4.4%; no FOV.)

6.4.2 Optic nerve head (ONH) segmentation

We tested the method for the ONH segmentation described in Section 6.3.2 on the database of Digital Retinal Images for Optic Nerve Segmentation (DRIONS) [dri 2013a], [Carmona 2008]. The database consists of 110 color eye fundus images of 600×400, belonging to 55 patients with glaucoma (23.1%) and eye hypertension (76.9%) and randomly selected from an eye fundus image base belonging to the Ophthalmology Service at Miguel Servet Hospital, Saragossa (Spain). 50 images in the database have different type of defects. And for each image, two manual segmented ONH contours traced by two experts are also available. More details about the DRIONS database can be found in [dri 2013a].

Figure 6.7 shows some segmentation results of the DRIONS database. Qualitatively, except the last image (Figure 6.7 (l)), the ONH are correctly segmented for the other cases, although the approximate elliptical shapes do not fit very precisely the manual segmentations at the level of contours.

Quantitative assessment is based on the performance measurement named discrepancy δ [Lowell 2004], [Carmona 2008]. It measures the mean discrepancy between the points of contour obtained with the segmentation approach and a gold standard. For each image j, the gold standard is defined as the average of the two manual segmented contours. The discrepancy δ^j for image j is defined by

$$\delta^j = \frac{\sum_{i=1}^{N} \left((|m_i^j - \mu_i^j|)/(\sigma_i^j + \varepsilon) \right)}{N} \qquad (6.6)$$

Where N is the number of angularly equispaced radial segments, m_i^j is the length of the radius defining the i-th point of the ellipse proposed for the image j, μ_i^j and σ_i^j are the mean and typical deviation, respectively, of the lengths of the radii defining the i-th point of the two manual segmented contours for the image j, and $\varepsilon = 0.5$ is a small factor to prevent division by zero where the two experts are in exact agreement.

Based on the quantitative measurement δ, the segmentation results are divided into four categories: *excellent, good, fair* and *poor*, containing images with discrepancy value up to one, two, five, or more. Figure 6.7 illustrates some examples of ONH contours obtained with our method with different quality on the basis of discrepancy δ. Note that qualitatively, the segmentation result of Figure 6.7 (g) ($\delta_g = 3.02$) is as good as Figure 6.7 (b) ($\delta_b = 0.92$) and Figure 6.7 (c) ($\delta_c = 1.15$) or even better. The reason is that the discrepancy value defined in 6.6 is also a function of inter-observer variability. The greater the experts' variability when tracing the ONH contour, the smoother the discrepancy value obtained. The two experts get better agreement for the image Figure 6.7 (g) than the images Figure 6.7 (b) and

(a) $\delta = 0.88$ (b) $\delta = 0.92$ (c) $\delta = 1.15$

(d) $\delta = 1.60$ (e) $\delta = 2.22$ (f) $\delta = 2.96$

(g) $\delta = 3.02$ (h) $\delta = 3.40$ (i) $\delta = 4.27$

(j) $\delta = 4.63$ (k) $\delta = 5.04$ (l) $\delta = 7.21$
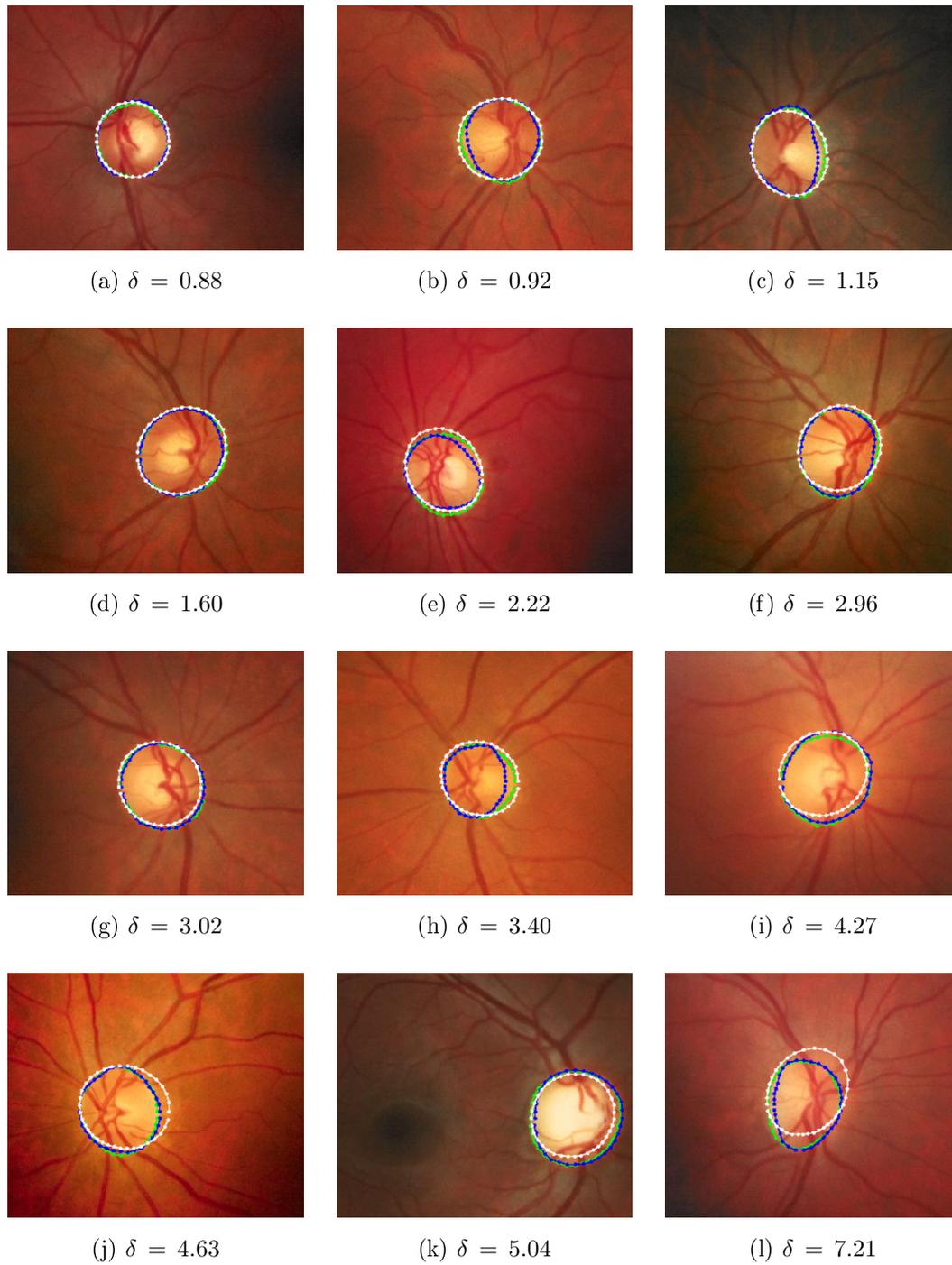
Figure 6.7: Some examples of ONH segmentation obtained with the approach described in Section 6.3.2: our segmented ONH in white; and two manual segmentations in green and blue.

Carmona [Carmona 2008]	96%
Our method	95.4%
Molina [Molina 2011]	95%
Lowell [Lowell 2004]	80%

Table 6.5: Percentage of images whose discrepancy δ is under 5.

Figure 6.7 (c).

A benchmark of different approaches is shown in Table 6.5. It is based on the percentage of images whose segmented ONH is at least fair ($\delta \leq 5$). We compare our proposed method with 3 other approaches which are considered as the best approaches tested on the DRIONS database. Our result is very close to the best one given by the approach of Carmona *et al.* [Carmona 2008], and is slightly better than Monlina's [Molina 2011]. it performs much better than Lowell's method [Lowell 2004].

For visualisation purposes, the ogive of discrepancy is plotted, namely, the number of images with discrepancy less than a given discrepancy value D. Thus the number (percentage) of images that fit any given level of accuracy can be read off the y-axis. The discrepancy curve is given in Figure 6.8. The comparison of the discrepancy curve obtained with Lowell's method [Lowell 2004] and our method reveals two different areas with opposite behavior, whose boundary is marked by the value $\delta = 2.7$. Belows this value, Lowell's results are slightly better than our proposal, and above this value, our approach performs better. We think the explanation is that our method approaches the solution using a non deformable ellipse, the local deformation phase in Lowell's method makes it possible to do a slight deformation of the ellipse, which make the results approach better the manual segmentations. However, the discrepancy curve of the second subinterval (2.7, 5) reveals the robustness of our method. Compared with the discrepancy curve obtained with Carmona's method [Carmona 2008] and Molina's approach [Molina 2011], our method performs slightly worse for the subinterval where discrepancy $\delta \leq 4$. We think the explanation is that the other 3 methods is based on the detection of interesting points around the ONH contour, and our method is based on the region. So they are more accurate for the low discrepancy. However, If we only consider the number of images whose detection results are fair, as shown in Table 6.5, our results are very close to the best one.

Note that our proposed method is much more simple than the 3 other approaches. It is a best node selection problem based on a pre-processing using the elongation-
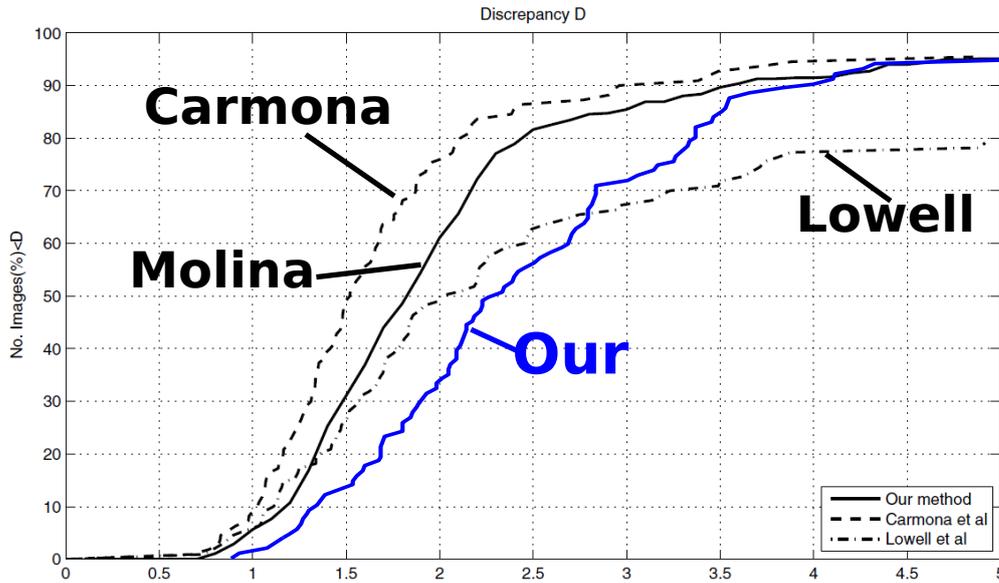
Figure 6.8: Accumulated discrepancy results for our detection method versus Carmona et al, Molina et al. and Lowell et al.

based upper leveling. And again, our process is not "complete" in the sense that further post-processing can improve the results (for example, local deformation of the best fitting ellipse).

6.4.3 Illustration of shaping

As stated in Section 2.3, when we want to process both upper and lower level sets, we use as $\mathcal{T}$ the topographic map, and the operator such created is called a morphological shaping $\mathcal{S}$. In both Figure 3.1 and Figure 6.9, we use for attribute $\mathcal{A}$ the circularity. The result of the shaping on Figure 3.1 (a) is shown in Figure 3.1 (g), and looks indeed better than the one of Figure 3.1 (f). In Figure 6.9, we compare our extinction-based self dual shaping approach with a variant of the state-of-the-art thresholding approach [Urbach 2007]. When the threshold of $\mathcal{A}$ is low, some objects do not appear (Figure 6.9 (c)). To be able to get all expected objects, we have to set a high threshold; however, in this case, too many unwanted objects are present (Figure 6.9 (d)). With our shaping , all the expected objects can be found, as depicted in Figure 6.9 (b). The results can be improved by combining some shape attributes. In Figure 6.9 (e) and Figure 6.9 (f), we use a combination of circularity and the I/A^2 [Urbach 2007], the moment of inertia divided by the square of area. The combination of shape attributes improves significantly the results. Still, our

shaping in Figure 6.9 (f) performs much better than the threshold-based strategy in Figure 6.9 (e).

6.5 Conclusion

Connected operators are filtering tools that have very good contour preservation properties and are capable of both low level filtering and high level object recognition [Salembier 2009]. The shape-based morphology being an extension of connected operators is a versatile framework for efficient shape processing, object segmentation, and hierarchical image simplification/segmentation. In this chapter, we presented the aspect of shape filtering, We tested one of the many shape-based lower/upper levelings on retinal image analysis. The first tested application is the blood vessel segmentation in retinal images. We used an elongation attribute characterizing long and thin structures. The quantitative assessments have shown that such a filter achieves results almost as good as the second human observer. Nevertheless, the proposed elongation-based upper leveling is only a "simple" filtering step, we have only two parameters which are relatively easy to set up. Other approaches dedicated for this application are more complicated. The second application is another important task in retinal image analysis, the optic nerve head (ONH) segmentation, which took advantage of the shape-based lower/upper levelings as image pre-processing. Then the ONH segmentation problem was solved by the best node selection using a specific designed attribute, the best fitting ellipse of the selected connected component approximates well the ONH. We have also shown that it achieves state-of-the-art results based on a numerical benchmark. Besides, in both applications, the processes are not "complete": Further post-processings can improve the results. We illustrated also an interesting example of the morphological shapings, where the proposed shaping filtered all the unwanted objects while keeping perfectly the desired bright and dark objects.
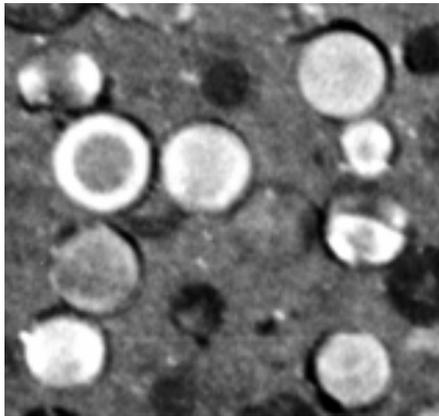
(a) Input image
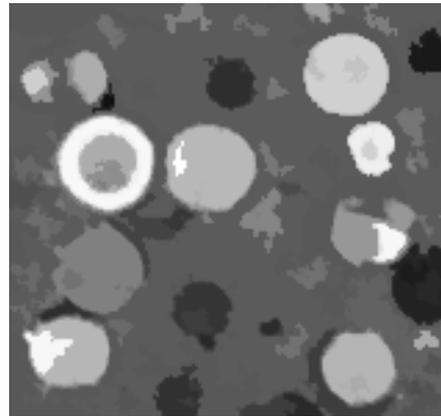
(b) Shaping 1
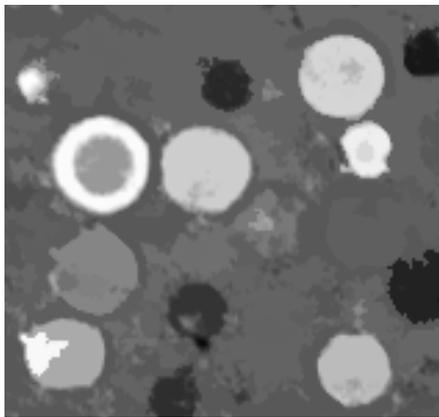
(c) Low threshold of $\mathcal{A}$
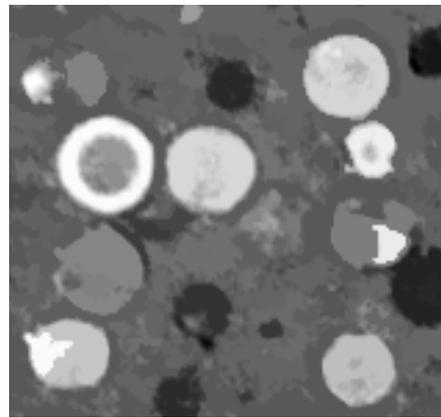
(d) Higher threshold of $\mathcal{A}$
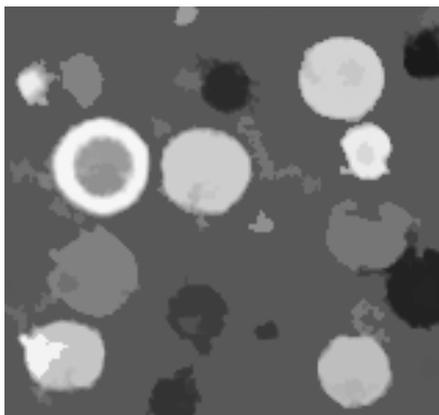
(e) Threshold strategy
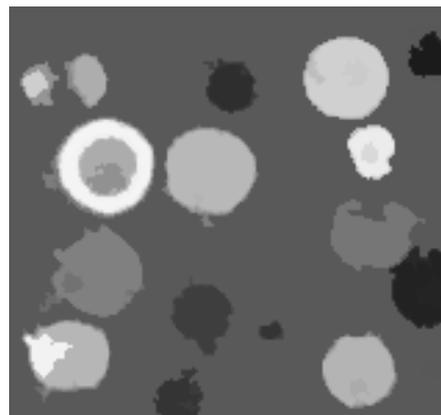
(f) Shaping 2

Figure 6.9: Comparison of extinction-based shapings with attribute thresholding. (b-d): Using one shape attribute; (e-f): Using a combination of shape attributes.

Object segmentation on the shape spaces

The tree-based shape spaces provide a tremendously reduced searching space for object segmentation. The shape space can be created by many means of tree-based image representations described in Section 2.3 to make that most part of meaningful objects are presented in the tree-based shape space. As described in Section 3.5, once the shape space is created, we can use the framework of shape-based morphology to extract those meaningful objects. To achieve this, we need an attribute function $\mathcal{A}$ characterizing the meaningfulness of each node (i.e., object). On the other hand, Image segmentation can be defined as the detection of closed contours surrounding objects of interest. Given a family of closed curves obtained by some means, a difficulty is to extract the relevant ones. A classical approach is to define an energy minimization framework, where interesting contours correspond to local minima of this energy. Active contours, graph cuts or minimum ratio cuts are instances of such approaches. In this chapter, we propose a novel efficient ratio-cut estimator which is both context-based and can be interpreted as an active contour. This ratio-cut estimator is used as the attribute function $\mathcal{A}$ in the framework of shape-based morphology. In this chapter, as a first example, we consider the shape space given by the topographic map representation. The estimator can be computed incrementally in an efficient fashion relying on this shape space. Experimental results on synthetic and real images demonstrate the robustness and usefulness of the proposed context-based energy estimator, and that the shape-based morphology is a versatile framework well suited for object segmentation tasks.

7.1 Introduction

In natural images, significant contours are usually smooth and have a good contrast. Following the seminal work of Mumford and Shah [Mumford 1989], finding contours is often tackled thanks to an energy-based approach, as a compromise between some internal force (regularity) and some image-driven force (image contrast along the contours, data attachment, etc.) Among the most notable methods in this class, the

snakes approach [Kass 1988] finds a local optimum of the energy, while Chan-Vese approach [Chan 2001] finds a global optimum.

It can be observed in practice that local optima are usually too local, in the sense that only information along the curve are taken into account. Conversely, a global optimum takes the whole image into account and some details can be found difficult to be included in the final result. Let us also mention that the search space for the optimal curve is quite large and depends on many parameters.

The main contribution of this chapter is the design of a ratio-cut energy evaluator that takes into account some context along the curve, by looking at a couple of regions around it. This estimator aims at assessing the possibility that the curve under scrutiny is the contour of some object. It thus can be used to choose, in a set of closed curves, for the ones that best represent the objects of the scene. By applying the aspect of object detection/segmentation in the framework of shape-based morphology, it is easy to find the objects: they are the minima of the estimator, when compared to their children and parent. Besides, the connected filtering in the shape space which is the basic idea of the framework of shape-based morphology helps to filter those meaningless local minima, so does the meaningless objects.

Recently, many authors (see for example [Caselles 1999]) claim that meaningful contours coincide with segments of the level lines of the image. The inclusion relationship of these level lines yields the topographic map [Monasse 2000b], a complete representation of the image that is invariant to changes of contrast. In this chapter, we illustrate the soundness of the proposed estimator as well as the aspect of object detection/segmentation of the framework of shape-based morphology relying on the shape space given by the topographic map representation.

There exists several works that use the topographic map for image simplification and segmentation. In [Ballester 2007], the authors propose to remove any level lines that, when removed, decrease the Mumford-Shah energy. In [Cardelino 2006], the authors remove the level lines that enclose a region similar to its parent w.r.t. an histogram-based distance. They then select interesting regions by identifying parts of the tree having an *homogeneous* histogram. Both methods require an information update after each removal of a non-relevant level line; such an update mechanism is a major bottleneck of those methods. In [Pardo 2002], the author proposes a segmentation algorithm that selects the perceptible level lines matching some criteria: number of T-junctions, compactness, and contrast.

Papers [Desolneux 2001] and [Cao 2005] are the closest ones to what we propose here. The authors define the *meaningfulness* of a given level line using the *a contrario* model. Only the smallest gradient along a level line and its length are used to estimate the *meaningfulness* of this line, based on the computation of a number

of false alarms (we will call this method "NFA" in the following). This makes their estimator sensitive to noise. In [Cao 2005], they improve the method by introducing a multi-scale approach, less sensitive to noise. In this chapter, our estimator is scale-invariant and designed from snake-like principles. We use the average of curvature along the curve as the internal force. A key contribution is the image force we propose, which integrates some contextual information. Eventually, we show that our estimator is robust both to noise and blur. Another advantage of our method is its efficiency for it has a quasi-linear time complexity.

The rest of this chapter is organized as follows. In Section 7.2, we will give a short review about the snake/active contour models [Kass 1988] based on which our context-based energy estimator (see Section 7.3) is proposed. Then in Section 7.4, we will detail how to make use of the framework of shape-based morphology taking the context-based energy estimator as the attribute function $\mathcal{A}$, the underlying shape space is given by the topographic map. In Section 7.5, we will show some experimental results of object segmentation on synthetic and real images, and a hierarchical version of the object segmentation will also be illustrated. Finally we conclude in Section 7.6.

7.2 Related work

Some related works about the object segmentation relying on the use of shape space are already shortly reviewed in Section 2.6 and Section 7.1. So in this section, we focus on the snake/active contour models based on which the proposed context-based energy estimator is formed.

7.2.1 Snake/active contour models

Snake or active contour model is an energy minimizing, deformable spline influenced by constraint and image forces that pull it towards object boundaries. It has been widely used for object segmentation. Snake is an "active" model as it always minimize its energy functional and therefore exhibits dynamic behavior. Since its original introduction by Kass et al. [Kass 1988], many active contour models have been proposed, such as the geodesic active contours proposed by Caselles et al. [Caselles 1997], the active contours with edges proposed by Chan and Vese [Chan 2001]. We review only the original active contour model and the active contours without edges.

1) Classical active contour

The idea of snakes or active contours [Kass 1988] is to evolve a curve under the influence of internal forces coming from the curve itself and of external forces computed from the input image or added by the user to impose some constraints. Those forces are mapped into the respective terms of an energy

$$E^{local} = \alpha E^{snk}_{int} + E^{snk}_{ext} + \beta E_{con} \qquad (7.1)$$

whose minimization drives the curve evolution. α and β are two positive parameters.

Let $C(s) : [0,1] \to \mathbb{R}^2$ be a parameterized curve. Then the internal force E^{snk}_{int} can be written

$$E^{snk}_{int} = \alpha_1 \int_0^1 |C'(s)|^2 ds + \alpha_2 \int_0^1 |C''(s)| ds. \qquad (7.2)$$

The internal force controls the smoothness of the contour. In the classical snakes [Kass 1988], the external force relies usually on an edge-detector depending on the gradient of the image f, that stops the evolving curve on the boundary of desired object. The external force is usually given by

$$E^{snk}_{ext} = -\int_0^1 |\nabla f(C(s))|^2 ds. \qquad (7.3)$$

The constraint force can be as simple as the length of the curve denoted by Length$(C(s))$. Consequently, the energy to be minimized in classical active model is given by

$$E^{local} = \alpha_1' \int_0^1 |C'(s)|^2 ds + \alpha_2' \int_0^1 |C''(s)| ds - \int_0^1 |\nabla f(C(s))|^2 ds + \beta \text{Length}(C(s)). \qquad (7.4)$$

Observe that, by minimizing the energy in Eq (7.4), we are trying to locate the curve at the points of maxima $|\nabla f|$, performing as an edge-detector, while maintaining a smoothness in the curve and satisfying the constraints imposed by the user.

In problem of curve evolution, the level set method and in particular the motion by mean curvature of Osher and Sethian [Osher 1988] is usually used to solve the energy minimization problem of Eq (7.4).

2) Active contour without edges

The classic active contour models reviewed as above segment objects whose boundaries are defined by gradient. In [Chan 2001], Chan and Vese introduced an active contour model without edges that segments objects whose boundaries are not necessary defined by gradient. It relies on techniques of curve evolution, Mumford-Shah functional and level sets.

For a given evolving curve C in the image domain Ω, which is the boundary of an open subset $U \subset \Omega$ (i.e., $C = \partial U$), let $inside(C)$ denotes the region U, and $outside(C)$ denotes the region $\Omega \backslash \bar{U}$. The energy functional to be minimized in the model of active contour without edges is given by

$$
\begin{aligned}
F(c1, c2, C) \ &= \mu \text{Area}(inside(C)) + \nu \text{Length}(C) \\
&+ \lambda_1 \int_{inside(C)} |f(x, y) - c_1|^2 dxdy \\
&+ \lambda_2 \int_{outside(C)} |f(x, y) - c_2|^2 dxdy,
\end{aligned}
\tag{7.5}
$$

where $\text{Area}(inside(C))$ denotes the area of the region inside the curve C (i.e., the subset U), c_1 and c_2 are two constants depending on the evolving curve C being the average of f inside C and respectively outside of C given by Eq (5.3), and $\mu \geq 0, \nu \geq 0, \lambda_1 > 0, \lambda_2 > 0$ are fixed parameters. Usually μ is set to 0, λ_1 and λ_2 is set to $\lambda_1 = \lambda_2 = 1$. In this case the energy functional $F(c1, c2, C)$ in Eq (7.5) becomes

$$
F(c1, c2, C) \ = \nu \text{Length}(C) + \int_{inside(C)} |f(x, y) - c_1|^2 dxdy + \int_{outside(C)} |f(x, y) - c_2|^2 dxdy.
\tag{7.6}
$$

The energy functional $F(c1, c2, C)$ in Eq (7.6) is related with the piecewise-constant Mumford-Shah functional in Eq (5.4). In fact, this energy functional of active contour without edges can be formulated by the piecewise-constant Mumford-Shah functional taking two regions: the "region" inside the curve C and the "region" outside the curve C. Note however that the notion of "region" used here is not necessary a connected component due to the fact that the evolving curve C might be composed of several disjoint closed contours.

This particular case of the piecewise-constant Mumford-Shah functional minimization can also be formulated and solved using the level set method [Osher 1988]. Let us point out again that the level set methods have solid theoretical foundations, yet they are often computational expensive.

7.3   Context-based energy estimator

We will detail the proposed context-based energy estimator in this section. It is inspired from the active contour model. The classic active model drives the curve evolution by minimizing a local energy written in term of Eq (7.1), in which the external force is usually given by an edge-detector depending on the gradient. This energy functional is too local, and it segments objects whose boundaries are defined by gradient. Whereas, the active contour without edges drives the curve evolution

by minimizing a global energy functional defined in Eq (7.5) and Eq (7.6). The context-based energy estimator is a contextual energy $E^{context}$ which is based on the classic active contour model and the active contour without edges. This energy estimator $E^{context}$ can also be written as Eq (7.1) given as below:

$$E^{context} = \alpha E_{int} + E_{ext} + \beta E_{con}. \tag{7.7}$$

Different from the classic active contour model, the external force E_{ext} is inspired from the model of active contour without edges. We compute an indicator sharing the same idea of applying the active contour without edges on a contextual patch around a closed contour. It is in fact related to the popular Fisher's test [Phillips 1989]. So the proposed context-based energy estimator is neither too local nor too global, which forms its main contribution. Let us detail each term in Eq (7.7) for a discretized closed contour $\partial\tau$ of a region τ in the following sections. The discretized closed contour is composed of a set of pixel edges e (i.e., the 1-faces) and 0-faces as shown in Figure 3.12.

7.3.1 Internal energy E_{int}

The internal energy controls the smoothness of the contour which is usually measured by the curvature along the contour. For a discretized curve $\partial\tau$, the internal force can be given by

$$E_{int} = \sum_{e \in \partial\tau} |curv(f)(e)|, \tag{7.8}$$

Where $curv(f)$ is the scalar curvature of image f defined on each point $x_0 \in \Omega$ [Ciomaga 2010], it is given by

$$curv(f)(x_0) = \frac{f_{xx}f_y^2 - 2f_{xy}f_xf_y + f_{yy}f_x^2}{(f_x^2 + f_y^2)^{3/2}}(x_0). \tag{7.9}$$

In our case, the region contour is materialized by the pixel edge lying between points (i.e., 1-faces) and 0-faces. To compute the measurement based on region contour, we only consider the pixel edges, i.e., 1-faces. We adapt the computation of those differentials in Eq (7.9) to the pixel edges lying between points. For a horizontal pixel edge, e.g., the blue one in Figure 7.1), those differentials are given by

$$f_x = f(x_{2,1}) - f(x_{1,1}) \tag{7.10}$$

$$f_{xx} = 1/2 \times (f(x_{3,1}) - f(x_{2,1}) - f(x_{1,1}) + f(x_{0,1})) \tag{7.11}$$

$$f_y = 1/4 \times (f(x_{2,2}) + f(x_{1,2}) - f(x_{2,0}) - f(x_{1,0})) \tag{7.12}$$

$$f_{yy} = 1/2 \times (f(x_{1,2}) + f(x_{2,2}) + f(x_{1,0}) + f(x_{2,0})) - f(x_{1,1}) - f(x_{2,1}) \tag{7.13}$$

$$f_{xy} = 1/2 \times (f(x_{2,2}) + f(x_{1,0}) - f(x_{1,2}) - f(x_{2,0})). \tag{7.14}$$
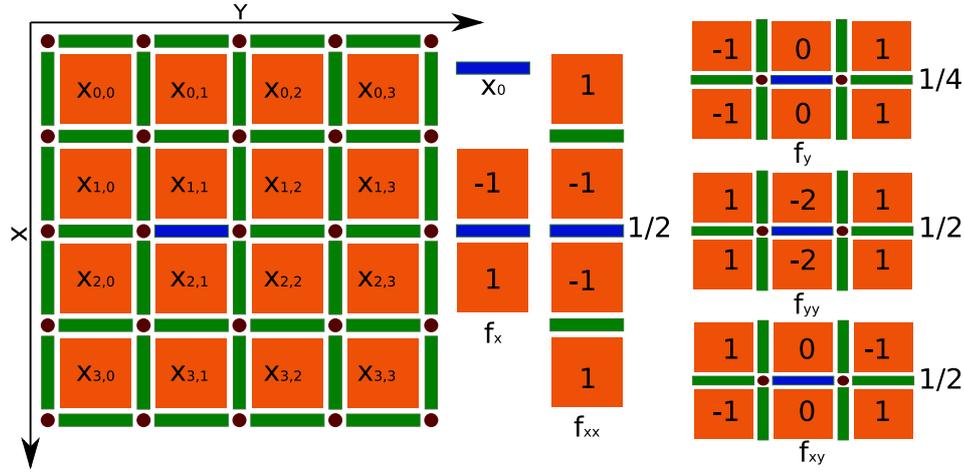
Figure 7.1: Differential operators for the horizontal blue pixel edge. The mask for those differentials in Eq (7.9) is depicted on the right side.

For the vertical pixel edges, the differentials can be computed symmetrically.

Finally, we also propose to modify the internal energy from the snakes approach: it is normalized to make it invariant to scale. The average of curvature along the region contour $\mathcal{A}_\kappa$ defined in Eq (5.28) is used instead.

7.3.2 External energy E_{ext}

The curvature $\mathcal{A}_\kappa$ and the gradient $|\nabla|$ is *local* to the curve. The active contour without edges proposed in [Chan 2001] minimizes a *global* energy which is similar with the piecewise-constant Mumford-Shah functional based on the segmentation error defined for a Given a region $\mathcal{R}$ by

$$V(f, \mathcal{R}) \ = \ \sum_{p \in \mathcal{R}} (f(p) - \overline{f}(\mathcal{R}))^2, \tag{7.15}$$

where $\overline{f}(\mathcal{R})$ is the mean value of u over $\mathcal{R}$.

We propose to replace the external energy in the classical active contour model by a *context* one, inspired by the segmentation error (see Eq (7.15)) term in energy functional for active contour without edges. For that, given a curve $\partial\tau$, we define the regions $\mathcal{R}_{in}^{\varepsilon}(\partial\tau)$ and $\mathcal{R}_{out}^{\varepsilon}(\partial\tau)$ as the sets of points of maximal distance ε from $\partial\tau$, respectively inside and outside of the curve. Those two regions are illustrated by Figure 7.2 (b). The proposed external energy E_{ext} is:

$$E_{ext}(f, \partial\tau) \ = \ \frac{V\big(f, \mathcal{R}_{in}^{\varepsilon}(\partial\tau)\big) + V\big(f, \mathcal{R}_{out}^{\varepsilon}(\partial\tau)\big)}{V\big(f, \mathcal{R}_{in}^{\varepsilon}(\partial\tau) \cup \mathcal{R}_{out}^{\varepsilon}(\partial\tau)\big)}. \tag{7.16}$$

This energy is low when the segmentation error is much lower for two classes than for one class, meaning that the curve is an object contour. Thanks to the numerator, it is also inversely proportional to the object contrast. In fact this external energy is related to the popular Fisher's test [Phillips 1989] based on the *Fisher distance* given by

$$\frac{(n_1 + n_2)(\mu_1 - \mu_2)^2}{n_1 \sigma_1^2 + n_2 \sigma_2^2} = \frac{n\sigma^2}{n_1 \sigma_1^2 + n_2 \sigma_2^2} - 1, \tag{7.17}$$

where n_1, n_2 is the number of pixels inside the interior region $\mathcal{R}_{in}^{\varepsilon}(\partial\tau)$ and respectively the exterior region $\mathcal{R}_{out}^{\varepsilon}(\partial\tau)$, σ_1^2, σ_2^2 denote the variance in those two regions, and $n = n_1 + n_2$ is the number of pixels of the mixture region, σ^2 is the variance inside the mixture region. In fact, the external energy E_{ext} defined in Eq (7.16) can be written as

$$E_{ext}(f, \partial\tau) = \frac{n_1 \sigma_1^2 + n_2 \sigma_2^2}{n\sigma^2}, \tag{7.18}$$

which corresponds to the inverse of the first term in Fisher's test given by Eq (7.17). Let us also remark that this contextual energy introduces some spatial information that is not naturally present in any of the shape space given by a tree-structure representation for a set of closed curves in an image.

7.3.3   Snake-like energy estimator

For the purpose of meaningful object segmentation, we add the constraint energy that aims at penalizing too small objects:

$$E_{con}(f, \partial\tau) = 1/\text{Length}(\partial\tau). \tag{7.19}$$

Consequently, the proposed context-based energy estimator is finally given by

$$
\begin{aligned}
E^{context}(f, \partial\tau) = &\alpha \frac{\sum_{e \in \partial\tau} |curv(f)(e)|}{\text{Length}(\partial\tau)} \\
&+ \frac{V\big(f, \mathcal{R}_{in}^{\varepsilon}(\partial\tau)\big) + V\big(f, \mathcal{R}_{out}^{\varepsilon}(\partial\tau)\big)}{V\big(f, \mathcal{R}_{in}^{\varepsilon}(\partial\tau) \cup \mathcal{R}_{out}^{\varepsilon}(\partial\tau)\big)} \\
&+ \beta/\text{Length}(\partial\tau).
\end{aligned}
\tag{7.20}
$$

In our experiments we take $\alpha = 60$, $\beta = 2$, and $\varepsilon = 5$.

7.4   Application on the topographic map

The context-based energy estimator $E^{context}$ described in Section 7.3 provides a good manner measuring the possibility of the presence of a object. Moreover, it will be shown in Section 7.4.2 that this energy estimator can be computed efficiently upon the shape space. All these motivates us to use it as an attribute

function $\mathcal{A}_o$ in the framework of shape-based morphology, whose aspect of object detection/segmentation provides an efficient object segmentation scheme. As a first example, we use the shape space given by the topographic map representation in this chapter. The overview of the method will be described in Section 7.4.1. The connected filtering in the shape space to remove those meaningless objects will be detailed in Section 7.4.3. Finally, the hierarchical version of object segmentation will be discussed in Section 7.4.4.

7.4.1 Overview of the method

The method for object segmentation relying on the framework of shape-based morphology we propose is composed of four steps.

1. First the contents of the input image is transformed to a tree-based shape space. Many tree-based representations of an image do exist (*e.g.* min/max tree, hierarchies, ...). In this chapter, we use the topographic map [Monasse 2000b].

2. Second our context-based energy estimator is computed for each node of this tree as the attribute function $\mathcal{A}_o$.

3. Third connected filtering in the shape space weighted by the attribute function $\mathcal{A}$. Morphological closing removes meaningless minima, so does the meaningless objects. We identify the resistant significant local minima of the energy. Each node corresponding to such a minimum is a component whose contour is the one of an object.

4. Last based on the idea that resistant minima represent the meaningful objects. We weight each minimum (so the corresponding objects) by the filtering force at which this minimum vanishes, which leads to a saliency map representing a hierarchical object segmentation.

7.4.2 Making the method efficient

For all four steps of the method, we need to deal not only with image pixels but also with contours. We thus rely on a representation that handle both pixels *and* pixel edges. In other words, we materialize the elements what lie in-between pixels, as depicted in Figure 7.2.

Tree Computation. To compute the topographic map, we use an algorithm [Géraud 2013] very similar to the one described in [Berger 2007]. A first pass sorts the pixels by a propagation process starting from the image boundary.

<center>(a)   (b)</center>
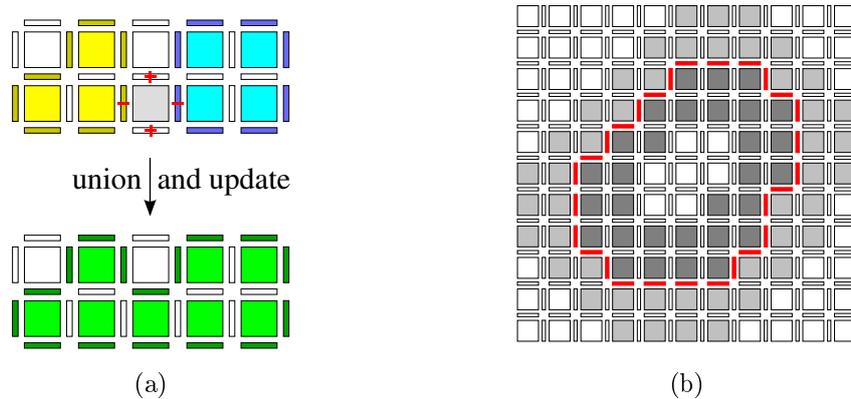
Figure 7.2: (a): Updating contour information when a union between two components (yellow and blue) occurs thanks to a pixel (gray). (b): Regional context of a level line (red) is the inner (dark gray) and outer (light gray) regions.

Then a second pass, in reverse order, build the tree while performing the union-find algorithm. This algorithm has a quasi-linear time complexity when pixel values have a low quantization (see Section 9.1 for more details).

Contours Information Computation. During the union-find pass, contour information can be efficiently updated. As depicted in Figure 7.2 (a), when performing a union of two components (resp. yellow and blue) due to adding a pixel (gray), it is easy to know how to update the contour. More algorithmic details can be found in Section 9.2.2.

Regions Information Computation. In order to calculate the regions information efficiently, we approximate the inner region and the outer region of each level line by only taking into account the pixels which are aligned perpendicularly to each edge of the level line. Note that some pixels may be counted several times. An example with $\varepsilon = 2$ is given in Figure 7.2 (b). The efficient algorithm dealing with the context information will be presented in Section 9.2.3.

Energy Computation. All energy terms can be computed incrementally during the union-find pass since they can be decomposed into linear (additive and subtractive) parts. The curvature, as a linear filter, is computed for every pixel edges in a preliminary step.

7.4.3 Energy filtering in shape spaces

As illustrated in Figure 3.9 and depicted in Section 3.5, although an improvement of object evidence is achieved by the context-based energy estimator compared with the classic active contour model, there are still too many local minima in the shape space weighted by the attribute function $\mathcal{A}_o$ given by

$$\mathcal{A}_o = E^{context}. \tag{7.21}$$

A connected filtering step in the shape space is required to remove those meaningless local minima. We thus construct a Min-tree representation of the tree-based shape space $S_{\mathcal{T}}$ to implement a connected filtering known as minima killer. The second attribute function $\mathcal{A}\mathcal{A}$ based on which perform the connected filtering in the shape space is can be the height of $\mathcal{A}_o$ in Eq (7.21) and the total variation described in Section 3.2.2. In the filtered attribute function $\mathcal{A}'_o$, some local minima may contain several nodes in the shape space, we select the node having the smallest $\mathcal{A}_o$ as the representative node $\mathcal{N}_r$, the connected component represented by $\mathcal{N}_r$ is segmented as the meaningful object for those local minima. Note that with this method, the meaningful objects presented in the same branch of a tree representation can be all extracted.

7.4.4 Hierarchical object segmentation

Following the idea of object segmentation which says that significant local minima of attribute $\mathcal{A}_o$ in the shape space correspond to meaningful objects, so for a given local minimal node $\mathcal{N}_m$, its importance compared with its neighbors in the shape space measures the significance of the objects represented by that node $\mathcal{N}_m$. By augmenting the filtering force in the shape space, more and more local minima vanish, in this sense, the resistance of being filtered by the connected filtering in the shape space based on a second attribute function $\mathcal{A}\mathcal{A}$ can be used to produce a soft object segmentation decision. This resistance is usually known as the extinction value defined on the local minima [Vachier 1995] (see Section 3.2.3 for more details). By weighting this extinction value of each local minimal node $\mathcal{N}_m$ to the region contour $\partial\mathcal{N}_m$, we obtain a saliency map that represent a soft object segmentation result corresponding to every possible filtering force applied in the shape space. Each threshold of this map produces an object segmentation result with certain filtering force. This hierarchical decision is depicted in Section 3.6.

7.5   Results

In this section, we will show some results about the aspect of object detection/segmentation in the framework of shape-based morphology. The attribute function $\mathcal{A}_o$ used is the context-based energy estimator described in Section 7.3. We will begin with in Section 7.5.1 a illustration of this attribute function $\mathcal{A}_o$ compared with the classical snake energy [Kass 1988] and the NFA [Cao 2005]. Then in Section 7.5.2, we will compare the proposed method with several related approaches on a synthetic image, which shows the usefulness of our method. Somme illustrations of the proposed on natural images will be shown in Section 7.5.3. Finally, we will shown some hierarchical object segmentation results in Section 7.5.4.

7.5.1   Energy estimator evaluation

Figure 7.3 illustrates an evolution of the proposed context-based energy estimator along a branch of the topographic map. In this Figure, one can see that the energy that we have defined (the blue curve) is a function that evolves rather smoothly upon the topographic map from a node to its parent (from left, a leaf, to right, the root node). Compared with the classical snake energy (the green curve), the context-based energy estimator defined by replacing the gradient energy E_{ext}^{snk} with the context-based indicator dramatically improves the evidence of an object presence. The NFA is also shown in this Figure (the orange curve), it does not evolve regularly enough, because the NFA approach [Cao 2005] is sensitive to noise since the method relies on the minimum value of contrast along boundaries. As a consequence, selecting relevant minima is hazardous and objects are not well located; see Figures 7.3 and 7.4 (b). That confirms the importance of taking into account some regional information.

7.5.2   Comparison on a synthetic image

The results obtained by our method on a synthetic image are shown in Figure 7.4 (f) and they are compared to the ones of several related approaches.

The result of NFA approach [Cao 2005] is shown in Figures 7.4 (b). The contours are not precisely located due to its sensitivity to noise. A quantitative error comparison between NFA and our method is given in Figure 7.5 to depict the robustness to noise and blur of both methods.

Figure 7.4 (c) is the result obtained by "active contour without edges" [Chan 2001], with many circles as initial contours; the bottom object is missed since it is too similar to the background. That problem comes from the
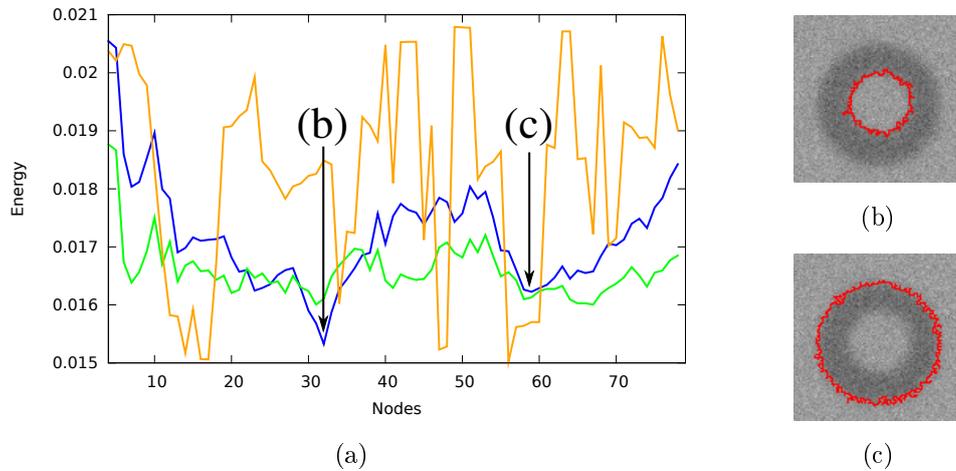
Figure 7.3: Energy estimator on a set of curves of an image, ordered by inclusion; classical snake energy is depicted in green, our proposed energy is in blue, and NFA is in orange. (b) and (c): the curves corresponding to significant energy minima.

use of a global energy: such an object is not contrasted enough w.r.t. the rest of the image.

The approach of Ballester et al. [Ballester 2007] relying on Mumford-Shah functional gives pretty good results; see Figure 7.4 (d). Unfortunately there is a false object, the triangle boundary, that does not disappear when increasing the regularization strength parameter λ; see Figure 7.4 (e). Instead of that, an actual object disappears.

7.5.3   Some illustrations on natural images

Figure 7.6 presents the results of our method on two natural images. On the left image, the three vehicles are successfully identified but the roof of one car is missed since it is very similar to the background. On the satellite image, the buildings and their shadows are well segmented.

We have implemented the proposed method [1] using our C++ image processing library [Levillain 2010], available on the Internet as free software. Processing a 512×512 pixels image takes less than 0.5 second on a regular PC station.

7.5.4   Hierarchical object segmentation

The saliency maps using the extinction value for the images in Figure 7.6 are illustrated in Figure 7.7. The saliency map represents a hierarchical object segmentation,

---

[1] Demo available on http://olena.lrde.epita.fr/ICIP2012

(a) Input image   (b) NFA   (c) Chan-Vese

(d) Ballester, $\lambda{=}2k$ (e) Ballester, $\lambda{=}3k$ (f) Our method
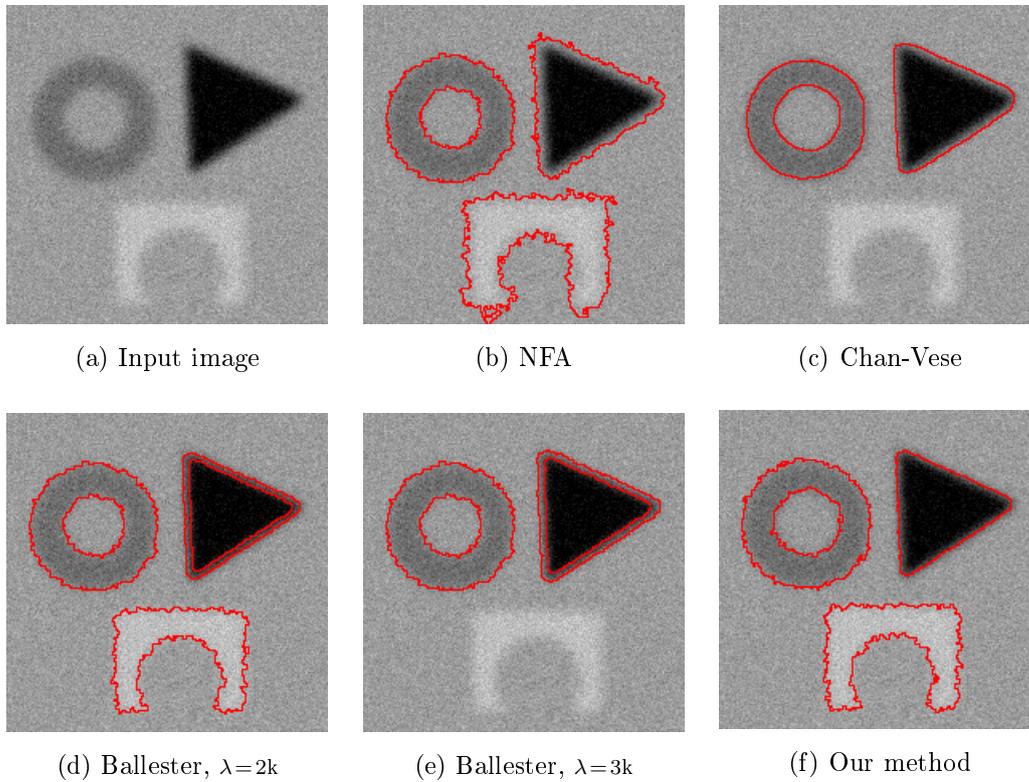
Figure 7.4: Comparison with three other methods.

(a) Error due to increasing noise   (b) and to increasing blur
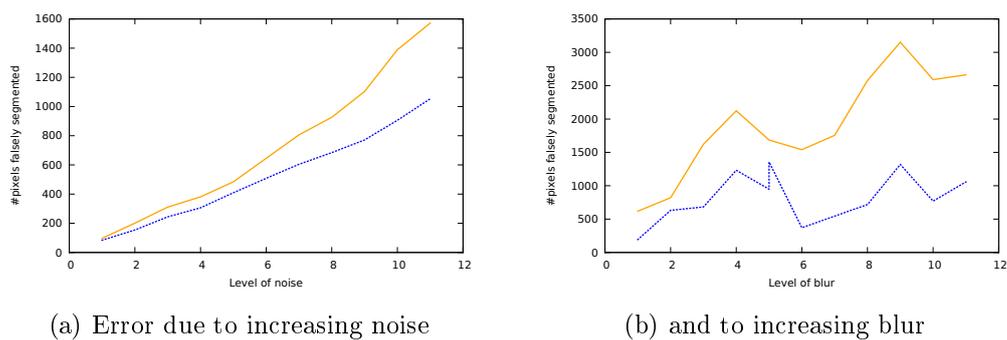
Figure 7.5: Evolution of number of falsely segmented pixels (ordinate) w.r.t. increasing noise and blur (abscissa) for the synthetic image in the Figure 7.4(a); NFA error evolution is depicted in orange and our method error in blue.
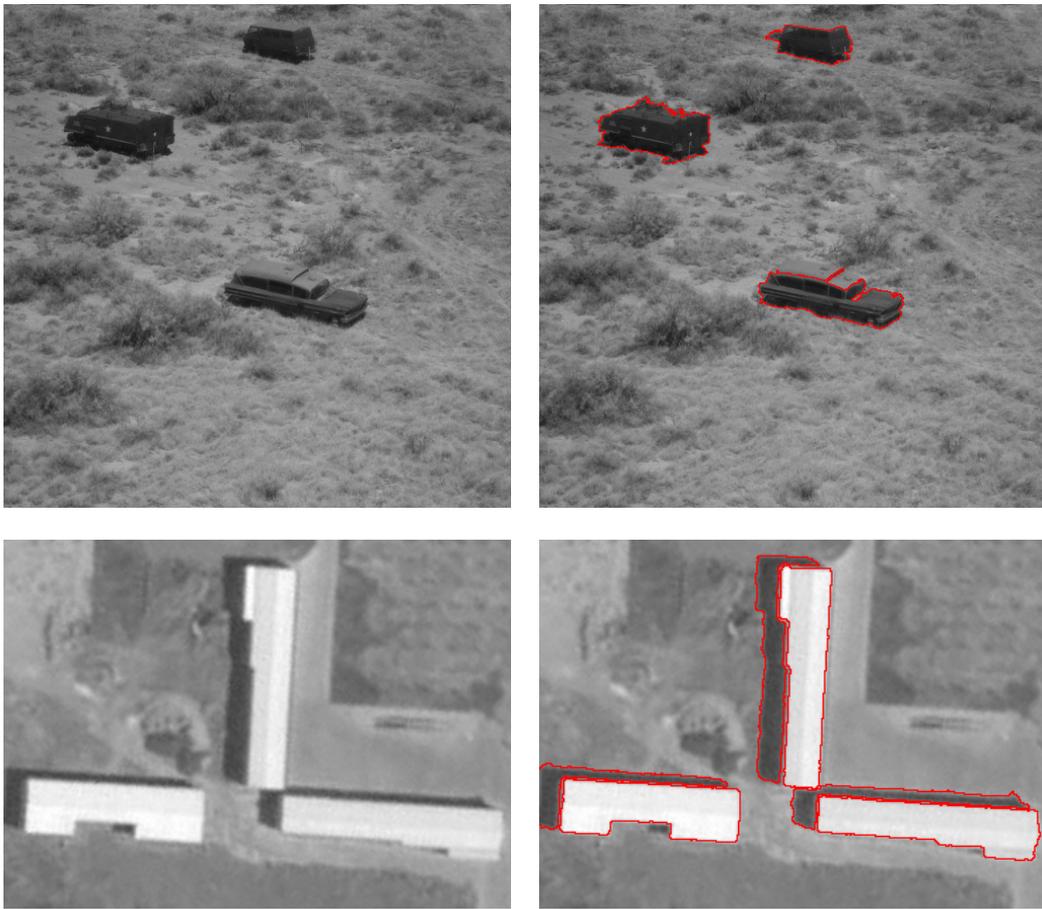
Figure 7.6: Results on natural images. Left column: input images; Right column: Segmented objects.
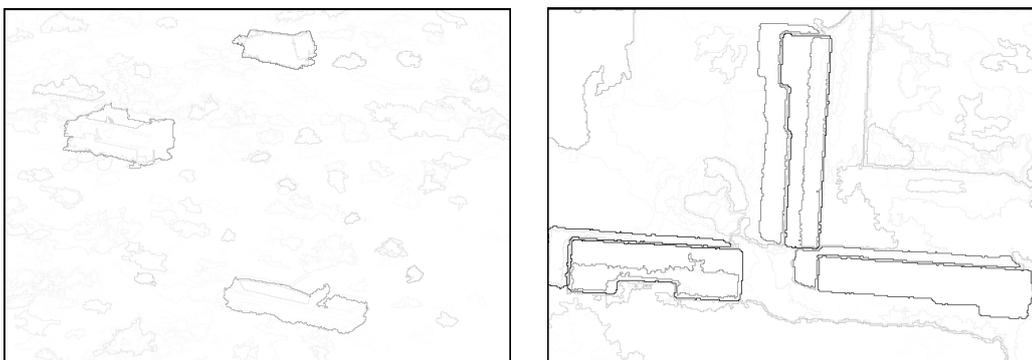
Figure 7.7: Inversed saliency map representing hierarchical object segmentation.

thresholding it with certain value can give a segmentation result shown in Figure 7.6.

7.6   Conclusion

This chapter presented an object detection/segmentation approach using the framework of shape-based morphology. A major contribution of this chapter is a new energy functional, well-suited to characterize object contours. We have shown that taking into account a regional context, so that the energy is neither too local nor global, improves both the relevance and robustness of object detection, as compared to three state-of-the-art approaches. We have also shown that the segmentation scheme relying on the framework of shape-based morphology has a quasi-linear time complexity. Moreover, it provides a soft object segmentation result via the saliency map. A major perspective of our work is to demonstrate its usefulness with other tree-based representations, to extend it to color and, more generally, multi-valued images.

Extending constrained connectivity

For all the above chapters in Part III about the applications, the underlying shape spaces are given by threshold decompositions based trees, i.e., the morphological trees including Min-tree, Max-tree, and the topographic map. Each cut of these trees gives a subset of the image domain Ω. In this chapter, we propose an application relying on the shape space given by the hierarchy of segmentations whose cut results in a partition of Ω. It has been show in many works that the hierarchies are a powerful tool for image segmentation, they provide a multi-scale representation which allows to design robust and fast algorithms. In this chapter, we focus on the aspect of hierarchy transformation of the framework of shape-based morphology. As a first example, we work on the shape space given by the α-tree, known also as the hierarchy of constrained connectivity, or quasi-flat zones, which has been shown efficient for hierarchical image partitioning and simplification. What we propose can be seen as an extension of the constrained connectivity by transforming the α-tree into another hierarchy of segmentations represented by a saliency map. Experiments on the dataset of BSDS500 show that this later saliency map might better represent the content of the input image, which demonstrates the usefulness of the hierarchy transformation aspect in the framework of shape-based morphology.

8.1 Introduction

Image segmentation is one of the oldest and most challenging problem in image processing. Given an image f, even for a human observer, it is hard to determine a unique meaningful segmentation of f, and it is even harder to establish consensus between different observers. Indeed, as shown in the Berkeley Segmentation Data Set and Benchmarks 500 (BSDS500) [Arbelaez 2011], for each image, the ground truths drawn by different observers are usually different. It is harder to say which single segmentation is better. The image segmentation is a usually low level task. As remarked by Guigues et al. in [Guigues 2006], a low level segmentation tool should remain scale uncommited, because the structures which can be useful to high level

task can have arbitrary size. A segmentation should output a multi-scale description of the image f. Higher level information, such as finding a unique partition of the image, could be achieved afterwards by introducing additional criterion or by manually inspecting the multi-scale representation. An usual approach to overcome the difficulty of finding a unique meaningful partition, and to satisfy the multi-scale property is to compute a hierarchy of segmentations instead of a single one.

There exist many methods for building a hierarchical segmentation. They can be divided into three classes: bottom-up, top-down or split-and-merge. The bottom-up approaches are usually based on merging algorithms which share three basic ingredients: a region model, a region criterion, and a merging order. Starting from a fine initial partition, the regions merge with the adjacent regions consecutively which forms a partition coarser and coarser, until the whole image domain is considered as a single region. The binary partition tree (BPT) reviewed in Section 2.3.3 and the α-tree reviewed in Section 2.3.4 are such instances. The quadtree reviewed in Section 2.3.5 is an example of top-down or split-and-merge approach. A recent review of the approaches of building hierarchies of segmentations can be found in [Marfil 2006] and [Soille 2008].

A useful representation of hierarchical segmentations was originally introduced in [Najman 1996] under the name of saliency map, and it is also popularized under the name of ultrametric contour map recently. This representation of hierarchies has been used by several authors, for example for visualization purposes [Guigues 2006], or for hierarchies comparison [Arbeláez 2006b, Arbelaez 2011]. Besides, its has been shown in [Najman 2011] that any hierarchical segmentation, i.e. any saliency map is equivalent to a ultrametric watershed, which provides an efficient algorithm based on the scheme of watershed algorithm to compute the whole hierarchy.

A hierarchy of segmentations encodes a set of segmentations from fine to coarse. Given an image f and two associated hierarchies H_1 and H_2, it is not trivial to judge which hierarchy is better. In fact, this question leads to a more basic and important question: how to cut the hierarchy to obtain meaningful partitions, bearing in mind that a hierarchy H generates a subset of all the possible partitions of image f. Let us denote $\mathcal{P}art(H)$ as all the possible partitions by cutting the hierarchy H, then we have

$$\mathcal{P}art(H) \subset \mathcal{P}art(\Omega). \tag{8.1}$$

The most simple approach consists in cutting the hierarchy based on the height of nodes in H, in other word, the hierarchy H is seen as a hierarchy indexed with the height that can be represented by saliency map M based on the height. There exist also several works that cut the hierarchy H "optimally" subordinated to some conditions, such as optimal energy minimization (the work of Guigues et

al. [Guigues 2006] and Serra and Kiran [Serra 2013, Kiran 2014] are such instances), or a simple condition as the global range ω used in the hierarchy of constrained connectivity [Soille 2008]. These "optimal" partitions corresponding to partitions obtained by thresholding an associated saliency map $\mathcal{M}$ denoted as $\mathcal{P}art(\mathcal{M})$. Then we have

$$\mathcal{P}art(\mathcal{M}) \subset \mathcal{P}art(H) \subset \mathcal{P}art(\Omega), \qquad (8.2)$$

which means that those existing methods can be seen as a hierarchy index modification method. The underlying hierarchy structure remains intact.

With these "optimal" partitions extracted from a hierarchy H, we are now able to decide which hierarchy is better if the ground truth(s) is (are) available. We compute the distance of each partition $\mathcal{P} \in \mathcal{P}art(\mathcal{M})$ to the ground truth(s), and the best distance is used to compare two hierarchies H_1 and H_2. Many measurements of distance based on region and based on contour are available and reviewed in [Arbelaez 2011].

In this chapter, we propose an application of the hierarchy transformation aspect (see Section 3.6) in the framework of shape-based morphology. As shown in Figure 3.13, the partitions obtained by thresholding the new saliency map using extinction value $\mathcal{M}_{\mathcal{E}}$ might be different from any partition given by the initial hierarchy H, which means

$$\exists \mathcal{P} \in \mathcal{P}art(\mathcal{M}_{\mathcal{E}}) \text{ such that } \mathcal{P} \in \mathcal{P}art(\Omega), \mathcal{P} \notin \mathcal{P}art(H). \qquad (8.3)$$

Consequently, the scheme of hierarchy transformation described in Section 3.6 does not only modify the index of the input hierarchy H, but also modifies the structures of H. This might give interesting results in some cases. This new saliency map $\mathcal{M}_{\mathcal{E}}$ relies on the extinction value that measures the importance of each interesting region compared with its neighbor regions in the hierarchy H. In this chapter, as a first example, we set the input hierarchy as the α-tree, which means in the scheme of shape-based morphology $\mathcal{T} = \alpha$-tree. From a algorithmic point of view, the use of the hierarchy of constrained connectivity in [Soille 2008] is using the local range α or global range ω as the attribute function $\mathcal{A}$ being increasing attribute. Whereas, we propose to use an attribute function $\mathcal{A}$ (e.g., the context-based energy estimator described in Section 7.3) measuring the meaningfulness of each region in H. This attribute function $\mathcal{A}$ is usually of type non-increasing. The saliency map obtained by the hierarchy transformation described in Section 3.6 represents a new hierarchy of segmentation being different from the input hierarchy H. In this sense, This application can be seen as an extension of the hierarchy of constrained connectivity. Experimental results on the dataset of BSDS500 demonstrates the usefulness of the hierarchy transformation aspect given by the framework of shape-based morphology.

The rest of this chapter is organized as follows. In Section 8.2, we will give a short review of the works related to the the minimum spanning tree (MST) and α-tree which can be seen as the Min-tree of MST. Then in Section 8.3, we will detail how to extend the constrained connectivity with the framework of shape-based morphology using an attribute function $\mathcal{A}_f$ inspired from the work of Felzenszwalb and Huttenlocher in [Felzenszwalb 2004]. In Section 8.4, we will show some experimental results applied on the dataset of BSDS500. The qualitative and quantitative results demonstrates the usefulness of the proposed method. Finally we conclude in Section 8.5.

8.2   Related work

In this section, we will review several works related to a widely used structure in image processing: the Minimum Spanning Tree (MST) reviewed in Section 2.3.6. Firstly, we will review in Section 8.2.1 the work of Najman et al. in [Najman 2013] which states that that α-tree reviewed in Section 2.3.4 is equivalent to the Min-tree of MST in the algorithmic view. Then in Section 8.2.2, we will review an popular efficient image segmentation method relying on MST [Felzenszwalb 2004]. In Section 8.2.3, a recent work that propose a hierarchical version of the method in [Felzenszwalb 2004] will be presented.

8.2.1   Another point of view of α-tree

As reviewed in Section 2.3.4, the α-tree, known also as the hierarchy of quasi-flat zones, or constrained connectivity is based on the notion of α-connectivity defined by Eq (2.9). This notion of α-connectivity was originally proposed by Soille in [Soille 2008], and it has been shown efficient for hierarchical image partitioning and simplification. However, the algorithm published in [Soille 2008] only computes one level of the hierarchy which corresponds to a cut of the hierarchy based on the local range α or the global range ω (see Section 2.3.4, [Soille 2008], [Ouzounis 2011a] and [Ouzounis 2012b] for more details). A first algorithm computing the whole hierarchy was proposed in [Najman 2011] relying on the Min-tree of the edge-weighted graph (G, F_e). Recently, a similar algorithm having a complexity lower the one proposed in [Najman 2011] and being more memory efficient was proposed in [Najman 2013]. We will detail this algorithm in the following.

The algorithm proposed in [Najman 2013] relies on a tree structure called the *binary partition tree by altitude ordering* Q_{BT} while computing the MST on an edge-weighted graph (G, F_e). The main idea of Q_{BT} is the following: initially, each vertex is considered as a node represented by the leaves of Q_{BT}. During the MST

(a) Edge-weighted graph

(b) First nodes of Q_{BT}

(c) Adding a node to Q_{BT}

(d) Final Q_{BT}
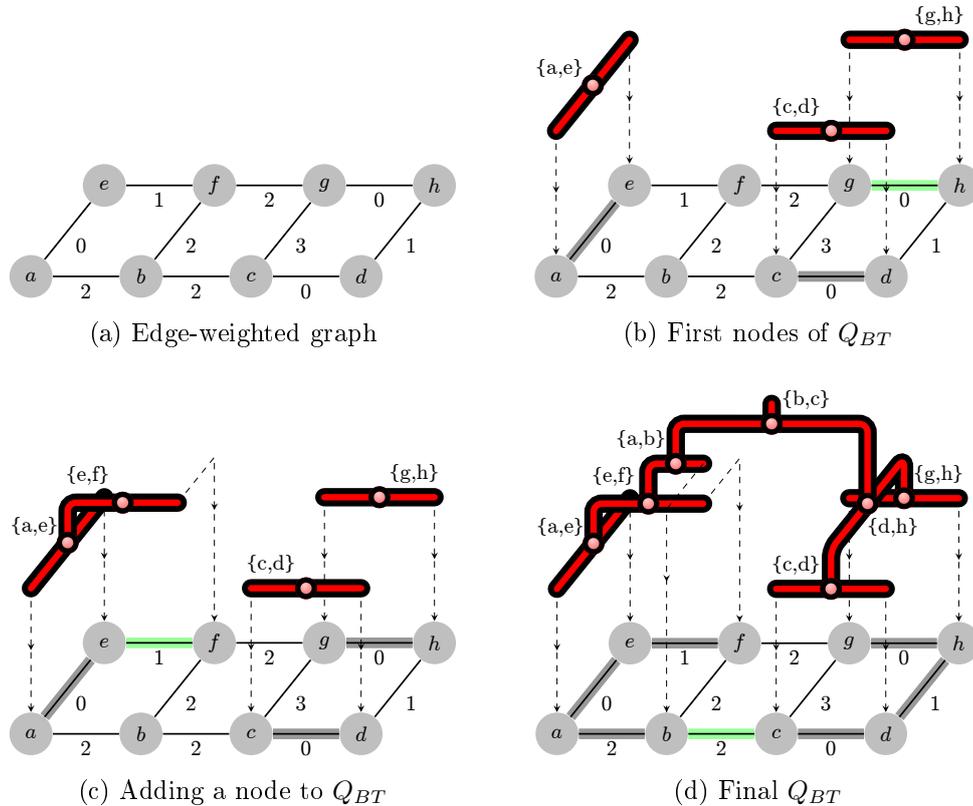
Figure 8.1: A simple process for obtaining Q_{BT} [Najman 2013], a binary tree providing a strict total order relation on the edges of the MST.

construction in Kruskal algorithm [Kruskal 1956], each time an edge $\{x, y\}$ is put into the MST, i.e. each time a union is made, we create a new node whose children are the two disjoint sets containing x and y. This added node becomes the canonical element of the union of these two points. An example of Q_{BT} is shown in Figure 8.1.

This binary partition tree by altitude ordering Q_{BT} can be constructed efficiently based on the union-find process after increasing sorting of the edges in the edge-weighted graph (G, F_e). It is similar with the algorithm of the Min-tree construction which will be described in Section 9.1. In fact, it is equivalent to the Min-tree constructed on the edges without canonization. The edges of Q_{BT} (i.e., the nodes which do not lie on the leaves of Q_{BT}) forms the MST. As shown in [Najman 2013], a post-processing of canonization of Q_{BT} by grouping the neighboring nodes (i.e., edges in (G, F_e)) in Q_{BT} with the same weight is applied, and the output of this post-processing is a tree structure in which each node represents a α-connected component. In fact, the output tree is the α-tree, the Q_{BT} construction along with

the canonization process is a Min-tree of the MST (see the Min-tree algorithm in Section 9.1). Consequently, in a algorithmic point of view, the α-tree is equivalent to a Min-tree of the MST.

8.2.2 Efficient graph-based image segmentation relying on MST

In [Felzenszwalb 2004], Felzenszwalb and Huttenlocher proposed an efficient graph-based image segmentation approach relying on the MST. The basic idea of the method is to merge regions linked by the edges belongs to the MST. To define the merging criterion, they introduced two measurements: the *internal difference* measuring the intra-dissimilarity and the *difference between* two components measuring the inter-dissimilarity.

For a connected component $X \subseteq V$, the internal difference $Int(X)$ is defined as the largest weight in the MST of the component, $MST(X, E)$. That is

$$Int(X) = \max_{e \in MST(X,E)} F_e(e). \tag{8.4}$$

The difference between two connected components $X, Y \subseteq V$, $Diff(X, Y)$ is defined as the minimum weight edge that connects the two components. It is given by

$$Diff(X, Y) = \min_{v_i \in X, v_j \in Y, (v_i,v_j) \in E} F_e(v_i, v_j). \tag{8.5}$$

If there is no edge connecting X and Y, we set $Diff(X, Y) = \infty$.

With the definition of $Int(X)$ and $Diff(X, Y)$, the merging criterion is described as follows: Two regions X and Y are merged if

$$Diff(X, Y) \le \min\{Int(X) + \frac{k}{|X|}, Int(Y) + \frac{k}{|Y|}\}, \tag{8.6}$$

where, $|\cdot|$ denotes the area, and k is some constant parameter. That is for small components, a stronger evidence for a boundary is required. In other words, a large k favors the merging of small regions.

The algorithm proposed in [Felzenszwalb 2004] can be sketched as follows: Firstly, compute the MST on a edge-weighted graph (G, F_e) of an image f. Initially, each leaf node (i.e. an individual point is considered as an individual region), then for each edge $e \in$ MST linking two vertices x and y in non-decreasing order of their weight, perform the following operations:

(i) Find the largest component X containing x.

(ii) Find the largest component Y containing y.

<table>
<tr><td>(a) Input image</td><td>(b) $k = 7500$ (8 regions)</td><td>(c) $k = 9000$ (14 regions)</td></tr>
</table>
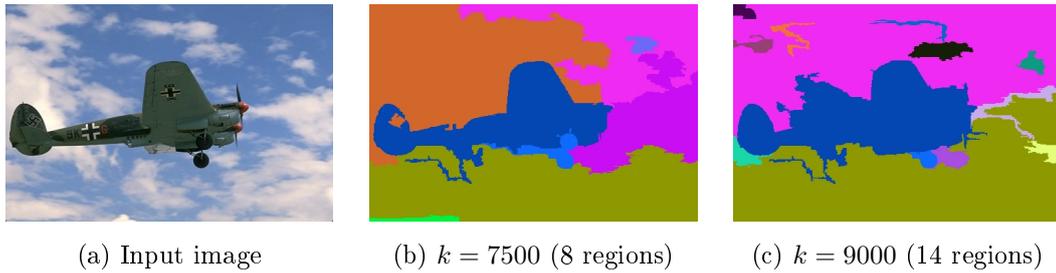
Figure 8.2:   An example illustrating the violation of the causality principle by [Felzenszwalb 2004]: the number of regions (in parentheses) increases from 8 to 14, instead of decreasing when the parameter k increases.

(iii) If $X = Y$, which means the two vertices x and y are already in the same component, do nothing. Otherwise, check the merging criterion between component X and Y, if the condition in Eq (8.6) holds, merge X and Y.

At the end of the algorithm, an image partition given by those components is obtained.

   Note that the parameter k plays a very important role in the algorithm. However, this parameter k is not a scale parameter according to the causality principle, which says that a boundary present at a scale k_1 should also be present at any scale $k_2 < k_1$. This is not true for this efficient segmentation algorithm. For instance, the boundaries of the region above the head of the plane in Figure 8.2 (c) obtained with $k = 9000$ are not present in Figure 8.2 (b) obtained with $k = 7500$. Moreover, the number of regions increase with a bigger k, which is a non-desired property and makes also the tunning of the parameter a difficult task.

8.2.3   Hierarchical graph based image segmentation

The first algorithm to produce a hierarchical version of the method in [Felzenszwalb 2004] was proposed by Haxhimusa and Kropatsch in [Haxhimusa 2004], which is an iterative version of [Felzenszwalb 2004] using a threshold function, and a tunning of the threshold parameter is required. Recently, Guimaraes et al. proposed in [Guimarães 2012] an efficient hierarchical graph-based image segmentation based on the method in [Felzenszwalb 2004]. Observe that the merging criterion given by Eq (8.6) depends on the parameter k at which the regions X and Y are observed, they propose to consider the scale $S_Y(X)$ of X relative to Y as a measure for which X will first merge with Y. It is given by

$$S_Y(X) = \big(Diff(X,Y) - Int(X)\big) \times |X|. \qquad (8.7)$$

In a symmetrical way, the scale $S_X(Y)$ measures when Y will merge with X can be defined. Then they define the scale $S(X, Y)$ by

$$S(X, Y) = \max(S_Y(X), S_X(Y)). \tag{8.8}$$

Based on this notion of scale $S(X, Y)$, the merging criterion in Eq (8.6) can be interpreted as follows: the neighboring region X and Y will merge when their scale is less than the threshold parameter k.

Let $\mathcal{P}_\lambda^{F_e}$ of V denotes the set of connected components of the graph made by V and the edges of weight lower than λ. Then based on these notions, their proposed algorithm is given as follows: It compute a saliency map being a weight map L (scales of observations). For every edge e_i, the weight map $L(e_i)$ is initialized to ∞. Then for each edge e_i linking two vertices x and y in non-decreasing order of their weights F_e, perform the following steps:

(i) Find the connected component X of $\mathcal{P}_{F_e(e_i)}^{F_e}$ containing x.

(ii) Find the connected component Y of $\mathcal{P}_{F_e(e_i)}^{F_e}$ containing y.

(iii) Compute the hierarchical observation scale $L(e_i)$.

The hierarchical observation scale $L(e_i)$ computation at step (iii) is obtained via the notion of hierarchical scale $S_Y'(X)$ of X relative to Y, defined as the lowest observation scale at which some sub-region of X, denoted as X^*, will merge with Y. More precisely, using an internal parameter ν, $S_Y'(X)$ is computed by following steps:

(1) Initialize the value of $\nu = 0$.

(2) Increment the value of ν by 1.

(3) Find the region X^* of $\mathcal{P}_\nu^L$ that contains x.

(4) repeat steps 2 and 3 while $S_Y(X^*) > \nu$.

(5) $S_Y(X^*) = \nu$

In a symmetrical way, $S_X'(Y)$ is equivalently defined. Then $L(e_i)$ is simply set to

$$L(e_i) = \max\{S_Y'(X), S_X'(Y)\}. \tag{8.9}$$

Thresholding this saliency map L with thresholding values from small to great yields a set of hierarchical partitions from thin to coarse.

8.3   Extending constrained connectivity

In practice, the use of the α-tree is usually based on the tree pruning strategy using the local range α or global range ω. For instance, in the original paper that introduces formally the notion of constrained connectivity, the whole hierarchy of α-tree is not computed, but a level of this hierarchy given by thresholding it according to the local range α and/or the global range ω. Note that both α and ω are instances of increasing attributes. And as shown in Section 8.2.1, in the algorithmic point of view, the α-tree is equivalent to the Min-tree of MST that can be efficiently obtained via binary partition tree by altitude ordering Q_{BT} by a step of post-processing. We propose to use the framework of shape-based morphology taking the binary partition tree by altitude ordering Q_{BT} as the input tree, and the attribute function $\mathcal{A}_f$ inspired from the work of Felzenszwalb and Huttenlocher in [Felzenszwalb 2004] that relies on the MST. This attribute function $\mathcal{A}_f$ measures the reluctancy of each merge that forms the parent node. It is a non-increasing attribute, and its computation will be detailed in Section 8.3.1. Then in Section 8.3.2, we will show the saliency map $\mathcal{M}_\mathcal{E}$ computation using extinction values defined on minima of the attribute function $\mathcal{A}_f$. This process can be seen as an extension of the hierarchy of constrained connectivity, in the sense that a non-increasing attribute function $\mathcal{A}_f$ is used instead of a increasing one. A new hierarchy of segmentation is obtained via this saliency map $\mathcal{M}_\mathcal{E}$. A refinement of the saliency map $\mathcal{M}_\mathcal{E}$ by removing the regions being too small will is also applied.

8.3.1   Shape space given by Q_{BT} with non-increasing attributes

In a binary partition tree by altitude ordering Q_{BT}, Each node in this tree Q_{BT} apart from the leaf nodes has exactly two children, and except the root node, each region merges with another region linked by an edge in MST, and forms a parent node. Based on the work in [Felzenszwalb 2004], this merging process is controlled by a constant parameter k of the merging criterion in Eq (8.6). Following the idea of the work in [Guimarães 2012], this merging criterion can be interpreted via the notion of scale $S(X,Y)$ defined in Eq (8.8), which says the neighboring region X and Y will merge when their scale is less than the given parameter k. We thus propose to use $S(X,Y)$ as the attribute function. For a given node $\mathcal{N}$, its attribute $\mathcal{A}_f(\mathcal{N})$ is given by

$$\mathcal{A}_f(\mathcal{N}) = \max\{\big(Diff(\mathcal{N}_{c_1}, \mathcal{N}_{c_2}) - Int(\mathcal{N}_{c_1})\big) \times |\mathcal{N}_{c_1}|,$$
$$\big(Diff(\mathcal{N}_{c_1}, \mathcal{N}_{c_2}) - Int(\mathcal{N}_{c_2})\big) \times |\mathcal{N}_{c_2}|\}, \qquad (8.10)$$

where $\mathcal{N}_{c_1}$, $\mathcal{N}_{c_2}$, denote the two children node of node $\mathcal{N}$. As described in Section 8.2.3, a big $\mathcal{A}_f$ means a large "observation scale", which is to say more meaningful in the hierarchy. Thus the maxima of the attribute function $\mathcal{A}_f$ correspond to meaningful regions.

8.3.2   Saliency map using extinction value

To cope the attribute function $\mathcal{A}_f$ with the hierarchy transformation aspect of the shape-based morphology framework, we use the attribute function $Attribute'_f$, for a given node $\mathcal{N}$, that is

$$\mathcal{A}'_f(\mathcal{N}) = \max_{\mathcal{N}_k \in Q_{BT}} \mathcal{A}_f(\mathcal{N}_k) - \mathcal{A}_f(\mathcal{N}), \tag{8.11}$$

so that the local minima of $\mathcal{A}'_f$ correspond to meaningful regions. Use the method of hierarchy transformation described in Section 3.6.2, we are able to compute a saliency map $\mathcal{M}^0_{\mathcal{E}}$ based on the extinction value of the attribute function $\mathcal{A}'_f$. An example of this saliency map $\mathcal{M}^0_{\mathcal{E}}$ is illustrated in Figure 8.3 (b) for the input image in Figure 8.3 (a). Observe that in this saliency map, there are many small regions that are very salient, a step of refinement is required. We propose to apply a grain filter followed by a topological watershed on the edges of the initial saliency map $\mathcal{M}^0_{\mathcal{E}}$ whose results for the image in Figure 8.3 are shown in (c) and respectively (d). For the image $\mathcal{M}^1_{\mathcal{E}}$ obtained after a topological watershed applied on the edges, the boundaries are included in the initial saliency map $\mathcal{M}^0_{\mathcal{E}}$, thus the contour preservation properties are still maintained. By lowering the catchment basins of $\mathcal{M}^1_{\mathcal{E}}$ to 0, we have the final saliency map $\mathcal{M}_{\mathcal{E}}$ which is illustrated in (e) for the example in Figure 8.3. One level of segmentation obtained by thresholding this final saliency map $\mathcal{M}_{\mathcal{E}}$ is shown in Figure 8.3 (f).

8.4   Results

In this section, we will show some results about the proposed method in the framework of shape-based morphology, that extends the constrained connectivity. The experiments are conducted on the Berkeley Segmentation Dataset [Arbelaez 2011], an extension of the BSDS300 [Martin 2001]. The dataset consists of 500 natural images divided into 200 test images, 200 images for training, and 100 validation images, together with human annotations. Each image is segmented by an average of five different subjects. In Section 8.4.1, some qualitative results are illustrated, and quantitative benchmark on the BSDS500 will be detailed in Section 8.4.2, compared with the original method of Felzenszwalb and Huttenlocher in [Felzenszwalb 2004].

(a) Input image

(b) Initial saliency map $\mathcal{M}_{\mathcal{E}}^0$

(c) Filtered saliency map $\mathcal{M}_{\mathcal{E}}^1$

(d) Topological watershed on edges of $\mathcal{M}_{\mathcal{E}}^1$

(e) Final saliency map $\mathcal{M}_{\mathcal{E}}$

(f) One level of segmentation (11 regions)
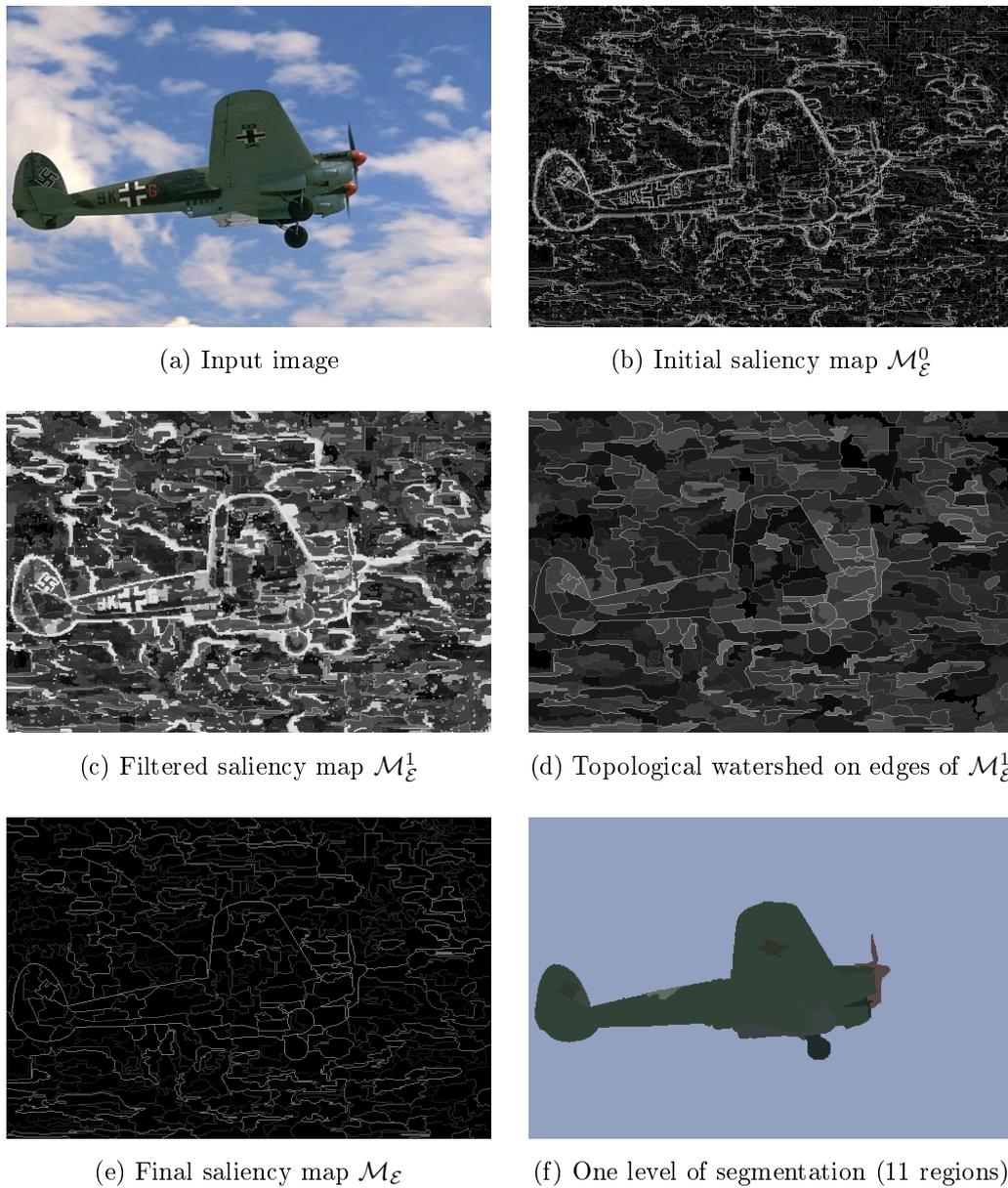
Figure 8.3: An example showing the scheme of the saliency map computation (a-e). (f): One level of segmentation by thresholding the final saliency map in (e).

8.4.1   Qualitative results

Some saliency maps $\mathcal{M}_{\mathcal{E}}$ obtained with the proposed method are illustrated in Figure 8.4, 8.5, 8.6, 8.7. For each image in those figures, two partitions are illustrated. They are given by thresholding the saliency map $\mathcal{M}_{\mathcal{E}}$ with the same fixed threshold level called *Optimal Dataset Scale (ODS)* for the whole dataset, and respectively the image dependent threshold level called *Optimal Image Scale (OIS)*. Most of the partitions are reasonable. However, note that This method of extending the constrained connectivity still belongs to the family of connected operators, so the boundaries' location remain intact, no boundaries regularization/smoothing process is applied, which explains the form of the boundaries (white curves) in those partitions.

8.4.2   Benchmark on the BSDS500

Quantitative evaluation is performed using the region-based performance measurements described in [Arbelaez 2011], in terms of Ground-Truth (GT) Covering criterion and Probabilistic Rand Index (PRI). The ground-truth covering measurement is defined relying on the *overlap* between two regions R and R' given by:

$$\hat{\mathcal{O}}(R, R') = \frac{|R \cap R'|}{|R \cup R'|}.\tag{8.12}$$

The overlap is widely used for the pixel-wise classification task in recognition [Malisiewicz 2007]. The covering of a segmentation $\mathcal{P}$ by a segmentation $\mathcal{P}'$ is defined as:

$$\hat{\mathcal{C}}(\mathcal{P}' \to \mathcal{P}) = \frac{1}{N} \sum_{R \in \mathcal{P}} |R| \cdot \sum_{R' \in \mathcal{P}'} \hat{\mathcal{O}}(R, R'),\tag{8.13}$$

where N is the total number of pixels in the image. A bigger ground-truth covering using the Eq (8.13) means that the partition is closer to the ground truth.

The probabilistic rand index measurement is based on the notion of Rand Index [Rand 1971] that was originally introduced for general clustering evaluation. The Rand Index between a partition $\mathcal{P}$ and a ground truth partition G is given by the sum of the number of pairs of pixels having the same label in $\mathcal{P}$ and G and those having different labels in both partitions, divided by the total number of pairs of pixels. Variants of Rand Index have been proposed [Unnikrishnan 2007, Yang 2008] to deal with the case of multiple ground truth partitions. For a partition $\mathcal{P}$ and a given set of ground truth partitions $\{G_k\}$, the PRI is defined as follows:

$$PRI(\mathcal{P}, \{G_k\}) = \frac{1}{T} \sum_{i<j} \big(c_{ij} p_{ij} + (1 - c_{ij})(1 - p_{ij})\big),\tag{8.14}$$

where c_{ij} is the event that pixel i, j have the same label and p_{ij} its probability. T denotes the total number of pixel pairs. Using the sample mean to estimate p_{ij}, the
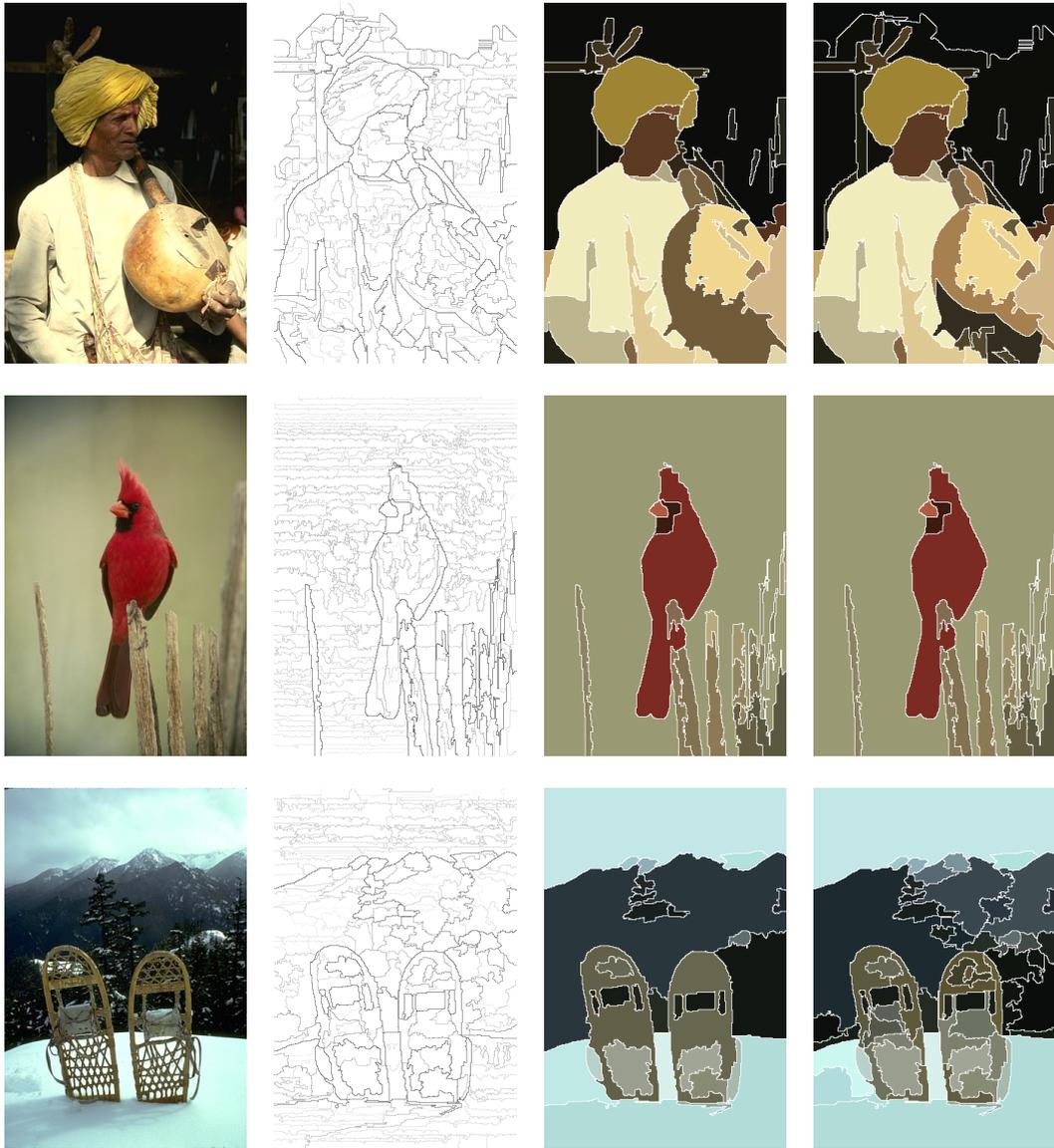
Figure 8.4: Some hierarchical segmentation results on the BSDS500. From left to right: Input image, saliency map, and segmentations at the optimal dataset scale (ODS) and at the optimal image scale (OIS).
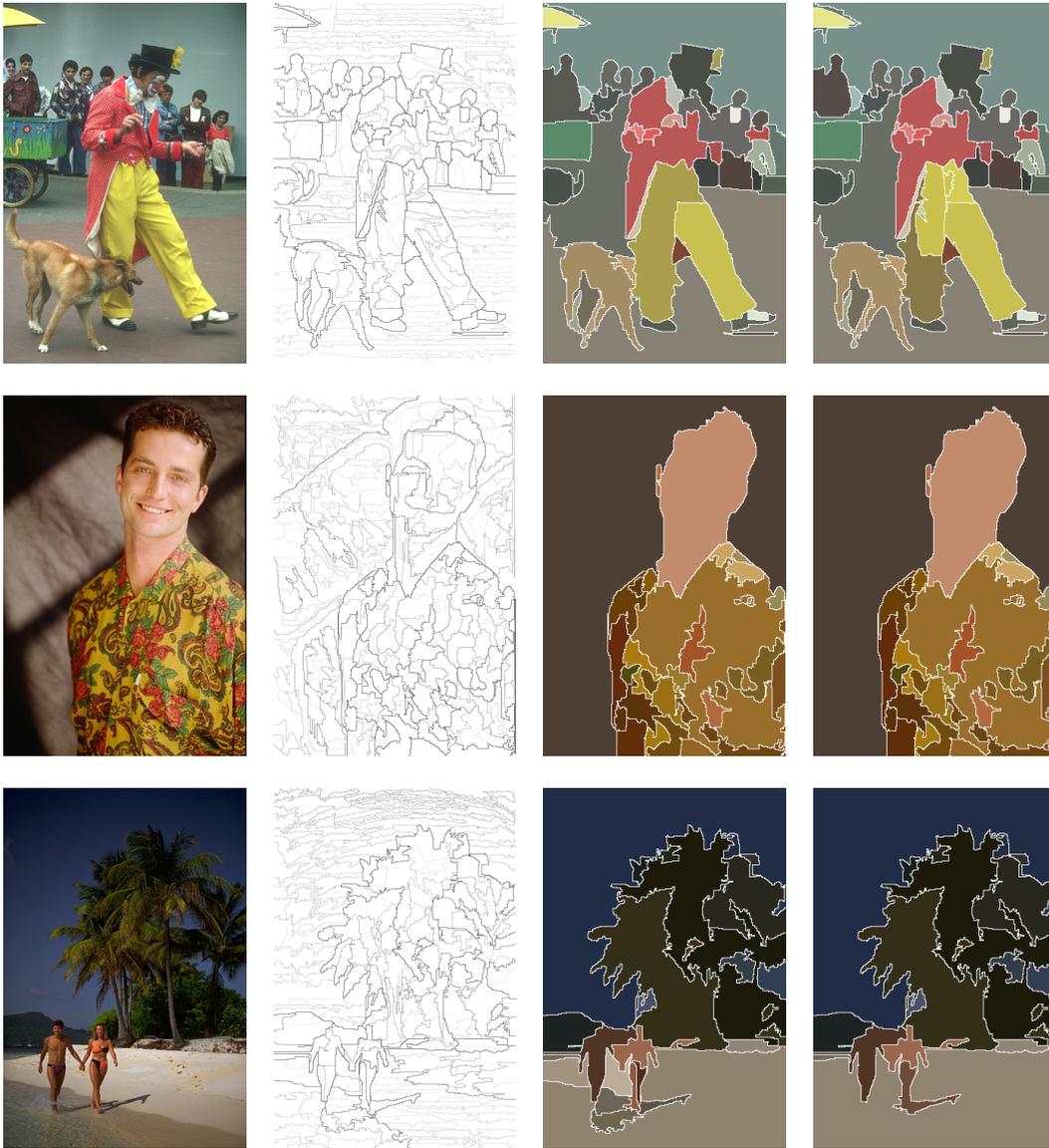
Figure 8.5: Some hierarchical segmentation results on the BSDS500. From left to right: Input image, saliency map, and segmentations at the optimal dataset scale (ODS) and at the optimal image scale (OIS).
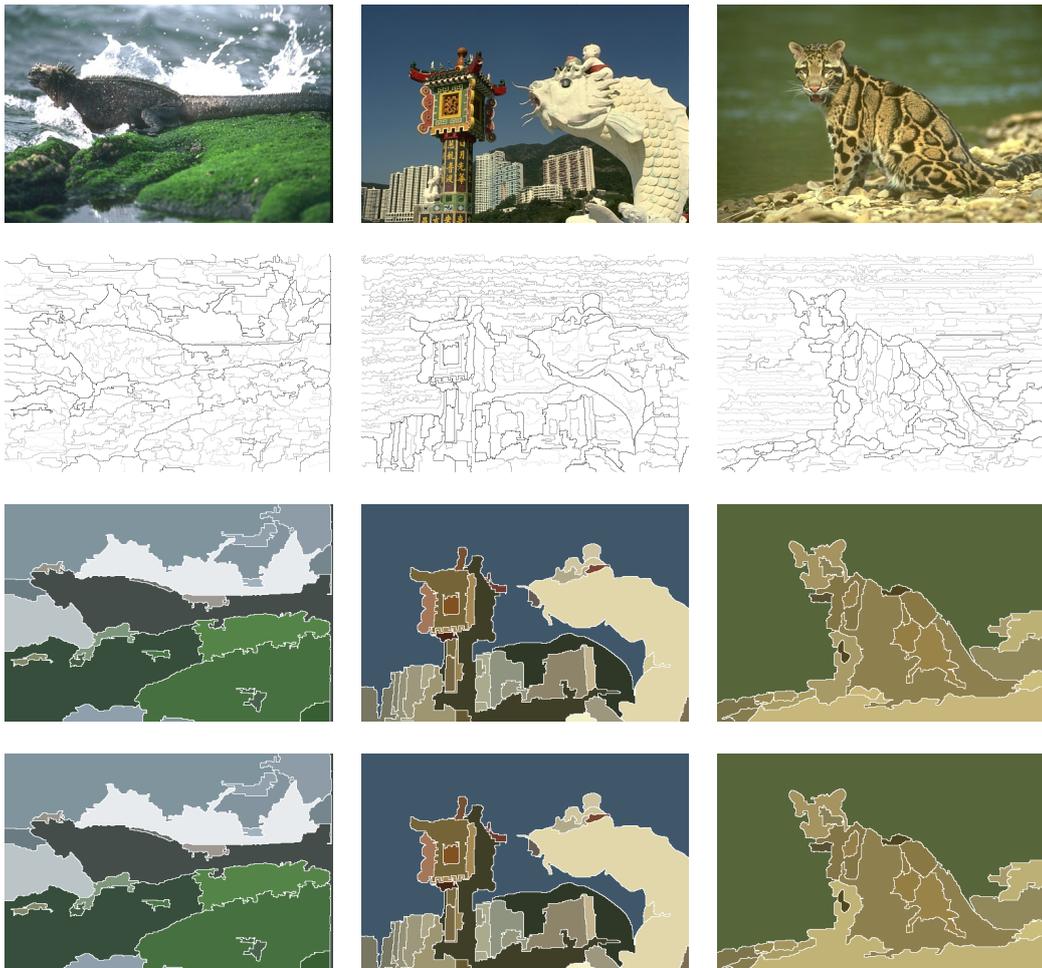
Figure 8.6: Some additional hierarchical segmentation results on the BSDS500. From top to bottom: Input mage, saliency map, and segmentations at the optimal dataset scale (ODS) and at the optimal image scale (OIS).

Figure 8.7: Some additional hierarchical segmentation results on the BSDS500. From top to bottom: Input mage, saliency map, and segmentations at the optimal dataset scale (ODS) and at the optimal image scale (OIS).

Method	GT Covering			Prob. Rand. Index	
	ODS	OIS	Best	ODS	OIS
FH [Felzenszwalb 2004]	0.43	0.53	0.68	0.76	0.79
Guimarães [Guimarães 2012]	0.46	0.53	0.60	0.76	0.81
Ours	**0.50**	**0.57**	**0.66**	**0.77**	**0.82**

Table 8.1: Preliminary region benchmarks on the BSDS500.

definition of PRI in Eq (8.14) amounts to averaging the Rand Index among different ground truth partitions. A partition $\mathcal{P}$ having a bigger PRI means that $\mathcal{P}$ is closer to the set of ground truth partitions.

Here, we compare our results with the graph-based image segmentation (Felz-Hutt) [Felzenszwalb 2004], and with another the method of hierarchical graph based image segmentation proposed by (Guimarães *et al.*) [Guimarães 2012], also relying on the same criterion popularized by [Felzenszwalb 2004]. The comparison is given in Table 8.1. Our method ranks first, for both the optimal dataset scale (ODS) and for optimal image scale (OIS).

8.5   Conclusion

This chapter has presented one application of the aspect of hierarchy transformation in the framework of shape-based morphology: an extension of the constrained connectivity framework to non-increasing constraints. The first tree representation $\mathcal{T}$ used in the framework of shape-based morphology is the binary partition tree by altitude ordering from which the α-tree can be efficiently obtained by a simple post-processing. The used attribute function $\mathcal{A}$ is inspired from the work of efficient graph-based image segmentation in [Felzenszwalb 2004]. The experienced second attribute function $\mathcal{A}\mathcal{A}$ is the height of $\mathcal{A}$. The scheme of hierarchy transformation yields a saliency map that represents a set of partitions from fine to coarse, which are generally different from the cuts of the input α-tree. Qualitative results show that a partition being close to the ground truth can be obtained by thresholding the produced saliency map. Besides, a quantitative evaluation based on the Ground Truth Covering and the Probability Rand Index demonstrates that our approach compares favorably to previous works.

Part IV

Algorithm

Algorithms to compute information on the tree

All the applications discussed above relying on the framework of shape-based morphology were implemented using our C++ image processing library Olena [Levillain 2009, Levillain 2012a, Levillain 2010, Levillain 2012b], which is a generic and efficient image processing framework, available on the Internet as free software [1]. In this chapter, we will detail those algorithms. We will start with the algorithm for tree construction in Section 9.1 that relies on the union-find process. Some additional information to prepare the attribute $\mathcal{A}$ computation will also be presented in this section. Then in Section 9.2, we will show how to efficiently compute the attributes relying on the accumulated information on region, contour, and context. We will then illustrate the computation of the minimal information along the contour in Section 9.3, that is the case of the number of false alarms (NFA). In section 9.4, we will show the algorithm for saliency map computation relying on the extinction values. Finally, in Section 9.5, we will present an algorithm of disjoint level lines selection relying on the topographic map representation. This algorithm provides an image simplification method that make the image well-composed without doubling the image size.

9.1 Tree construction

As mentioned in Section 8.2.1, the computation of the binary partition tree by altitude ordering Q_{BT} can be implemented by the Min-tree construction applied on the pixel edges lying between pixels instead of pixels themselves, so does the α-tree, which can be obtained by a post-processing of Q_{BT}. So in this section, we focus on the algorithms to compute the three morphological trees, i.e., Min-tree, Max-tree, and the topographic map. There exist several algorithms to compute the Min/Max-tree that can be divided into two categories: the flooding process based approaches [Salembier 1998, Nistér 2008] and the union-find based approaches [Najman 2006, Berger 2007]. A detailed comparison of those Min/Max-

---

[1]http://olena.lrde.epita.fr

tree computation algorithms is presented in [Carlinet 2013]. We will detail the algorithm based on union-find process. Concerning the topographic map, there are four different algorithms. The first one is the fast level lines transform (FLLT) proposed by Monasse and Guichard in [Monasse 2000b]. Then Song proposed a topdown algorithm for computation of level lines in [Song 2007]. Another algorithm was proposed by Caselles and Monasse [Caselles 2009]. The complexity of these three algorithms is $O(N^2)$, and they are hard to implement. Recently, Géraud et al. proposed a quasi-linear algorithm for the topographic map computation [Géraud 2013]. Here we will detail this algorithm which is also based on the union-find process.

These union-find based algorithms feature a common scheme of process that is composed of two steps given as follows:

1) Sort the pixels in the *decreasing tree order*.

2) In the *reverse order*, rely on the union-find process to compute the tree.

For those three different trees, they differ in the sorting step. All of them have a quasi-linear complexity $O(\alpha(N) \times N)$ for low quantized image, where $\alpha()$ is the extremely slowly growing inverse of the single-valued Ackermann function. Now let us start with the union-find algorithm, which is given in Algorithm 1.

For a low quantized image f, we can use Bucket sort algorithm to sort the pixels in increasing order or decreasing order. Then the Min-tree and Max-tree representation can be obtained by Algorithm 2, where SORT_MONOTONE Is either a increasing sorting (Max-tree) or a decreasing sorting (Min-tree).

Concerning the topographic, the sorting step is more complicated. It first interpolate the scalar image to an image of range using the Khalimsky's grid (see Figure 3.12). Then the sorting step is based on a priority queue q, and two operations of the priority queue q defined in Algorithm 3. Then the sorting algorithm is given in Algorithm 4, and the algorithm for computing the topographic map is depicted in Algorithm 5. More details about this quasi-linear algorithm that computes the topographic map can be found in the work of Géraud et al. [Géraud 2013].

9.2   Attributes accumulated on region, contour and context

Most of the attributes used in those applications described in Part III can be computed incrementally. They rely on some information that can be accumulated during the tree construction. We distinguish those information into three categories: information accumulated on region (see Section 9.2.1), on contour (see Section 9.2.2), and

Explanation of variables:

$\mathcal{R}$	set of sorted pixels
N	total number of pixels
p, q, n, x	pixel
$\mathcal{N}$	neighbors
$parent, zpar$	image of parenthood, temporary parenthood

input : A set of sorted pixels $\mathcal{R}$.

output: Image of parenthood $parent$.

FIND_ROOT($zpar, x$)

begin

    if $zpar(x) = x$ **then**

        return x

    else

        $zpar(x) \leftarrow$ FIND_ROOT($zpar, zpar(x)$)

        return $zpar(x)$

    end

end

UNION_FIND($\mathcal{R}$)

begin

    for all p **do**

        $zpar(p) \leftarrow$ undef

    end

    for $i \leftarrow N - 1$ **to** 0 **do**

        $p \leftarrow \mathcal{R}[i]$

        $parent(p) \leftarrow p$

        $zpar(p) \leftarrow p$

        for all $n \in \mathcal{N}(p)$ **such as** $zpar(n) \neq$ undef **do**

            $r \leftarrow$ FIND_ROOT($zpar, n$)

            if $r \neq p$ **then**

                $parent(r) \leftarrow p$

                $zpar(r) \leftarrow p$

            end

        end

    end

    return $parent$

end

Algorithm 1: Union-find process.

input : An image f.

output: Image of parenthood *parent*.

CANONIZE_TREE($f, \mathcal{R}, parent$)

begin

    for $i \leftarrow 0$ **to** $N - 1$ **do**

        $p \leftarrow \mathcal{R}[i]$

        $q \leftarrow parent(p)$

        if $f(parent(q)) = f(q)$ **then** $parent(p) \leftarrow parent(q)$

    end

    return *parent*

end

COMPUTE_MIN_MAX_TREE(f)

begin

    $\mathcal{R} \leftarrow$ SORT_MONOTONE(f)

    parent $\leftarrow$ UNION_FIND($\mathcal{R}$)

    parent $\leftarrow$ CANONIZE_TREE($f, \mathcal{R}, parent$)

    return *parent*

end

Algorithm 2: Min-tree and Max-tree construction.

on context (see Section 9.2.3). For the purpose of simplicity, we consider the Min-tree or Max-tree representation. The attributes computation for the topographic map representation shares the same principle.

9.2.1 Attributes accumulated on region

In the tree construction process, the algorithm starts with the pixels lying on the leaves, and the union-find process acts as kind of region merging process. So the connected components in the tree representation are created by region growing process. We are capable to handle the information accumulated on region efficiently, such as the area, the sum of gray level or sum of square of gray level that can be used to compute the mean and variance inside each region, the moments of each region based on which we can compute some shape attributes $\mathcal{A}_s$, e.g., the elongation defined in Eq (5.31), and the attribute measuring how much a given shape fits an ellipse defined in Eq (6.1). Taking the area, sum of gray level, and sum of coordinates x as examples, we can add some operations to the union-find process during the tree construction. It is given in Algorithm 6.

Explanation of variables:

$\mathcal{R}$	set of sorted d-faces $(d = 0, 1, 2)$
q	priority queue
h, n	d-face in Khalimsky's grid $(d = 0, 1, 2)$
F	Set-valued map
l, l'	gray level

PRIORITY_PUSH(q, h, F, l)

/* modify q */

begin
    $[lower, upper] \leftarrow F(h)$
    if $lower > l$ **then**
    | $l' \leftarrow lower$
    else if $upper < l$ **then**
    | $l' \leftarrow l$
    else
    | $l' \leftarrow l$
    end
    PUSH$(q[l'], h)$
end

PRIORITY_POP(q, l)

/* modify q, and sometimes l */

begin
    if $q[l]$ is empty **then**
    | $l' \leftarrow$ level next to l such as $q[l']$ is not empty
    | $l \leftarrow l'$
    end
    return POP$(q[l])$
end

Algorithm 3: Push and Pop operation of the priority queue q.

input   : A set-valued map.

output: A pair of sorted d-faces $\mathcal{R}$ and a interpolated image f^b.

SORT(F)

begin

    for all h **do**

      | $deja\_vu(h) \leftarrow$ false

    end

    $i \leftarrow 0$

    PUSH($q[l_\infty], p_\infty$)

    $deja\_vu(p_\infty) \leftarrow$ true

    $l \leftarrow l_\infty$ /* start from root level */

    while q is not empty **do**

        $h \leftarrow$ PRIORITY_POP(q, l)

        $f^b(h) \leftarrow l$

        $\mathcal{R}[i] \leftarrow h$

        for all $n \in \mathcal{N}(h)$ **such as** $deja\_vu(n) =$ false **do**

           | PRIORITY_PUSH(q, n, F, l)

           | $deja\_vu(n) \leftarrow$ true

        end

        $i \leftarrow i + 1$

        /* if $q[l]$ is empty, we are done with level l */

    end

    return $(\mathcal{R}, f^b)$

end

Algorithm 4: Sorting step for computing the topographic map.

---

input   : An image f.

output: The topographic map representation *parent*.

COMPUTE_TOPOGRAPHIC_MAP(f)

begin

    $F \leftarrow$ INTERPOLATE(f)

    $(\mathcal{R}, f^b) \leftarrow$ SORT(F)

    $parent \leftarrow$ UNION_FIND($\mathcal{R}$)

    $parent \leftarrow$ CANONIZE_TREE($f^b, \mathcal{R}, parent$)

    return *parent*

end

Algorithm 5: Topographic map computation.

Explanation of variables:

area	image of area
sum_f	image of sum of gray level
sum_x	image of sum of coordinates x

UNION_FIND($\mathcal{R}$)
begin
    for all p **do**
        $zpar(p) \leftarrow$ undef
        $area(p) \leftarrow 0$
        $sum\_f(p) \leftarrow 0$
        $sum\_x(p) \leftarrow 0$
    end
    for $i \leftarrow N - 1$ **to** 0 **do**
        $p \leftarrow \mathcal{R}[i]$
        $parent(p) \leftarrow p$
        $zpar(p) \leftarrow p$
        $area(p) \leftarrow 1$
        $sum\_f(p) \leftarrow f(p)$
        $sum\_x(p) \leftarrow p.row()$
        for all $n \in \mathcal{N}(p)$ **such as** $zpar(n) \neq$ undef **do**
            $r \leftarrow$ FIND_ROOT($zpar, n$)
            if $r \neq p$ **then**
                $parent(r) \leftarrow p$
                $zpar(r) \leftarrow p$
                $area(p) \leftarrow area(p) + area(r)$
                $sum\_f(p) \leftarrow sum\_f(p) + sum\_f(r)$
                $sum\_x(p) \leftarrow sum\_x(p) + sum\_x(r)$
            end
        end
    end
    return $parent$
end

Algorithm 6: Incremental computation of region-based information.

9.2.2   Attributes accumulated on contour

Besides the efficient computation of those attributes accumulated on region, the attributes accumulated on contour can be managed in the same way during the union-find process that can be seen as a region growing process. The basic idea is that every time a pixel p is added to the current region to form a parent region, verify its four pixel edges, that is the neighbors (4-connectivity $\mathcal{N}_4(p)$) of the current pixel in the Khalimsky's grid. If a pixel edge e is already added to the current region (i.e., belongs to its boundary), then remove e after adding p, since that pixel edge e will be inside the parent region, consequently it is no longer on the boundary. Otherwise, add this pixel edge e. This process is illustrated in Figure 7.2 (a). It is based on the use of an image defined on pixel edges that indicates if the pixel edge belongs to the boundary of some region. Taking the length of region boundary, the sum of gradient's magnitude as examples, the algorithm to handle the contour information is illustrated in Algorithm 7 by adding some operations to the union-find process. The attribute of average of gradient's magnitude and curvature along the region boundary defined in Eq (3.8) and respectively Eq (5.28) can then be easily obtained.

9.2.3   Attributes accumulated on context

The approximated context-based information illustrated in Figure 7.2 (b) relies on the region boundary. In fact, using this approximation, for a given shape τ and a thickness ε, the area of the interior region $\mathcal{R}_{in}^{\varepsilon}(\partial\tau)$ and exterior region $\mathcal{R}_{out}^{\varepsilon}(\partial\tau)$ are equivalent, and given by

$$\mathcal{R}_{in}^{\varepsilon}(\partial\tau) = \mathcal{R}_{out}^{\varepsilon}(\partial\tau) = \varepsilon|\partial\tau|, \tag{9.1}$$

where $|\partial\tau|$ denotes the contour length of region represented by τ. The attributes accumulated on context can be computed in the same way as the computation of attributes accumulated on contour. But we have to pay attention to the interior and exterior information while doing the update operation. The algorithm for computing the sum of gray level inside the exterior region $\mathcal{R}_{out}^{\varepsilon}(\partial\tau)$ is shown in Algorithm 8, which relies on two pre-computed images *con_top_right* and *con_down_left* defined on pixel edges e that encodes the accumulated information of the ε pixels above (horizontal pixel edge) or on the right side (vertical pixel edge) of e, and respectively below (horizontal pixel edge) or on the left side (vertical pixel edge) of e. Other accumulated information on context required in Eq (7.16) for the context-based energy estimator can be computed in the same way.

Explanation of variables:

ima_grad	pre-computed image of gradient's magnitude on pixel edges
is_boundary	image indicating whether a pixel edge e lies in boundary
length	image of length of region boundary
sum_grad	image of sum of gradient's magnitude along the boundary

UNION_FIND($\mathcal{R}$)
begin
    for all p **do**
        $zpar(p) \leftarrow$ undef
        $length(p) \leftarrow 0$
        $sum\_grad(p) \leftarrow 0$
    end
    for all e **do** $is\_boundary(e) \leftarrow$ false
    for $i \leftarrow N-1$ **to** 0 **do**
        $p \leftarrow \mathcal{R}[i]$
        $parent(p) \leftarrow p$
        $zpar(p) \leftarrow p$
        for all $n \in \mathcal{N}(p)$ **such as** $zpar(n) \neq$ undef **do**
            $r \leftarrow$ FIND_ROOT($zpar, n$)
            if $r \neq p$ **then**
                $parent(r) \leftarrow p$
                $zpar(r) \leftarrow p$
                $length(p) \leftarrow length(p) + length(r)$
                $sum\_grad(p) \leftarrow sum\_grad(p) + sum\_grad(r)$
            end
        end
        for all $e \in \mathcal{N}_4(p)$ **do**
            if $!is\_boundary(e)$ **then**
                $is\_boundary(e) \leftarrow$ true
                $length(p) \leftarrow length(p) + 1$
                $sum\_boundary(p) \leftarrow sum\_boundary(p) + ima\_grad(e)$
            else
                $is\_boundary(e) \leftarrow$ false
                $length(p) \leftarrow length(p) - 1$
                $sum\_boundary(p) \leftarrow sum\_boundary(p) - ima\_grad(e)$
            end
        end
    end
    return *parent*
end

Algorithm 7: Incremental computation of contour-based information.

Explanation of variables:

con_top_right	pre-computed image of context on pixel edges
con_down_left	pre-computed image of context on pixel edges
sum_f_ext	image of sum of gray level in exterior region $\mathcal{R}^{\varepsilon}_{out}(\partial\tau)$

UNION_FIND($\mathcal{R}$)
begin
 for all p **do**
  $zpar(p) \leftarrow$ undef
  $sum\_f\_ext(p) \leftarrow 0$
 end
 for all e **do** $is\_boundary(e) \leftarrow$ false
 for $i \leftarrow N-1$ **to** 0 **do**
  $p \leftarrow \mathcal{R}[i]$
  $parent(p) \leftarrow p$
  $zpar(p) \leftarrow p$
  for all $n \in \mathcal{N}(p)$ **such as** $zpar(n) \neq$ undef **do**
   $r \leftarrow$ FIND_ROOT($zpar, n$)
   if $r \neq p$ **then**
    $parent(r) \leftarrow p$
    $zpar(r) \leftarrow p$
    $sum\_f\_ext(p) \leftarrow sum\_f\_ext(p) + sum\_f\_ext(r)$
   end
  end
  for all $e \in \mathcal{N}_4(p)$ **do**
   if $!is\_boundary(e)$ **then**
    $is\_boundary(e) \leftarrow$ true
    if e is above or on the right of p **then**
     $sum\_f\_ext(p) \leftarrow sum\_f\_ext(p) + con\_top\_right(e)$
    else
     $sum\_f\_ext(p) \leftarrow sum\_f\_ext(p) + con\_down\_left(e)$
    end
   else
    $is\_boundary(e) \leftarrow$ false
    if e is above or on the right of p **then**
     $sum\_f\_ext(p) \leftarrow sum\_f\_ext(p) - con\_down\_left(e)$
    else
     $sum\_f\_ext(p) \leftarrow sum\_f\_ext(p) - con\_top\_right(e)$
    end
   end
  end
 end
 return $parent$
end

Algorithm 8: Incremental computation of context-based information.

9.3 Attributes based on minimum information along the contour

Apart from those attributes based on the accumulated information, the number of false alarms (NFA) requires a minimal information, i.e., the minimal gradient's magnitude along the boundary of each region. Here we propose an efficient algorithm that handle this kind of attributes. It relies on two images *appear* and *vanish* defined on pixel edges that encode the node $\mathcal{N}_a$ for which a pixel edge e firstly appears as part of the boundary, and respectively the node $\mathcal{N}_v$ for which a pixel edge e is firstly no longer belonging to the boundary. Note that $\mathcal{N}_a$ and $\mathcal{N}_v$ might be equal, e.g., the case of pixel edges inside a flat zone. The computation of the minimal gradient's magnitude along the contour is depicted in Algorithm 9.

9.4 Saliency map computation

As discussed in Section 3.6.2, the saliency map introduced in the framework of shape-based morphology relies on the extinction value defined on the local minima, that has been reviewed in Section 3.2.3 and illustrated in Figure 3.5. Once we have the extinction value for all the minima, we can weight the extinction value to the boundary of the region corresponding to the minima. That is to say for each pixel edge, take the maximal extinction value of those minima for which this pixel edge is on their boundaries. This can be easily achieved via two images *appear* and *vanish* that have been used in Algorithm 9. Finally, a saliency map is obtained.

The basic idea for calculating the extinction value for each minimum is based on a Min-tree $\mathcal{TT}$ of the first tree $\mathcal{T}$ weighted with some attribute function $\mathcal{A}$, and a propagation of the minima (Hence the leaves of the Min-tree $\mathcal{TT}$) to the rest nodes. More specifically, for each minimum m_i, so a leaf, we propagate the nodes from this leaf to the root of the Min-tree $\mathcal{TT}$, if the node is not met yet by the propagation of any other minimum m_j, we say that the original of this node is m_i. Otherwise, let m_j be its current original minimum, if attribute of the original of this node $\mathcal{A}(m_h) > \mathcal{A}(m_i)$, update m_i as the original of this node. if for certain node, $\mathcal{A}(m_j) < \mathcal{A}(m_i)$, there is no need to do the update of original minimum, we can stop the propagation for this minimum. Once the propagation of all the minima is finished, taking the dynamic as an example, the extinction value for a minimum m_i is the difference between $\mathcal{A}(m_i)$ and the attribute of the highest ancestor of m_i on the Min-tree $\mathcal{TT}$ whose original minimum is still m_i. This process of saliency map computation relying on the extinction value is illustrated in Algorithm 10. The saliency map described in this algorithm is defined only on the pixel edges, i.e., 1-

Explanation of variables:

appear	image storing node for which e belongs firstly to its boundary
vanish	image storing node for which e is firstly not on its boundary
min_grad	image of minimal gradient's magnitude

UNION_FIND($\mathcal{R}$)

begin

    for all p **do**

        $zpar(p) \leftarrow$ undef

        $min\_grad(p) \leftarrow$ max_value

    end

    for all e **do** $is\_boundary(e) \leftarrow$ false

    for $i \leftarrow N - 1$ **to** 0 **do**

        $p \leftarrow \mathcal{R}[i], parent(p) \leftarrow p, zpar(p) \leftarrow p$

        for all $n \in \mathcal{N}(p)$ **such as** $zpar(n) \neq$ undef **do**

            $r \leftarrow$ FIND_ROOT($zpar, n$)

            if $r \neq p$ **then**

                $parent(r) \leftarrow p, zpar(r) \leftarrow p$

            end

        end

        for all $e \in \mathcal{N}_4(p)$ **do**

            if $!is\_boundary(e)$ **then**

                $is\_boundary(e) \leftarrow$ true, $appear(e) \leftarrow p$

            else

                $is\_boundary(e) \leftarrow$ false, $vanish(e) \leftarrow p$

            end

        end

    end

    for all e **do**

        $\mathcal{N}_a \leftarrow appear(e), \mathcal{N}_v \leftarrow vanish(e)$

        while $\mathcal{N}_a \neq \mathcal{N}_v$ **do**

            $min\_grad(\mathcal{N}_a) \leftarrow \min(min\_grad(\mathcal{N}_a), ima\_grad(e))$

            $\mathcal{N}_a \leftarrow parent(\mathcal{N}_a)$

        end

    end

    return *parent*

end

Algorithm 9: Computation of minimal information along the contour.

faces. For the 0-faces, it take the maximal value among the four 1-faces that share this 0-face in common.

9.5   Disjoint level lines selection

Each node of the topographic map represents a connected component without holes, the boundaries of the connected components are the level lines. It has been shown that significant contours of objects in images coincide with segments of the level lines [Caselles 1999]. In natural images, the number of level lines is in the same order of number of pixels, which make the topographic map difficult to be visualized. In fact, many level lines share some parts in common, which is to say that two neighboring level lines in the topographic map may only differ in a few pixel edges (in the case such that the level lines are materialized into pixel edges, i.e., 1-faces). In this section, we provide an efficient algorithm to select a set of disjoint level lines from the topographic map, such that any two selected level lines do not intersect. This algorithm yields a simplified image f' reconstructed from those selected level lines. The main structure of the topographic map of the original image f can be easily visualized from this simplified image f'. In fact, this simplified image is a well-composed image which is usually obtained by doubling the image size. In our case, the generated well-composed image f' has the same size as the original image f. The core of this algorithm will be detailed in Section 9.5.1. Then in Section 9.5.2, we will depict the algorithm and illustrate several rules of disjoint level lines selection.

9.5.1   Incompatible nodes preparation

The core of the algorithm for selecting a set of disjoint level lines relies on an image defined on the nodes that we call $last\_not\_allowed$, that encodes for each node $\mathcal{N}$ the highest ancestor node $\mathcal{N}_a$ for which they still share some pixel edges. When this image is available, for each level line $\partial \mathcal{N}$, we are able to predict a set of incompatible nodes of this node $\mathcal{N}$. In fact, if a node $\mathcal{N}$ is selected, we cannot select its ancestor nodes $\mathcal{N}_a$ till the one encoded by the image $last\_not\_allowed(\mathcal{N})$, and the descendant nodes $\mathcal{N}_d$ for which the selected node $\mathcal{N}$ is in the subbranch starting from $\mathcal{N}_d$ till the ancestor node given by $last\_not\_allowed(\mathcal{N}_d)$ cannot be selected either. Based on this principle, we are able to select a set of nodes such that any pair of selected nodes are not incompatible, i.e., they are disjoint level lines.

Now let us show how to compute the image $last\_not\_allowed$. To compute it efficiently, we need another image $depth$ storing the depth of each node on the tree (starting from 0 for the root node). For each node $\mathcal{N}$ apart from the root node:

$$depth(\mathcal{N}) \,=\, depth(parent(\mathcal{N})) + 1 \tag{9.2}$$

Explanation of variables:

pparent	image of parenthood of the second tree $\mathcal{TT}$
m	local minima node
n	node of $\mathcal{T}$
is_seen	image indicating whether a node is already propagated
original_min	image defined on nodes storing original minimum
extinct_value	image of extinction value

input : An input image f.
output: Saliency map.

COMPUTE_SALIENCY_MAP(f)
begin
  $(\mathcal{T}, \mathcal{A}) \leftarrow$ COMPUTE_TREE(f)
  $pparent \leftarrow$ COMPUTE_MIN_TREE($\mathcal{T}, \mathcal{A}$)
  for all $n \in \mathcal{T}$ **do** $is\_seen(n) \leftarrow$ false, $original\_min(n) \leftarrow$ undef,
  $extinct\_value(m) \leftarrow 0$
  for all e **do** $saliency\_map(e) \leftarrow 0$
  for all m **do**
    $original\_min(m) \leftarrow m$, $p \leftarrow pparent(m)$
    while $p \neq pparent(p)$ **do**
      if $!is\_seen(p)$ **then**
        $is\_seen(p) \leftarrow$ true, $original\_min(p) \leftarrow m$
      else
        if $\mathcal{A}(original\_min(p)) > \mathcal{A}(m)$ **then** $original\_min(p) \leftarrow m$
        else break
      end
    end
    if $p = pparent(p)$ **then**
      if $!is\_seen(p)$ **then** $is\_seen(p) \leftarrow$ true, $original\_min(p) \leftarrow m$
      else if $\mathcal{A}(original\_min(p)) > \mathcal{A}(m)$ **then** $original\_min(p) \leftarrow m$
    end
  end
  for all m **do**
    $p \leftarrow m$
    while $original\_min(pparent(p)) = m$ **and** $p \neq pparent(p)$ **do**
      $p \leftarrow pparent(p)$
    end
    $extinct\_value(m) \leftarrow \mathcal{A}(p) - \mathcal{A}(m)$
  end
  for all e **do**
    $\mathcal{N}_a \leftarrow appear(e), \mathcal{N}_v \leftarrow vanish(e)$
    while $\mathcal{N}_a \neq \mathcal{N}_v$ **do**
      $saliency\_map(\mathcal{N}_a) \leftarrow \max(extinct\_value(\mathcal{N}_a), saliency\_map(e))$
      $\mathcal{N}_a \leftarrow parent(\mathcal{N}_a)$
    end
  end
  return $saliency\_map$
end

Algorithm 10: Computation of saliency map.

And thanks to the image *appear* and *vanish* used in Section 9.3, the computation of the image *last_not_allowed* can be achieved with the following process: For each pixel edge e, let the node $a = appear(e)$ and $v = vanish(e)$ be respectively the first node where e is on its boundary and the first node where e is no longer on its boundary. Let v_c be the child of v being also an ancestor node of a. We can update the image *last_not_allowed* for the nodes on the sub-branch starting from a to v', which cannot be lower than v'. That is to say the *depth* of the nodes of the image *last_not_allowed* on a to v' cannot be smaller than $depth(v')$. The algorithm to compute this image *last_not_allowed* is depicted in Algorithm 11, where *vec_nodes* stands for the sorted set of nodes of the topographic map in the tree descending order, and $N_{\mathcal{N}}$ is the total number of nodes.

Explanation of variables:

vec_nodes	set of sorted nodes in tree descending order
$N_{\mathcal{N}}$	total number of nodes
n	node
depth	image of depth
last_not_allowed	image of the highest incompatible node

COMPUTE_LAST_NOT_ALLOWED(*parent*, *vec_nodes*)
begin
  for all n **do**
    | $depth(n) \leftarrow -1, last\_not\_allowed(n) \leftarrow n$
  end
  for $i \leftarrow 0$ **to** $N_{\mathcal{N}} - 1$ **do**
    | $n \leftarrow vec\_nodes[i]$
    | $depth(n) \leftarrow depth(parent(n)) + 1$
  end
  for all e **do**
    | $a \leftarrow appear(e), v \leftarrow vanish(e)$
    | **if** $a \neq v$ **then**
      | $n \leftarrow a$
      | **while** $parent(n) \neq v$ **do** $n \leftarrow parent(n)$
      | **while** $a \neq n$ **do**
        | **if** $depth(last\_not\_allowed(a)) > depth(n)$ **then**
        | | $last\_not\_allowed(a) \leftarrow n$
        | $a \leftarrow parent(a)$
      | **end**
    | **end**
  end
  return *last_not_allowed*
end

Algorithm 11: Computation of image *last_not_allowed*.

9.5.2   Final disjoint nodes selection

Once the image *last_not_allowed* is available, we are able to select a set of nodes such that any pair of nodes are compatible. Now let us detail how to process the choices according to a pre-computed order of nodes selection, such as top-down or bottom-up order. We propagate all the nodes in the pre-computed order, for each node $\mathcal{N}$, we perform two operations:

1) Check if there is an ancestor node being incompatible with $\mathcal{N}$ has already been selected. This verification is based on the image *last_not_allowed*.

2) If none of the incompatible ancestor nodes given by the image *last_not_allowed*($\mathcal{N}$) is selected, then select this node $\mathcal{N}$, and disable all its incompatible ancestor nodes. This step makes sure that we don't have to check the incompatibility with the descendants for a given node. Otherwise, do nothing.

Finally, the algorithm for the disjoint level lines selection is depicted in Algorithm 12. We have experienced three different orders for disjoint level lines selection: top-down selection order (from root node to the leaf nodes), bottom-up (from the leaf nodes to the root node), and meaningfulness decreasing order, e.g., average of gradient's magnitude along the level lines defined by Eq (3.8). Two examples of such disjoint level lines selection are illustrated in Figure 9.1 and Figure 9.2, where a grain filter [Caselles 2002] is also applied to not select level lines of regions being too small. The main structure of the topographic map can be easily visualized through the simplified image reconstructed from the set of disjoint level lines.

Explanation of variables:

is_checked	image indicating whether a node is already checked
vec_nodes	set of sorted nodes in a given order
is_selected	image of indicating whether a node is selected

SELECT_DISJOINT_LEVEL_LINES(*parent, last_not_allowed*)

begin

    for all n **do** $is\_checked(n) \leftarrow$ false, $is\_selected(n) \leftarrow$ ture

    for $i \leftarrow 0$ **to** $N_{\mathcal{N}} - 1$ **do**

        $n \leftarrow vec\_nodes[i]$

        if $is\_selected(n)$ **then**

            $nnot = last\_not\_allowed(n)$

            if $n \neq nnot$ **then**

                $nt \leftarrow parent(n)$

                while $nt \neq nnot$ **do**

                    if $is\_selected(nt)$ **and** $is\_checked(nt)$ **then**

                        $is\_selected(n) \leftarrow$ false

                        break

                    $nt \leftarrow parent(nt)$

                end

                if $is\_selected(nnot)$ **and** $is\_checked(nnot)$ **then**

                    $is\_selected(n) \leftarrow$ false

        if $is\_selected(n)$ **then**

            $nnot = last\_not\_allowed(n)$

            if $n \neq nnot$ **then**

                $nt \leftarrow parent(n)$

                while $nt \neq nnot$ **do**

                    $is\_selected(nt) \leftarrow$ false

                    $is\_checked(nt) \leftarrow$ true

                    $nt \leftarrow parent(nt)$

                end

                $is\_selected(nnot) \leftarrow$ false

                $is\_checked(nnot) \leftarrow$ true

        $is\_checked(n) \leftarrow$ true

    end

    return $is\_selected$

end

Algorithm 12: Disjoint level lines selection relying on the image *last_not_allowed*.
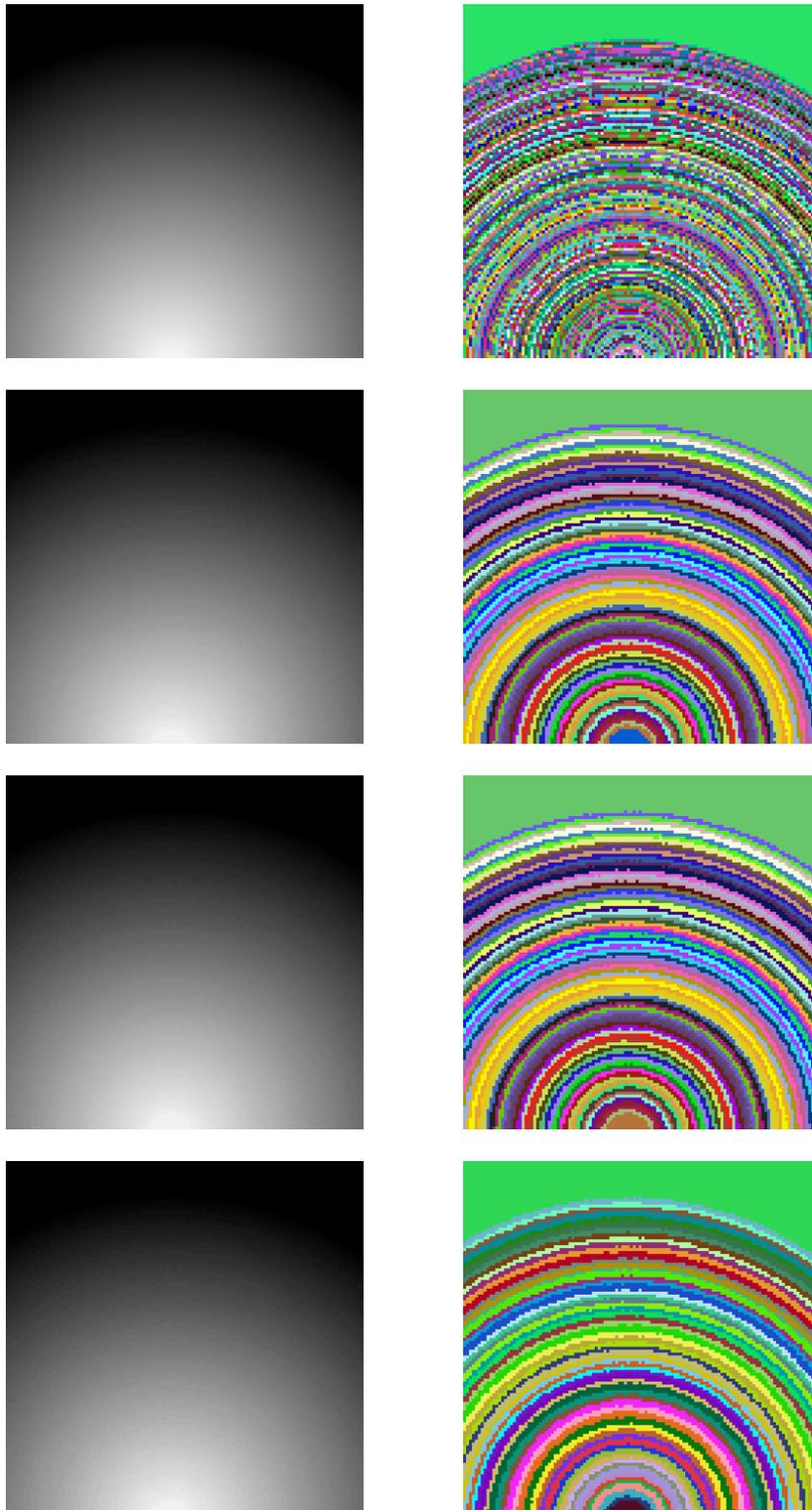
Figure 9.1: Illustration of the disjoint level lines selection with different orders. From top to down: original image, bottom-up, top-down, average of gradient's magnitude decreasing. Left: grayscale image; Right: corresponding randomly colorized image.
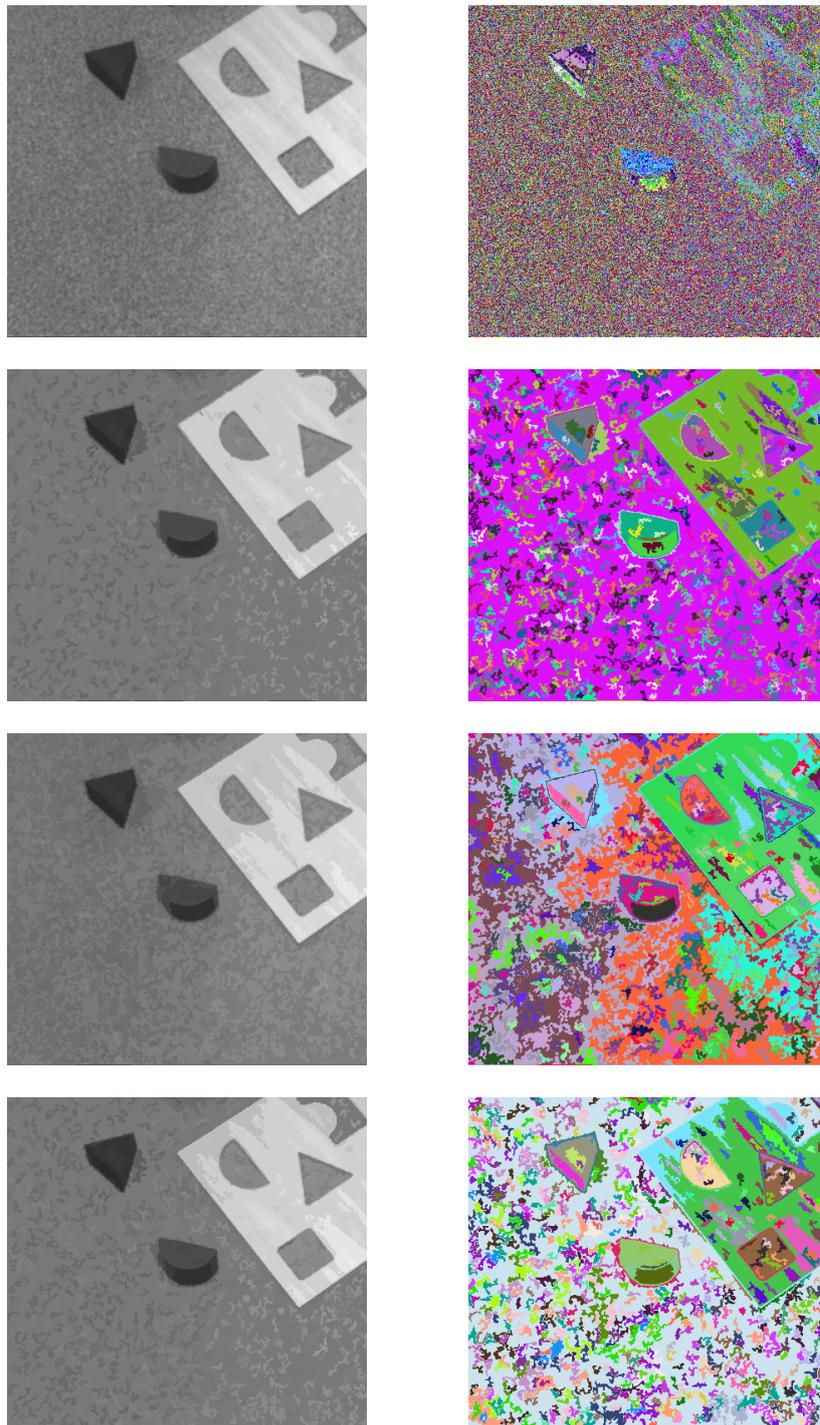
Figure 9.2: Another illustration of the disjoint level lines selection with different orders. From top to down: original image, bottom-up, top-down, average of gradient's magnitude decreasing. Left: grayscale image; Right: corresponding randomly colorized image.

Part V

Conclusion

Conclusion and perspectives

In this thesis, we have introduced the notion of tree-based shape space given by any tree-based image representation. The classical connected operators [Salembier 2009] can be seen as a limited analysis of this tree-based shape space. Many applications relying on the tree-based representations, such as meaningful level lines selection [Cao 2005] and MSER [Matas 2002], are equivalent to selecting relevant/representative points from the shape space. A variant of MSER, called TBMR, and presented in Chapter 4, selects some feature points in the tree-based shape space based on topological information. A main contribution of this thesis is to propose the versatile shape-based morphology framework (Chapter 3), which is summarized in Table 10.1. The applications of different aspects of this framework can be found in Part III. The main results of the work presented in this PhD thesis is summarized in the following section, and some perspectives are given in the last section.

10.1 Main results

The core contribution of our work is the framework of shape-based morphology presented in Chapter 3. It shares the same idea as the classical connected operators that one can rely on a tree-based image representation $\mathcal{T}$ to get interesting properties. We propose to expand on this idea by introducing the notion of tree-based shape space given by the first tree $\mathcal{T}$. This notion provides a new point of view of the classical connected operators and of the many applications in image processing and

Aspect	Applications	Details
Shape filtering	retinal image analysis and shapings	Sec. 3.4 and Chap. 6
Object detection	object detection/segmentation	Sec. 3.5 and Chap. 7
Hierarchy transformation	extending constrained connectivity	Sec. 3.6 and Chap. 8

Table 10.1: Three main aspects of the proposed framework of shape-based morphology, and some corresponding applications developed in the work presented in this thesis.

computer vision that makes use of the trees. A first consequence of this change of point of view is our proposition of the TBMR method, presented in Chapter 4, and which is a variant of the MSER method. The principle of the framework of shape-based morphology is to apply connected operators on the shape space through a second tree $\mathcal{TT}$, built from the shape space given by $\mathcal{T}$. Different aspects of this framework are given in Table 10.1. Some contributions of this PhD thesis are also several applications of the framework of shape-based morphology.

Tree-based shape space (see Chapter 2) - The introduction of this notion provides us with a new point of view of classical operators, they do not take benefit from the whole structure of the shape space. Besides, many applications in image processing and computer vision relying on some tree representations can be seen as a selection of relevant or representative points or components from the shape space. In some cases, the selection decision is made individually, and in some cases, the local structure of the tree is used to guide the selection.

Tree-based Morse regions (TBMR) (see Chapter 4) - Inspired from the widely used MSER method [Matas 2002], which can be seen as relevant node selection from the Min/Max-trees, the TBMR approach proposes to select relevant nodes from the shape spaces based on topological information motivated by Morse theory. We have shown that the TBMR is truly contrast independent, quasi parameter free, and fast to compute with a linear or quasi-linear complexity. The repeatability test shows that it achieves state-of-the-art results. In addition, applications to homographic registration and 3D reconstruction demonstrate its accuracy and robustness, as compared to some other state-of-the-art detectors.

Hierarchical image simplification (see Chapter 5) - We have presented an efficient salient level lines selection method that strongly simplifies images, while preserving their meaningful structures. It is a fast variant of the approach proposed by Ballester *et al.* in [Ballester 2007]. We have suggested a fast greedy algorithm that minimizes the piecewise-constant Mumford-Shah functional [Mumford 1989] subordinated to the topographic map [Monasse 2000b]. In fact, our method is one of the many variants of morphological shapings defined in the framework of shape-based morphology. The applications of pre-segmentation of color images and of autophagosome counting in cellular images demonstrate its usefulness and robustness. Besides, we have shown that an attribute function derived from this method along with the aspect of hierarchy transformation in Section 3.6 yields a hierarchical version of image simplification represented by a saliency map. Most of the salient level lines can be obtained by thresholding this map.

Shape filtering (see Chapter 6) - The shape-based upper/lower levelings obtained by some filtering using the proposed framework are applied to retinal image

analysis. It includes blood vessel segmentation and optic nerve head segmentation. In both cases, quantitative assessments on public datasets have shown that such a "simple" filtering step achieves some results that are almost as good as those obtained by human observer. Compared to some other dedicated approaches, it gives similar results with a much more simple process. An interesting example of the morphological shapings has also been illustrated with an image of zeolite. It filters out most of the unwanted objects, while keeping intact the desired bright and dark objects.

Object detection/segmentation (see Chapter 7) - We proposed a new energy functional, well-suited to characterize object contours. It makes use of a regional context, which make the energy neither too local nor global. The aspect of object detection/segmentation of the proposed framework along with this new energy functional improves the relevance and robustness of object detection, as compared to several state-of-the-art methods. It has a quasi-linear time complexity. Besides, a soft object detection/segmentation result can be obtained via saliency map using the aspect of hierarchy transformation.

Extending constrained connectivity (see Chapter 8) - We have presented an extension of constrained connectivity [Soille 2008] to non-increasing constraints. It is represented by a saliency map achieved by the aspect of the hierarchy transformation in our framework. Instead of using the local range of the gray levels (the case of constrained connectivity), we make use of an attribute function inspired from the work of efficient graph-based image segmentation of [Felzenszwalb 2004]. The obtained saliency map was benchmarked on the BSDS500 dataset [Arbelaez 2011]. Qualitative results show that we can obtain a partition that is close to the ground truth by thresholding the produced saliency map. Besides, quantitative evaluation demonstrates also that this extension of constrained connectivity achieves state-of-the-art results.

Algorithms to compute information on the tree (see Chapter 9) - We further present efficient algorithms able to compute useful information used for applications using the trees: information on the region, contour, and context. We have also suggested an efficient algorithm to compute the minimal information along the contour. Furthermore, we have presented an efficient algorithm selecting disjoint level lines from the topographic map that can be used to visualize directly the main structure of a topographic map representation. It provides also an image simplification method that makes the image well-composed without duplicating the image size.

10.2   Future work

The work presented in this PhD thesis offers many avenues for further investigation.

10.2.1   Future directions of the developed applications

For each application presented in Part III and developed under the proposed framework of shape-based morphology, some improvements and future works are conceivable.

Local feature detection - The TBMR method presented in Chapter 4 is a variant of the widely used MSER; it achieves a comparable repeatability score, but a significant higher number of points. A first perspective is to filter the TBMRs using the topological persistence. We might then expect a higher repeatability score by sacrificing slightly the number of detected points. A second possibility to improve the repeatability score is to change the "scale" associated to each critical region (*i.e.*, critical point) in the definition of TBMR. For instance, instead of selecting the *largest* topologically equivalent one, we can chose the one being the most meaningful and topologically equivalent. Another perspective is to extend directly the definition of MSER by replacing the relative area variation in Eq (3.2) with another attribute function bearing the same invariant properties (for instance, Eq 3.8), followed by a morphological filtering applied on the tree-based shape space in order to remove the non-relevant nodes.

Hierarchical image simplification - The hierarchical image simplification presented in Chapter 5 is achieved by salient level lines selection. This selection is guided by minimizing the piecewise-constant Mumford-Shah functional subordinated to the topographic map. One possible application of this simplification method is image compression. A first future work is to prove that the local optimal solution subordinated to the topographic map is close to a local optimal solution of the functional applied directly to the space of image. The second perspective is to find how to post-process the pre-segmented color image given by this method to obtain an effective segmentation result. The same perspective applies also to obtain a hierarchical segmentation from the hierarchical image simplification. One possible application of this hierarchical image simplification is hierarchical feature learning, such as shown in the work of Farabet *et al.* [Farabet 2013]. Finally, the extension of this method to 3D images is also one of the future works. For instance, in the application of this method to autophagosome counting, the original image is actually a 3D image, whereas, in this thesis, we process it slice by slice.

Retinal image analysis - The application of the "simple" filtering to retinal image analysis presented in Chapter 6 achieves comparable results with those

of dedicated and much more complex methods. For the blood vessel segmentation, a first perspective is to preprocess the images with the method of path opening [Talbot 2007] to improve the presentation of the blood vessels in the in the tree-based shape space. A thinning process might be useful to retrieve the very thin structures that have been missed. Besides, a more sophisticated attribute function characterizing the vesselness is also worth further investigation. Another step will be to extend this method to blood vessel segmentation in 3D images. Concerning the optic nerve head (ONH) segmentation, one possible improvement is to use a deformable model instead of an elliptical model that fits the ONH, so that we can improve the precision in terms of ONH boundaries.

Object detection/segmentation - The scheme of object detection/segmentation presented in Chapter 7 employs a context-based energy estimator along with the topographic map representation. The context information is computed approximately. A perspective is to compute exactly and efficiently the information based on context. Another perspective of this work is to demonstrate the usefulness with other tree-based shape spaces, in which the objects are better presented. Finally, an extra future work is to extend this method to color images, and more generally, multi-valued images.

Extending constrained connectivity - Extending constrained connectivity, as presented in Chapter 8, is an application of the aspect of hierarchy transformation of the proposed framework. The underlying tree is the α-tree, and the attribute function is inspired by the work described in [Felzenszwalb 2004]. We have benchmarked this method and compared it to the original method of [Felzenszwalb 2004] and to one hierarchical version [Guimarães 2012]. One future work would be to benchmark this method with the hierarchy of the α-tree, *i.e.*, the attribute map defined on boundaries using the local range of gray levels as an attribute function. One perspective of this work is to experiment some other attribute functions (for instance, some statistical measure) than the one inspired from [Felzenszwalb 2004]. This is still an extension of the constrained connectivity [Soille 2008]. Another perspective is to change the underlying tree. For example, the binary partition tree (BPT) [Salembier 2000], and apply the same process of hierarchy transformation.

10.2.2 Future directions for the proposed framework

The main contribution of the framework of shape-based morphology is a more robust method for the tree-based shape space analysis. It makes use of the tree structure to make a decision instead of making a decision individually. There are three important factors to make this framework work better.

Tree-based shape space - The choice of a tree structure to generate the tree-

based shape space is application-driven. The different tree types are presented in Section 2.3. One main purpose is to produce a pertinent tree for color images, for instance, a topographic-like map. Then we can exploit different similarity measures (such as the statistical measures presented in [Calderero 2010]) between regions to build a region-merging-based tree.

First attribute function $\mathcal{A}$ - The analysis of the tree-based shape space is performed on the basis of the attribute function $\mathcal{A}$ characterizing each point of the shape space. One major perspective is to rely on learning to chose the more relevant attribute $\mathcal{A}$.

Second attribute function $\mathcal{A}\mathcal{A}$ - The decision of the shape space analysis is achieved by pruning a second tree representation based on $\mathcal{A}\mathcal{A}$. In the future, we would like to experiment some second attributes $\mathcal{A}\mathcal{A}$. One major perspective is to compute it from some contextual information of the original image domain.

The three aspects of the proposed framework of shape-based morphology are open issues worth further study.

Filtering - We would like to study some properties of the shape filtering given by the framework, such as the condition of being idempotent. Another perspective is to compare the levelings [Meyer 1998, Meyer 2004] and the morphological shapings.

Object detection/segmentation - After filtering in the shape space to remove spurious minima, remaining minima are presented in terms of flat zones in the shape space. Instead of choosing the node having the smallest attribute as the representative for each flat zone, we might choose the node being the largest one in size. Another possibility is to apply a linear filter in the shape space instead of a connected filter. Finally, we would like to experiment some applications using the soft object detection via saliency map.

Hierarchy transformation - The saliency map obtained under the hierarchy transformation is based on the extinction values defined on the set of local minima. A major perspective is to develop more sophisticated method to post-process this saliency map, so that it represents a better hierarchy of segmentation. Another perspective is to compute the saliency map using the topological persistence.

We further consider to develop more applications of the framework of shape-based morphology.

Feature extraction on point-sampled surfaces - A minimum spanning tree (MST) is used in the application of feature extraction on point-sampled surfaces [Pauly 2003]. The MST is constructed on a set of selected feature nodes among a set of given points. The short branches of this MST correspond to artefacts that do not describe salient features. In [Pauly 2003] the authors proposed to prune these branches based on the branch importance, given as the length of the branch

multiplied by the product of all edge weights. The filtering aspect of the framework of shape-based morphology might give a more robust filtering result.

Specific structure detection in document image analysis - Some specific structures might be useful for document image analysis, such as the separation lines, and background boxes. Those structures are usually present in tree representations. The aspect of object detection/segmentation might be interesting in this case. Meanwhile, the relevant object detection/segmentation provides a simplified tree representation, the structure of the simplified tree might be useful for optical character recognition.

Saliency detection - One possible application of this framework is to apply the attribute map defined on regions to saliency detection [Itti 1998]. The better localization of salient regions might improve the result of saliency detection.

Publication list

Some of the materials presented in this manuscript have been published in peer-reviewed conferences.

- **Salient Level Lines Selection Using the Mumford-Shah Functional**
  Y. Xu, T. Géraud and L. Najman
  International Conference on Image Processing 2013

- **Two Applications of Shape-Based Morphology: Blood Vessels Segmentation and a Generalization of Constrained Connectivity**
  Y. Xu, T. Géraud and L. Najman
  International Symposium on Mathematical Morphology 2013

- **Morphological Filtering in Shape Spaces: Applications Using Tree-Based Image Representations**
  Y. Xu, T. Géraud and L. Najman
  International Conference on Pattern Recognition 2012

- **Context-Based Energy Estimator : Application to Object Segmentation on the Tree of Shapes**
  Y. Xu, T. Géraud and L. Najman
  International Conference on Image Processing 2012

Some work has been already submitted to journals.

- **Tree-Based Morse Regions: a Topological Approach to Local Feature Detection**
  Y. Xu, P. Monasse, T. Géraud and L. Najman
  Submitted to international journal 2014

- **Shape Filtering**
  Y. Xu, T. Géraud and L. Najman
  Submitted to international journal 2014

Some work presented in this thesis are in preparation for journal papers.

List of Figures

List of Tables

List of Algorithms

Index

Bibliography

[Aanæs 2012] Henrik Aanæs, Anders Lindbjerg Dahl and Kim Steenstrup Peder-
   sen. *Interesting Interest Points*. International Journal of Computer Vision,
   vol. 97, no. 1, pages 18–35, 2012. (Cited on pages 91, 94, 95 and 112.)

[Al-Diri 2009] Bashir Al-Diri, Andrew Hunter and David Steel. *An Active Contour
   Model for Segmenting and Measuring Retinal Vessels*. IEEE Transactions on
   Medical Imaging, vol. 28, no. 9, pages 1488–1497, 2009. (Cited on pages 149
   and 157.)

[Arbeláez 2006a] Pablo Arbeláez. *Boundary Extraction in Natural Images Using Ul-
   trametric Contour Maps*. In Proceedings of the 2006 Conference on Computer
   Vision and Pattern Recognition Workshop, pages 182–189, Washington, DC,
   USA, 2006. IEEE Computer Society. (Cited on pages 4 and 36.)

[Arbeláez 2006b] Pablo Arbeláez and Laurent D Cohen. *A Metric Approach to
   Vector-Valued Image Segmentation*. International Journal of Computer Vi-
   sion, vol. 69, no. 1, pages 119–126, 2006. (Cited on page 182.)

[Arbelaez 2011] Pablo Arbelaez, Michael Maire, Charless Fowlkes and Jitendra Ma-
   lik. *Contour Detection and Hierarchical Image Segmentation*. IEEE Trans-
   actions on Pattern Analysis and Machine Intelligence, vol. 33, no. 5, pages
   898–916, 2011. (Cited on pages xvi, 4, 7, 36, 132, 137, 139, 181, 182, 183,
   190, 192, 225 and 239.)

[Babaud 1986] Jean Babaud, Andrew P. Witkin, Michel Baudin and Richard O.
   Duda. *Uniqueness of the Gaussian Kernel for Scale-Space Filtering*. IEEE
   Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 1,
   pages 26–33, 1986. (Cited on page 34.)

[Bade 2007] A Bade and JJ Pizzimenti. *Interdisciplinary Management of Diabetic
   Eye Disease: a Global Approach to Care*. The Internet Journal of Allied
   Health Sciences and Practice, vol. 5, no. 1, pages 1–11, 2007. (Cited on
   page 147.)

[Ballester 2007] Coloma Ballester, Vicent Caselles, Laura Igual and Luis Garrido.
   *Level Lines Selection with Variational Models for Segmentation and Encod-
   ing*. Journal of Mathematical Imaging and Vision, vol. 27, pages 5–27, 2007.
   (Cited on pages 7, 49, 118, 119, 124, 126, 127, 129, 130, 132, 166, 177
   and 224.)

[Bangham 1996] J Andrew Bangham, Paul Ling and Richard Harvey. *Scale-Space from Nonlinear Filters*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, no. 5, pages 520–528, 1996. (Cited on page 41.)

[Benzécri 1973] Jean-Paul Benzécri. L'analyse des données: la taxinomie, volume 1. Dunod, 1973. (Cited on page 36.)

[Berger 2007] Christophe Berger, Thierry Géraud, Roland Levillain, Nicolas Widynski, Anthony Baillard and Emmanuel Bertin. *Effective Component Tree Computation with Application to Pattern Recognition in Astronomical Imaging*. In IEEE International Conference on Image Processing, volume 4, pages 41–44, 2007. (Cited on pages 25, 27, 173 and 201.)

[Bertrand 2005] Gilles Bertrand. *On Topological Watersheds*. Journal of Mathematical Imaging and Vision, vol. 22, no. 2-3, pages 217–230, may 2005. (Cited on page 36.)

[Beucher 1992] Serge Beucher and Fernand Meyer. *The Morphological Approach to Segmentation: the Watershed Transformation*. In Dougherty, editeur, Mathematical Morphology in Image Processing, Optical Engineering, chapter 12, pages 433–481. Marcel Dekker, 1992. (Cited on page 5.)

[Bloch 1996] Isabelle Bloch. *Information Combination Operators for Data Fusion: A Comparative Review with Classification*. IEEE Transactions on Systems, Man, and Cybernetics, Part A, vol. 26, no. 1, pages 52–67, 1996. (Cited on page 152.)

[Braga-Neto 2003] Ulisses Braga-Neto and John Goutsias. *A Theoretical Tour of Connectivity in Image Processing and Analysis*. Journal of Mathematical Imaging and Vision, vol. 19, no. 1, pages 5–31, 2003. (Cited on page 17.)

[Breen 1996] Edmond J Breen and Ronald Jones. *Attribute Openings, Thinnings, and Granulometries*. Computer Vision, Graphics and Image Understanding, vol. 64, no. 3, pages 377–389, 1996. (Cited on pages 21, 22, 43, 44, 45, 46, 53, 127 and 145.)

[Calderero 2010] Felipe Calderero and Ferran Marques. *Region Merging Techniques Using Information Theory Statistical Measures*. IEEE Transactions on Image Processing, vol. 19, no. 6, pages 1567–1586, 2010. (Cited on page 228.)

[Cao 2005] Frédéric Cao, Pablo Musé and Frédéric Sur. *Extracting Meaningful Curves from Images*. Journal of Mathematical Imaging and Vision, vol. 22,

no. 2-3, pages 159–181, 2005. (Cited on pages 7, 11, 14, 48, 49, 64, 77, 117, 166, 167, 176 and 223.)

[Cardelino 2006] Juan Cardelino, Gregory Randall, Marcelo Bertalmio and Vicent Caselles. *Region Based Segmentation Using the Tree of Shapes*. In IEEE International Conference on Image Processing, pages 2421–2424, 2006. (Cited on pages 117 and 166.)

[Cardelino 2013] Juan Cardelino, Vicent Caselles, Marcelo Bertalmío and Gregory Randall. *A Contrario Selection of Optimal Partitions for Image Segmentation*. SIAM J. Imaging Sciences (SIIMS), 2013. (Cited on pages 35 and 50.)

[Carlinet 2013] Edwin Carlinet and Thierry Géraud. *A Comparison of Many Max-tree Computation Algorithms*. In Internal Symposium on Mathematical Morphology, pages 73–85, 2013. (Cited on pages 26 and 202.)

[Carmona 2008] Enrique J Carmona, Mariano Rincón, Julián García-Feijoó and José M Martínez-de-la Casa. *Identification of the Optic Nerve Head with Genetic Algorithms*. Artificial Intelligence in Medicine, vol. 43, no. 3, pages 243–259, 2008. (Cited on pages xv, 146, 150, 152, 159 and 161.)

[Caselles 1997] Vicent Caselles, Ron Kimmel and Guillermo Sapiro. *Geodesic Active Contours*. International Journal of Computer Vision, vol. 22, no. 1, pages 61–79, 1997. (Cited on page 167.)

[Caselles 1999] Vicent Caselles, Bartomeu Coll and Jean-Michel Morel. *Topographic Maps and Local Contrast Changes in Natural Images*. International Journal of Computer Vision, vol. 33, no. 1, pages 5–27, 1999. (Cited on pages 5, 24, 25, 26, 117, 125, 166 and 213.)

[Caselles 2002] Vicent Caselles and Pascal Monasse. *Grain Filters*. Journal of Mathematical Imaging and Vision, vol. 17, no. 3, pages 249–270, 2002. (Cited on pages 20, 48, 216 and 233.)

[Caselles 2009] Vicent Caselles and Pascal Monasse. Geometric description of images as topographic maps. Springer Publishing Company, Incorporated, 1st édition, 2009. (Cited on pages xi, 25, 93, 96 and 202.)

[Chan 2001] Tony F Chan and Luminita A Vese. *Active Contours Without Edges*. IEEE Transactions on Image Processing, vol. 10, no. 2, pages 266–277, 2001. (Cited on pages 118, 120, 166, 167, 168, 171 and 176.)

[Chandrasekhar 2011] Vijay R. Chandrasekhar, David M. Chen, Sam S. Tsai, Ngai-Man Cheung, Huizhong Chen, Gabriel Takacs, Yuriy Reznik, Ramakrishna Vedantham, Radek Grzeszczuk, Jeff Bach and Bernd Girod. *The Stanford Mobile Visual Search Data Set*. In MMSYS, pages 117–122, 2011. (Cited on pages 98 and 105.)

[Chaudhuri 1989] Subhasis Chaudhuri, Shankar Chatterjee, Norman Katz, Mark Nelson and Michael Goldbaum. *Detection of Blood Vessels in Retinal Images Using Two-Dimensional Matched Filters*. IEEE Transactions on Medical Imaging, vol. 8, no. 3, pages 263–269, 1989. (Cited on page 148.)

[Chen 1989] M-H Chen and Ping-Fan Yan. *A Multiscanning Approach Based on Morphological Filtering*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 7, pages 694–700, 1989. (Cited on page 41.)

[Cheng 1995] Yizong Cheng. *Mean Shift, Mode Seeking, and Clustering*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 8, pages 790–799, 1995. (Cited on page 52.)

[Ciomaga 2010] Adina Ciomaga, Pascal Monasse and Jean-Michel Morel. *Level Lines Shortening Yields an Image Curvature Microscope*. In IEEE International Conference on Image Processing, pages 4129–4132, 2010. (Cited on page 170.)

[Ciulla 2003] Thomas A Ciulla, Armando G Amador and Bernard Zinman. *Diabetic Retinopathy and Diabetic Macular Edema Pathophysiology, Screening, and Novel Therapies*. Diabetes care, vol. 26, no. 9, pages 2653–2664, 2003. (Cited on pages 147 and 149.)

[Comaniciu 2002] Dorin Comaniciu and Peter Meer. *Mean Shift: A Robust Approach Toward Feature Space Analysis*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 5, pages 603–619, 2002. (Cited on page 52.)

[Couprie 1997] Michel Couprie and Gilles Bertrand. *Topological Grayscale Watershed Transformation*. In Proceedings of the SPIE Conference on Vision Geometry V, pages 136–146, 1997. (Cited on page 36.)

[Couprie 2005] Michel Couprie, Laurent Najman and Gilles Bertrand. *Quasi-Linear Algorithms for the Topological Watershed*. Journal of Mathematical Imaging and Vision, vol. 22, no. 2-3, pages 231–249, may 2005. (Cited on page 36.)

[Cousty 2009] Jean Cousty, Gilles Bertrand, Laurent Najman and Michel Couprie. *Watershed Cuts: Minimum Spanning Forests and the Drop of Water Principle*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 8, pages 1362–1374, 2009. (Cited on page 32.)

[Cousty 2013] Jean Cousty, Laurent Najman and Benjamin Perret. *Constructive Links Between Some Morphological Hierarchies on Edge-Weighted Graphs*. In Internal Symposium on Mathematical Morphology, pages 86–97. Springer, 2013. (Cited on page 32.)

[Cremers 2002] Daniel Cremers, Florian Tischhäuser, Joachim Weickert and Christoph Schnörr. *Diffusion Snakes: Introducing Statistical Shape Knowledge into the Mumford-Shah Functional*. International Journal of Computer Vision, vol. 50, no. 3, pages 295–313, 2002. (Cited on page 120.)

[Desolneux 2000] Agnès Desolneux, Lionel Moisan and Jean-Michel Morel. *Meaningful Alignments*. International Journal of Computer Vision, vol. 40, no. 1, pages 7–23, 2000. (Cited on pages 105 and 110.)

[Desolneux 2001] Agnès Desolneux, Lionel Moisan and Jean-Michel Morel. *Edge Detection by Helmholtz Principle*. Journal of Mathematical Imaging and Vision, vol. 14, no. 3, pages 271–284, 2001. (Cited on pages 11, 14, 48, 49 and 166.)

[Desolneux 2008] Agnès Desolneux, Lionel Moisan and Jean-Michel Morel. *From gestalt theory to image analysis: a probabilistic approach*, volume 34. Springer, 2008. (Cited on page 49.)

[Donoser 2006] Michael Donoser and Horst Bischof. *Efficient Maximally Stable Extremal Region (MSER) Tracking*. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 553–560, 2006. (Cited on pages 52, 66 and 94.)

[dri 2013a] *DRIONS-DB: Digital Retinal Images for Optic Nerve Segmentation Database*. http://www.ia.uned.es/~ejcarmona/DRIONS-DB.html, 2013. (Cited on pages 146, 149, 159 and 239.)

[dri 2013b] *DRIVE: Digital Retinal Images for Vessel Extraction*. http://www.isi.uu.nl/Research/Databases/DRIVE/, 2013. (Cited on pages 146, 148, 154 and 239.)

[Edelsbrunner 2000] Herbert Edelsbrunner, David Letscher and Afra Zomorodian. *Topological Persistence and Simplification.* In Foundations of Computer Science, Proceedings. 41st Annual Symposium on, pages 454–463, 2000. (Cited on page 52.)

[El-Zehiry 2011] Noha El-Zehiry, Prasanna Sahoo and Adel Elmaghraby. *Combinatorial Optimization of the Piecewise Constant Mumford-Shah Functional with Application to Scalar/Vector Valued and Volumetric Image Segmentation.* Image and Vision Computing, vol. 29, no. 6, pages 365–381, 2011. (Cited on page 120.)

[El-Zehiry 2013] NohaYoussry El-Zehiry and Leo Grady. *Combinatorial Optimization of the Discretized Multiphase Mumford-Shah Functional.* International Journal of Computer Vision, vol. 104, no. 3, pages 270–285, 2013. (Cited on page 120.)

[Epifanio 2007] Irene Epifanio and Pierre Soille. *Morphological Texture Features for Unsupervised and Supervised Segmentations of Natural Landscapes.* IEEE Transactions on Geoscience and Remote Sensing, vol. 45, no. 4, pages 1074–1083, 2007. (Cited on page 54.)

[Farabet 2013] Clément Farabet, Camille Couprie, Laurent Najman and Yann LeCun. *Learning Hierarchical Features for Scene Labeling.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013. (Cited on pages 7, 84 and 226.)

[Felzenszwalb 2004] P. Felzenszwalb and D. P. Huttenlocher. *Efficient Graph-Based Image Segmentation.* International Journal of Computer Vision, vol. 59, no. 2, pages 167–181, sep 2004. (Cited on pages xvi, xvii, 11, 32, 49, 184, 186, 187, 189, 190, 197, 225, 227 and 241.)

[Finkel 1974] Raphael A. Finkel and Jon Louis Bentley. *Quad Trees: A Data Structure for Retrieval on Composite Keys.* Acta Informatica, vol. 4, pages 1–9, 1974. (Cited on pages 4, 7, 24 and 30.)

[Forssen 2007] P-E Forssen and David G Lowe. *Shape Descriptors for Maximally Stable Extremal Regions.* In IEEE International Conference on Computer Vision, pages 1–8, 2007. (Cited on page 67.)

[Furukawa 2010] Yasutaka Furukawa and Jean Ponce. *Accurate, Dense, and Robust Multiview Stereopsis.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 8, pages 1362–1376, 2010. (Cited on pages 92, 112 and 236.)

[Garrido 1998] Luis Garrido, Philippe Salembier and David Garcia. *Extensive Operators in Partition Lattices for Image Sequence Analysis.* Signal Processing, vol. 66, no. 2, pages 157–180, 1998. (Cited on page 28.)

[Gatica-Perez 2001] Daniel Gatica-Perez, Chuang Gu, Ming-Ting Sun and Salvador Ruiz-Correa. *Extensive Partition Operators, Gray-Level Connected Operators, and Region Merging/Classification Segmentation Algorithms: Theoretical Links.* IEEE Transactions on Image Processing, vol. 10, no. 9, pages 1332–1345, September 2001. (Cited on page 19.)

[Géraud 2013] Thierry Géraud, Edwin Carlinet, Sébastien Crozet and Laurent Najman. *A Quasi-linear Algorithm to Compute the Tree of Shapes of nD Images.* In Internal Symposium on Mathematical Morphology, pages 98–110, 2013. (Cited on pages 27, 173 and 202.)

[Grady 2009] Leo Grady and Christopher V Alvino. *The Piecewise Smooth Mumford–Shah Functional on an Arbitrary Graph.* IEEE Transactions on Image Processing, vol. 18, no. 11, pages 2547–2561, 2009. (Cited on page 120.)

[Guigues 2006] Laurent Guigues, Jean Pierre Cocquerez and Hervé Le Men. *Scale-Sets Image Analysis.* International Journal of Computer Vision, vol. 68, no. 3, pages 289–317, 2006. (Cited on pages x, xiii, xv, 7, 34, 35, 36, 38, 50, 118, 121, 122, 123, 124, 181, 182, 183 and 234.)

[Guimarães 2012] Silvio Jamil Ferzoli Guimarães, Jean Cousty, Yukiko Kenmochi and Laurent Najman. *An Efficient Hierarchical Graph Based Image Segmentation.* In 14th International Workshop on Structural and Syntactic Pattern Recognition, Hiroshima, Japan, 2012. (Cited on pages xvii, 187, 189, 197 and 227.)

[Harris 1988] Chris Harris and Mike Stephens. *A Combined Corner and Edge Detector.* In Proc. of Fourth Alvey Vision Conference, pages 147–151, 1988. (Cited on pages 66 and 94.)

[Harvey 1997] Richard Harvey, J Andrew Bangham and Alison Bosson. *Scale-Space Filters and Their Robustness.* In Scale-space theory in computer vision, pages 341–344. Springer, 1997. (Cited on page 41.)

[Haxhimusa 2004] Yll Haxhimusa and Walter Kropatsch. *Segmentation Graph Hierarchies.* In Structural, Syntactic, and Statistical Pattern Recognition, pages 343–351. Springer, 2004. (Cited on page 187.)

[Heijmans 1999] Henk JAM Heijmans. *Connected Morphological Operators for Binary Images*. Computer Vision, Graphics and Image Understanding, vol. 73, no. 1, pages 99–120, 1999. (Cited on page 21.)

[Hoffmann 2007] Esther M Hoffmann, Linda M Zangwill, Jonathan G Crowston and Robert N Weinreb. *Optic Disk Size and Glaucoma*. Survey of Ophthalmology, vol. 52, no. 1, pages 32–49, 2007. (Cited on page 149.)

[Hoover 2000] Adam Hoover, Valentina Kouznetsova and Michael Goldbaum. *Locating Blood Vessels in Retinal Images by Piece-wise Threshold Probing of a Matched Filter Response*. IEEE Transactions on Medical Imaging, vol. 19, pages 203–210, 2000. (Cited on pages 146, 148, 154, 156, 157 and 158.)

[Itti 1998] Laurent Itti, Christof Koch and Ernst Niebur. *A Model of Saliency-Based Visual Attention for Rapid Scene Analysis*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 11, pages 1254–1259, 1998. (Cited on page 229.)

[Jackway 1996] Paul T. Jackway and Mohamed Deriche. *Scale-Space Properties of the Multiscale Morphological Dilation-Erosion*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, no. 1, pages 38–51, 1996. (Cited on page 41.)

[Jackway 1998] Paul T Jackway. *On the Scale-Space Theorem of Chen and Yan*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 3, pages 351–352, 1998. (Cited on page 41.)

[Jiang 2003] Xiaoyi Jiang and Daniel Mojon. *Adaptive Local Thresholding by Verification-Based Multithreshold Probing with Application to Vessel Detection in Retinal Images*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 1, pages 131–137, 2003. (Cited on pages 148, 156, 157 and 158.)

[Johnson 1967] Stephen Johnson. *Hierarchical Clustering Schemes*. Psychometrika, vol. 32, no. 3, pages 241–254, sep 1967. (Cited on page 36.)

[Jolion 1994] Jean-Michel Jolion and Azriel Rosenfeld. A pyramid framework for early vision: Multiresolutional computer vision. Kluwer Academic Publishers, Norwell, MA, USA, 1994. (Cited on pages 4 and 34.)

[Jones 1999] Ronald Jones. *Connected Filtering and Segmentation Using Component Trees*. Computer Vision, Graphics and Image Understanding, vol. 75, no. 3, pages 215–228, September 1999. (Cited on page 19.)

[Kass 1988] Michael Kass, Andrew Witkin and Demetri Terzopoulos. *Snakes : Active Contour Models*. International Journal of Computer Vision, vol. 1, pages 321–331, 1988. (Cited on pages 79, 166, 167, 168, 176 and 235.)

[Kauppi 2007] Tomi Kauppi, Valentina Kalesnykiene, Joni-Kristian Kamarainen, Lasse Lensu, Iiris Sorri, Asta Raninen, Raija Voutilainen, Hannu Uusitalo, Heikki Kälviäinen and Juhani Pietilä. *The DIARETDB1 Diabetic Retinopathy Database and Evaluation Protocol*. In British Machine Vision Conference, pages 1–10, 2007. (Cited on page 147.)

[Kiran 2013a] Bangalore Ravi Kiran and Jean Serra. *Ground Truth Energies for Hierarchies of Segmentations*. In Mathematical Morphology and Its Applications to Signal and Image Processing, pages 123–134. Springer, 2013. (Cited on page 35.)

[Kiran 2013b] Bangalore Ravi Kiran and Jean Serra. *Scale Space Operators on Hierarchies of Segmentations*. In Scale Space and Variational Methods in Computer Vision, pages 331–342. Springer, 2013. (Cited on page 35.)

[Kiran 2014] Bangalore Ravi Kiran and Jean Serra. *Global-Local Optimizations by Hierarchical Cuts and Climbing Energies*. Pattern Recognition, vol. 47, no. 1, pages 12–24, 2014. (Cited on pages 35, 50 and 183.)

[Koenderink 1984] Jan J Koenderink. *The structure of images*. Biological Cybernetics, vol. 50, no. 5, pages 363–370, 1984. (Cited on pages 8, 34, 41, 94 and 121.)

[Krasner 1944] Marc Krasner. *Espaces ultramétriques*. C.R. Acad. Sci., Paris, France, 1944. (Cited on page 36.)

[Kruskal 1956] Joseph B Kruskal. *On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem*. In Proc. of the American Mathematical Society, volume 7, pages 48–50, 1956. (Cited on pages x, 4, 7, 24, 30, 32 and 185.)

[Kweon 1994] In So Kweon and Takeo Kanade. *Extracting Topographic Terrain Features from Elevation Maps*. Journal of Computer Vision, Graphics and Image Processing: Image Understanding, vol. 59, no. 2, pages 171–182, 1994. (Cited on page 96.)

[Laliberté 2003] France Laliberté, Langis Gagnon and Yunlong Sheng. *Registration and Fusion of Retinal Images-an Evaluation Study*. IEEE Transactions on Medical Imaging, vol. 22, no. 5, pages 661–673, 2003. (Cited on page 147.)

[Levillain 2009]   Roland Levillain, Thierry Géraud and Laurent Najman. *Milena: Write Generic Morphological Algorithms Once, Run on Many Kinds of Images*. In Mathematical Morphology and Its Application to Signal and Image Processing, pages 295–306. Springer, 2009. (Cited on page 201.)

[Levillain 2010]   Roland Levillain, Thierry Géraud and Laurent Najman. *Why and How to Design a Generic and Efficient Image Processing Framework: The Case of the Milena Library*. In IEEE International Conference on Image Processing, pages 1941–1944, http://olena.lrde.epita.fr , 2010. (Cited on pages 14, 177 and 201.)

[Levillain 2012a] Roland Levillain, Thierry Géraud and Laurent Najman. *Writing Reusable Digital Geometry Algorithms in a Generic Image Processing Framework*. In U Köthe, A. Montanvert and P. Soille, editeurs, WADGMM 2010, volume 7346/2012 of *Lecture Notes in Computer Science*, pages 140–153, Istanbul, Turkey, August 2012. Springer. (Cited on page 201.)

[Levillain 2012b] Roland Levillain, Thierry Géraud and Laurent Najman. *Writing Reusable Digital Topology Algorithms in a Generic Image Processing Framework*. In Applications of Discrete Geometry and Mathematical Morphology, pages 140–153. Springer, 2012. (Cited on page 201.)

[Lindeberg 1994]   Tony Lindeberg. Scale-space theory in computer vision. Kluwer Academic Publishers, Norwell, MA, USA, 1994. (Cited on pages 40 and 41.)

[Lindeberg 1998]   Tony Lindeberg. *Feature Detection with Automatic Scale Selection*. International Journal of Computer Vision, vol. 30, pages 79–116, 1998. (Cited on pages 91 and 94.)

[Lisani 2003]   J.-L Lisani, Lionel Moisan, Pascal Monasse and Jean-Michel Morel. *On the Theory of Planar Shape*. Multiscale Modeling & Simulation, vol. 1, no. 1, pages 1–24, 2003. (Cited on pages 50 and 144.)

[Lowe 1999]   David Lowe. *Object Recognition from Local Scale-Invariant Features*. In IEEE International Conference on Computer Vision, pages 1150–1157, 1999. (Cited on pages 67, 93 and 94.)

[Lowe 2004]   David Lowe. *Distinctive Image Features from Scale-Invariant Keypoints*. International Journal of Computer Vision, vol. 60, pages 91–110, 2004. (Cited on pages 67, 91, 105 and 110.)

[Lowell 2004]   James Lowell, Andrew Hunter, David Steel, Ansu Basu, Robert Ryder, Eric Fletcher and Lee Kennedy. *Optic Nerve Head Segmentation*. IEEE

Transactions on Medical Imaging, vol. 23, no. 2, pages 256–264, 2004. (Cited on pages 149, 152, 159 and 161.)

[Lu 2007] Huihai Lu, John C Woods and Mohammed Ghanbari. *Binary Partition Tree Analysis Based on Region Evolution and its Application to Tree Simplification.* IEEE Transactions on Image Processing, vol. 16, no. 4, pages 1131–1138, 2007. (Cited on pages 49, 117 and 127.)

[Luo 2009] Bin Luo, Jean-François Aujol and Yann Gousseau. *Local Scale Measure from the Topographic Map and Application to Remote Sensing Images.* Multiscale modeling & simulation, vol. 8, no. 1, pages 1–29, 2009. (Cited on page 84.)

[Luo 2013] Bin Luo and Liangpei Zhang. *Robust Autodual Morphological Profiles for the Classification of High-Resolution Satellite Images.* IEEE Transactions on Geoscience and Remote Sensing, vol. 1, no. 1, pages 1–12, 2013. (Cited on page 54.)

[Ma 2000] Bing Ma, Alfred Hero, John Gomtan and Olivier Michel. *Image Registration with Minimum Spanning Tree Algorithm.* In IEEE International Conference on Image Processing, volume 1, pages 481–484, 2000. (Cited on page 32.)

[Malisiewicz 2007] Tomasz Malisiewicz and Alexei A Efros. *Improving Spatial Support for Objects via Multiple Segmentations.* In British Machine Vision Conference, pages 1–10, 2007. (Cited on page 192.)

[Maragos 1989] Petros Maragos. *Pattern Spectrum and Multiscale Shape Representation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 7, pages 701–716, 1989. (Cited on page 53.)

[Marfil 2006] Rebeca Marfil, Luis Molina-Tanco, Antonio Bandera, Juan Antonio Rodríguez and F Sandoval. *Pyramid Segmentation Algorithms Revisited.* Pattern Recognition, vol. 39, no. 8, pages 1430–1451, August 2006. (Cited on pages 35 and 182.)

[Marinez-Pérez 1999] M. Marinez-Pérez, A. Hughes, A. Stanton, S. Thom, A. Bharath and K. Parker. *Scale-Space Analysis for the Characterisation of Retinal Blood Vessels.* In C. Taylor and A. Colchester, editeurs, International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 90–97, 1999. (Cited on pages 148 and 157.)

[Markaki 2009] Vasiliki E Markaki, Pantelis A Asvestas and George K Matsopoulos. *Application of Kohonen Network for Automatic Point Correspondence in 2D Medical Images*. Computers in Biology and Medicine, vol. 39, no. 7, pages 630–645, 2009. (Cited on page 147.)

[Markas 1992] Tassos Markas and John Reif. *Quad Tree Structures for Image Compression Applications*. Inf. Process. Manage., vol. 28, no. 6, pages 707–721, November 1992. (Cited on pages 7 and 30.)

[Marr 1983] David Marr. Vision: A computational investigation into the human representation and processing of visual information. Henry Holt & Company, jun 1983. (Cited on pages 7 and 40.)

[Martin 2001] David Martin, Charless 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 IEEE International Conference on Computer Vision, volume 2, pages 416–423, July 2001. (Cited on pages 137 and 190.)

[Matas 2002] Jiri Matas, Ondrej Chum, Martin Urban and Tomás Pajdla. *Robust Wide Baseline Stereo from Maximally Stable Extremal Regions*. In British Machine Vision Conference, pages 384–393, 2002. (Cited on pages viii, xi, 7, 8, 10, 12, 51, 57, 58, 91, 94, 223 and 224.)

[Mendonca 2006] Ana Maria Mendonca and Aurelio Campilho. *Segmentation of Retinal Blood Vessels by Combining the Detection of Centerlines and Morphological Reconstruction*. IEEE Transactions on Medical Imaging, vol. 25, no. 9, pages 1200–1213, 2006. (Cited on pages 149, 156, 157 and 158.)

[Meyer 1998] Fernand Meyer. *From Connected Operators to Levelings*. In Internal Symposium on Mathematical Morphology, pages 191–198, Amsterdam, June 1998. The Netherlands: Kluwer. (Cited on pages 23, 60 and 228.)

[Meyer 2000] Fernand Meyer and Petros Maragos. *Nonlinear Scale-Space Representation with Morphological Levelings*. Journal of Visual Communication and Image Representation, vol. 11, no. 3, pages 245–265, 2000. (Cited on pages 4, 7, 28 and 41.)

[Meyer 2004] Fernand Meyer. *Levelings, Image Simplification Filters for Segmentation*. Journal of Mathematical Imaging and Vision, vol. 20, no. 1-2, pages 59–72, 2004. (Cited on pages 23, 60 and 228.)

[Mikolajczyk 2004] Krystian Mikolajczyk and Cordelia Schmid. *Scale & Affine Invariant Interest Point Detectors*. International Journal of Computer Vision, vol. 60, no. 1, pages 63–86, 2004. (Cited on pages 66 and 94.)

[Mikolajczyk 2005] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir and Luc Van Gool. *A Comparison of Affine Region Detectors*. International Journal of Computer Vision, vol. 65, no. 1-2, pages 43–72, November 2005. (Cited on pages viii, xi, xii, 52, 67, 91, 94, 97, 98, 101, 103, 115 and 237.)

[Milnor 1963] John Milnor. Morse Theory, volume 51 of *Annals of Mathematics Studies*. Princeton University Press, 1963. (Cited on pages xi, 10, 52, 71, 93 and 95.)

[Moisan 2004] Lionel Moisan and Bérenger Stival. *A Probabilistic Criterion to Detect Rigid Point Matches Between Two Images and Estimate the Fundamental Matrix*. International Journal of Computer Vision, vol. 57, no. 3, pages 201–218, 2004. (Cited on page 105.)

[Moisan 2012] Lionel Moisan, Pierre Moulon and Pascal Monasse. *Automatic Homographic Registration of a Pair of Images, with A Contrario Elimination of Outliers*. IPOL, 2012. `http://dx.doi.org/10.5201/ipol.2012.mmm-oh`. (Cited on page 105.)

[Molina 2011] José M Molina and Enrique J Carmona. *Localization and Segmentation of the Optic Nerve Head in Eye Fundus Images Using Pyramid Representation and Genetic Algorithms*. In Foundations on Natural and Artificial Computation, pages 431–440. Springer, 2011. (Cited on pages 150, 152 and 161.)

[Monasse 2000a] Pascal Monasse. *Morphological Representation of Digital Images and Application to Registration*. PhD thesis, Université Paris IX-Dauphine, Paris, France, June 2000. (Cited on page 27.)

[Monasse 2000b] Pascal Monasse and Frederic Guichard. *Fast Computation of a Contrast-Invariant Image Representation*. IEEE Transactions on Image Processing, vol. 9, no. 5, pages 860–872, 2000. (Cited on pages vii, ix, xvii, 5, 13, 24, 25, 26, 76, 137, 166, 173, 202 and 224.)

[Montero 2009] Raul S Montero and Ernesto Bribiesca. *State of the Art of Compactness and Circularity Measures*. In International Mathematical Forum, volume 4, pages 1305–1335, 2009. (Cited on page 44.)

[Moreels 2007] Pierre Moreels and Pietro Perona. *Evaluation of Features Detectors and Descriptors based on 3D Objects*. International Journal of Computer Vision, vol. 73, no. 3, pages 263–284, 2007. (Cited on pages 91 and 94.)

[Morel 1995] Jean Michel Morel and Sergio Solimini. Variational methods in image segmentation. Birkhauser Boston Inc., Cambridge, MA, USA, 1995. (Cited on pages 34 and 121.)

[Morris 1986] OJ Morris, M de J Lee and AG Constantinides. *Graph Theory for Image Analysis: an Approach Based on the Shortest Spanning Tree*. In IEE Proc. F, volume 133, pages 146–152, April 1986. (Cited on pages 24, 28 and 32.)

[Moulon 2012] Pierre Moulon, Pascal Monasse and Renaud Marlet. *Adaptive Structure from Motion with a contrario model estimations*. In Asian Conference on Computer Vision, 2012. (Cited on pages 92, 110 and 236.)

[Mumford 1989] David Mumford and Jayant Shah. *Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems*. Communications on Pure and Applied Mathematics, vol. 42, no. 5, pages 577–685, 1989. (Cited on pages 49, 68, 118, 119, 165 and 224.)

[Nagao 1979] Makoto Nagao, Takashi Matsuyama and Yoshio Ikeda. *Region Extraction and Shape Analysis in Aerial Photographs*. Computer Graphics and Image Processing, vol. 10, no. 3, pages 195–223, 1979. (Cited on pages 4, 7 and 28.)

[Najman 1994] Laurent Najman. *Morphologie Mathématique: de la Segmentation d'Images à l'Analyse Multivoque*. These, Université Paris Dauphine - Paris IX, Apr 1994. (Cited on page 35.)

[Najman 1996] Laurent Najman and Michel Schmitt. *Geodesic Saliency of Watershed Contours and Hierarchical Segmentation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, no. 12, pages 1163–1173, 1996. (Cited on pages x, xv, 4, 10, 24, 35, 49, 85 and 182.)

[Najman 2006] Laurent Najman and Michel Couprie. *Building the Component Tree in Quasi-Linear Time*. IEEE Transactions on Image Processing, vol. 15, no. 11, pages 3531–3539, 2006. (Cited on pages 25, 27, 52, 93 and 201.)

[Najman 2009] Laurent Najman. *Ultrametric Watersheds*. In Internal Symposium on Mathematical Morphology, pages 181–192, Berlin, Heidelberg, 2009. Springer-Verlag. (Cited on pages 4, 7, 24, 36 and 49.)

[Najman 2010] Laurent Najman and Hugues Talbot. Mathematical Morphology: From Theory to Applications. ISTE-Wiley, June 2010. (Cited on page 5.)

[Najman 2011] Laurent Najman. *On the Equivalence Between Hierarchical Segmentations and Ultrametric Watersheds*. Journal of Mathematical Imaging and Vision, vol. 40, pages 231–247, 2011. (Cited on pages x, xv, 4, 24, 29, 36, 38, 182, 184 and 234.)

[Najman 2013] Laurent Najman, Jean Cousty and Benjamin Perret. *Playing with Kruskal: Algorithms for Morphological Trees in Edge-Weighted Graphs*. In CrisL.Luengo Hendriks, Gunilla Borgefors and Robin Strand, editeurs, Internal Symposium on Mathematical Morphology, volume 7883 of *Lecture Notes in Computer Science*, pages 135–146. Springer Berlin Heidelberg, 2013. (Cited on pages 29, 32, 184, 185 and 241.)

[Niemeijer 2004] Meindert Niemeijer, Joes Staal, Bram van Ginneken, Marco Loog and Michael D Abramoff. *Comparative Study of Retinal Vessel Segmentation Methods on a New Publicly Available Database*. In Proceedings SPIE Medical Imaging, volume 5370, pages 648–656, 2004. (Cited on pages 148, 156 and 157.)

[Nistér 2008] David Nistér and Henrik Stewénius. *Linear Time Maximally Stable Extremal Regions*. In European Conference on Computer Vision, pages 183–196, Berlin, Heidelberg, 2008. Springer-Verlag. (Cited on pages 25, 52, 93 and 201.)

[Osher 1988] Stanley Osher and James A Sethian. *Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations*. Journal of Computational Physics, vol. 79, no. 1, pages 12–49, 1988. (Cited on pages 168 and 169.)

[Ouzounis 2007] Georgios K Ouzounis and Michael HF Wilkinson. *Mask-Based Second Generation Connectivity and Attribute Filters*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, pages 990–1004, 2007. (Cited on pages 17 and 145.)

[Ouzounis 2011a] Georgios Ouzounis and Pierre Soille. *Pattern Spectra from Partition Pyramids and Hierarchies*. In Internal Symposium on Mathematical Morphology, volume 6671, pages 108–119. Springer Berlin Heidelberg, 2011. (Cited on pages x, 4, 7, 24, 28, 29 and 184.)

[Ouzounis 2011b] Georgios K Ouzounis and Michael HF Wilkinson. *Hyperconnected Attribute Filters Based on k-Flat Zones*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 2, pages 224–239, 2011. (Cited on pages 17 and 145.)

[Ouzounis 2012a] Georgios K Ouzounis, Martino Pesaresi and Pierre Soille. *Differential Area Profiles: Decomposition Properties and Efficient Computation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 8, pages 1533–1548, 2012. (Cited on page 84.)

[Ouzounis 2012b] Georgios K Ouzounis and Pierre Soille. The alpha-tree algorithm. theory, algorithms, and applications. EU Bookshop, 1st édition, 2012. (Cited on pages 29 and 184.)

[Pan 2009a] Yongsheng Pan. *Top-down image segmentation using the Mumford-Shah functional and level set image representation*. In International Conference on Acoustics, Speech, and Signal Processing, pages 1241–1244, 2009. (Cited on pages 118, 124, 126 and 127.)

[Pan 2009b] Yongsheng Pan, J Douglas Birdwell and Seddik M Djouadi. *Preferential Image Segmentation Using Trees of Shapes*. IEEE Transactions on Image Processing, vol. 18, no. 4, pages 854–866, 2009. (Cited on pages 7, 8, 50 and 68.)

[Pardo 2002] Alvaro Pardo. *Semantic Image Segmentation Using Morphological Tools*. In IEEE International Conference on Image Processing, pages 745–748, 2002. (Cited on pages 117 and 166.)

[Paris 2007] Sylvain Paris and Frédo Durand. *A Topological Approach to Hierarchical Segmentation Using Mean Shift*. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 1–8, 2007. (Cited on page 52.)

[Park 1996] Kyeong-Ryeol Park and Chung-Nim Lee. *Scale-Space Using Mathematical Morphology*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, no. 11, pages 1121–1126, 1996. (Cited on page 41.)

[Patton 2006] Niall Patton, Tariq M Aslam, Thomas MacGillivray, Ian J Deary, Baljean Dhillon, Robert H Eikelboom, Kanagasingam Yogesan and Ian J Constable. *Retinal Image Analysis: Concepts, Applications and Potential*. Progress in Retinal and Eye Research, vol. 25, no. 1, pages 99–127, 2006. (Cited on page 147.)

[Pauly 2003] Mark Pauly, Richard Keiser and Markus Gross. *Multi-Scale Feature Extraction on Point-Sampled Surfaces.* Computer Graphics Forum, vol. 22, no. 3, pages 281–289, 2003. (Cited on page 228.)

[Pavlidis 1977] Theodosios Pavlidis. Structural pattern recognition. Springer-Verlag Berlin ; New York, 1977. (Cited on page 30.)

[Pavlidis 1979] Theodosios Pavlidis. *Hierarchies in Structural Pattern Recognition.* Proceedings of the IEEE, vol. 67, no. 5, pages 737–744, 1979. (Cited on pages 4 and 34.)

[Perdoch 2007] Michal Perdoch, Jiri Matas and StepÃ¡n ObdrzÃ¡lek. *Stable Affine Frames on Isophotes.* In IEEE International Conference on Computer Vision, pages 1–8, 2007. (Cited on pages 52 and 94.)

[Perona 1990] Pietro Perona and Jitendra Malik. *Scale-Space and Edge Detection Using Anisotropic Diffusion.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 7, pages 629–639, 1990. (Cited on page 41.)

[Phillips 1989] Tsai-Yun Phillips, Azriel Rosenfeld and Allen C Sher. *O(log n) Bimodality Analysis.* Pattern Recognition, vol. 22, no. 6, pages 741–746, 1989. (Cited on pages 170 and 172.)

[Pietikainen 1981] Matti Pietikainen and Azriel Rosenfeld. Image segmentation by texture using pyramid node linking. Defense Technical Information Center, 1981. (Cited on pages 24 and 30.)

[Rand 1971] William M Rand. *Objective Criteria for the Evaluation of Clustering Methods.* Journal of the American Statistical Association, vol. 66, no. 336, pages 846–850, 1971. (Cited on page 192.)

[Rangayyan 2010] Rangaraj M Rangayyan, Xiaolu Zhu, Fábio J Ayres and Anna L Ells. *Detection of the Optic Nerve Head in Fundus Images of the Retina with Gabor Filters and Phase Portrait Analysis.* Journal of Digital Imaging, vol. 23, no. 4, pages 438–453, 2010. (Cited on page 149.)

[Reeb 1946] Georges Reeb. *Sur les Points Singuliers d'une Forme de Pfaff Complètement Intégrable ou d'une Fonction Numérique.* Comptes Rendus Acad. Sciences, vol. 222, pages 847–849, 1946. (Cited on page 96.)

[Ronse 2008] Christian Ronse. *Partial Partitions, Partial Connections and Connective Segmentation.* Journal of Mathematical Imaging and Vision, vol. 32, no. 2, pages 97–125, October 2008. (Cited on page 19.)

[Rosenfeld 1983] Azriel Rosenfeld. *Quadtrees and Pyramids: Hierarchical Repre-sentation of Images*. In RobertM. Haralick, editeur, Pictorial Data Analysis, volume 4 of *NATO ASI Series*, pages 29–42. Springer Berlin Heidelberg, 1983. (Cited on pages 24 and 30.)

[Rosenfeld 1984] Azriel Rosenfeld. Multiresolution image processing and analysis. Springer, Berlin, 1984. (Cited on pages 4, 6 and 34.)

[Salembier 1995] Philippe Salembier and Jean Serra. *Flat Zones Filtering, Con-nected Operators and Filters by Reconstruction*. IEEE Transactions on Image Processing, vol. 3, no. 8, pages 1153–1160, 1995. (Cited on pages vii, x, 3, 18 and 145.)

[Salembier 1998] Philippe Salembier, Albert Oliveras and Luis Garrido. *Antiexten-sive Connected Operators for Image and Sequence Processing*. IEEE Trans-actions on Image Processing, vol. 7, no. 4, pages 555–570, 1998. (Cited on pages vii, ix, x, 3, 5, 6, 9, 18, 21, 22, 24, 25, 42, 44, 45, 46, 49, 52, 60, 93, 127, 201 and 234.)

[Salembier 2000] Philippe Salembier and Luis Garrido. *Binary Partition Tree as an Efficient Representation for Image Processing, Segmentation and Informa-tion Retrieval*. IEEE Transactions on Image Processing, vol. 9, no. 4, pages 561–576, 2000. (Cited on pages vii, x, 4, 7, 24, 27, 28, 42, 49, 127, 227 and 234.)

[Salembier 2009] Philippe Salembier and Michael HF Wilkinson. *Connected Opera-tors*. Signal Processing Magazine, vol. 26, no. 6, pages 136–157, 2009. (Cited on pages vii, x, 3, 6, 9, 18, 19, 28, 42, 44, 118, 145, 146, 163 and 223.)

[Samet 1984] Hanan Samet. *The Quadtree and Related Hierarchical Data Structures*. ACM Comput. Surv., vol. 16, no. 2, pages 187–260, June 1984. (Cited on page 24.)

[Samet 1985] Hanan Samet. *Data Structures for Quadtree Approximation and Com-pression*. Commun. ACM, vol. 28, no. 9, pages 973–993, September 1985. (Cited on pages 7 and 30.)

[Schmid 2000] Cordelia Schmid, Roger Mohr and Christian Bauckhage. *Evalua-tion of Interest Point Detectors*. International Journal of Computer Vision, vol. 37, no. 2, pages 151–172, 2000. (Cited on pages 91 and 94.)

[Serra 1982] Jean Serra. Image analysis and mathematical morphology, volume 1. Academic Press, New York, 1982. (Cited on pages 5, 21, 53 and 145.)

[Serra 1988] Jean Serra. Image analysis and mathematical morphology: Theoretical advances, volume 2. Academic Press, London, 1988. (Cited on pages 5 and 21.)

[Serra 1993] Jean C Serra and Philippe Salembier. *Connected Operators and Pyramids*. In E. R. Dougherty, P. D. Gader and J. Serra, editeurs, Image Algebra and Mathematical Morphology, volume 2030, pages 65–76. San Diego, CA: SPIE, July 1993. (Cited on pages vii, x, 3 and 18.)

[Serra 1998] Jean Serra. *Connectivity on Complete Lattices*. Journal of Mathematical Imaging and Vision, vol. 9, pages 231–251, 1998. (Cited on page 17.)

[Serra 2006] Jean Serra. *A Lattice Approach to Image Segmentation*. Journal of Mathematical Imaging and Vision, vol. 24, no. 1, pages 83–130, January 2006. (Cited on page 19.)

[Serra 2012a] Jean Serra. *Tutorial on Connective Morphology*. Selected Topics in Signal Processing, IEEE Journal of, vol. 6, no. 7, pages 739–752, 2012. (Cited on page 17.)

[Serra 2012b] Jean Serra, Bangalore Ravi Kiran and Jean Cousty. *Hierarchies and Climbing Energies*. In Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, pages 821–828. Springer, 2012. (Cited on page 35.)

[Serra 2013] Jean Serra and Bangalore Ravi Kiran. *Optima on Hierarchies of Partitions*. In CrisL.Luengo Hendriks, Gunilla Borgefors and Robin Strand, editeurs, Internal Symposium on Mathematical Morphology, pages 147–158, 2013. (Cited on pages 35, 50 and 183.)

[Sethian 1999] James A Sethian. Level set methods and fast marching methods. Cambridge University Press, 1999. (Cited on pages 118 and 120.)

[Smith 1994] John Smith and S-F Chang. *Quad-Tree Segmentation for Texture-Based Image Query*. In Proceedings of the second ACM international conference on Multimedia, pages 279–286, New York, NY, USA, 1994. ACM. (Cited on page 30.)

[Snavely 2006] Noah Snavely, Steven M. Seitz and Richard Szeliski. *Photo Tourism: Exploring Photo Collections in 3D*. ACM Transactions on Graphics (TOG), vol. 25, no. 3, 2006. (Cited on page 110.)

[Soille 2003] Pierre Soille. Morphological image analysis: Principles and applications. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2 édition, 2003. (Cited on page 5.)

[Soille 2008] Pierre Soille. *Constrained Connectivity for Hierarchical Image Partitioning and Simplification*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 7, pages 1132–1145, 2008. (Cited on pages x, xvi, 4, 7, 13, 24, 28, 29, 35, 50, 182, 183, 184, 225 and 227.)

[Song 2002] YuQing Song and Aidong Zhang. *Locating Image Background By Monotonic Tree*. In Proceedings of the 6th Joint Conference on Information Science, pages 879–884. JCIS / Association for Intelligent Machinery, Inc., 2002. (Cited on pages 7 and 53.)

[Song 2003] Yuqing Song and Aidong Zhang. *Analyzing Scenery Images by Monotonic Tree*. Multimedia Syst., vol. 8, no. 6, pages 495–511, 2003. (Cited on pages 7, 26 and 53.)

[Song 2007] Yuqing Song. *A Topdown Algorithm for Computation of Level Line Trees*. IEEE Transactions on Image Processing, vol. 16, no. 8, pages 2107–2116, August 2007. (Cited on pages 26, 27 and 202.)

[Spann 1985] Michael Spann and Roland Wilson. *A Quad-Tree Approach to Image Segmentation which Combines Statistical and Spatial Information*. Pattern Recognition, vol. 18, no. 3-4, pages 257–269, 1985. (Cited on page 30.)

[sta 2013] *STARE: STructured Analysis of the Retina*. http://www.ces.clemson.edu/~ahoover/stare/, 2013. (Cited on pages 146 and 154.)

[Staal 2004] Joes Staal, Michael D Abràmoff, Meindert Niemeijer, Max A Viergever and Bram van Ginneken. *Ridge Based Vessel Segmentation in Color Images of the Retina*. IEEE Transactions on Medical Imaging, vol. 23, no. 4, pages 501–509, 2004. (Cited on pages 146, 149, 154, 156 and 157.)

[Strecha 2008] Christoph Strecha, Wolfgang von Hansen, Luc Van Gool, Pascal Fua and Ulrich Thoennessen. *On Benchmarking Camera Calibration and Multi-View Stereo for High Resolution Imagery*. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 1–8, 2008. (Cited on pages 98, 110, 111 and 238.)

[Suk 1984] Minsoo Suk and Ohyoung Song . *Curvilinear Feature Extraction Using Minimum Spanning Trees*. Computer Vision, Graphics and Image Processing, vol. 26, no. 3, pages 400–411, 1984. (Cited on page 32.)

[Takahashi 1995] Shigeo Takahashi, Tetsuya Ikeda, Yoshihisa Shinagawa, Tosiyasu L. Kunii and Minoru Ueda. *Algorithms for Extracting Correct Critical Points and Constructing Topological Graphs from Discrete Geographical Elevation Data.* Comput. Graph. Forum, vol. 14, no. 3, pages 181–192, 1995. (Cited on page 96.)

[Talbot 2007] Hugues Talbot and Ben Appleton. *Efficient Complete and Incomplete Path Openings and Closings.* Image and Vision Computing, vol. 25, no. 4, pages 416–425, 2007. (Cited on page 227.)

[Tarjan 1975] Robert Endre Tarjan. *Efficiency of a Good but not Linear Set Union Algorithm.* Journal of the ACM, vol. 22, no. 2, pages 215–225, 1975. (Cited on page 25.)

[Torresani 2008] Lorenzo Torresani, Vladimir Kolmogorov and Carsten Rother. *Feature Correspondence via Graph Matching: Models and Global Optimization.* In European Conference on Computer Vision, pages 596–609. Springer, 2008. (Cited on page 68.)

[Tuia 2009] Devis Tuia, Fabio Pacifici, Mikhail Kanevski and William J Emery. *Classification of Very High Spatial Resolution Imagery Using Mathematical Morphology and Support Vector Machines.* IEEE Transactions on Geoscience and Remote Sensing, vol. 47, no. 11, pages 3866–3879, 2009. (Cited on page 54.)

[Tuytelaars 2004] Tinne Tuytelaars and Luc Van Gool. *Matching Widely Separated Views Based on Affine Invariant Regions.* International Journal of Computer Vision, vol. 59, no. 1, pages 61–85, 2004. (Cited on page 91.)

[Tuytelaars 2008] Tinne Tuytelaars and Krystian Mikolajczyk. Local invariant feature detectors: A survey. Now Publishers Inc., Hanover, MA, USA, 2008. (Cited on page 93.)

[Unnikrishnan 2007] Ranjith Unnikrishnan, Caroline Pantofaru and Martial Hebert. *Toward Objective Evaluation of Image Segmentation Algorithms.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, pages 929–944, 2007. (Cited on page 192.)

[Urbach 2004] Erik R Urbach, Jos BTM Roerdink and Michael HF Wilkinson. *Connected Rotation-Invariant Size-Shape Granulometries.* In International Conference on Pattern Recognition, volume 1, pages 688–691, 2004. (Cited on page 54.)

[Urbach 2007] Erik R Urbach, Jos BTM Roerdink and Michael HF Wilkinson. *Connected Shape-Size Pattern Spectra for Rotation and Scale-Invariant Classification of Gray-Scale Images*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 2, pages 272–285, 2007. (Cited on pages 7, 21, 44, 45, 46, 47, 53, 54 and 162.)

[Vachier 1995] Corinne Vachier and Fernand Meyer. *Extinction Value: a New Measurement of Persistence*. In IEEE Workshop on nonlinear signal and image processing, volume 1, pages 254–257, 1995. (Cited on pages xv, 10, 60, 64, 80 and 175.)

[Vallabha 2004] Deepika Vallabha, Ramprasath Dorairaj, Kamesh Namuduri and Hilary Thompson. *Automated Detection and Classification of Vascular Abnormalities in Diabetic Retinopathy*. In Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference on, volume 2, pages 1625–1629, 2004. (Cited on page 147.)

[Van Kreveld 1997] Marc Van Kreveld, René van Oostrum, Chandrajit Bajaj, Valerio Pascucci and Dan Schikore. *Contour Trees and Small Seed Sets for Isosurface Traversal*. In Proceedings of the Thirteenth Annual Symposium on Computational Geometry, pages 212–220, 1997. (Cited on page 96.)

[Vedaldi 2008] Andrea Vedaldi and Brian Fulkerson. *VLFeat: An Open and Portable Library of Computer Vision Algorithms*, 2008. http://www.vlfeat.org/. (Cited on pages 69, 70 and 95.)

[Vese 2002] Luminita A Vese and Tony F Chan. *A Multiphase Level Set Framework for Image Segmentation Using the Mumford and Shah Model*. International Journal of Computer Vision, vol. 50, no. 3, pages 271–293, 2002. (Cited on page 120.)

[Vilaplana 2008] Veronica Vilaplana, Ferran Marques and Philippe Salembier. *Binary Partition Trees for Object Detection*. IEEE Transactions on Image Processing, vol. 17, no. 11, pages 2201–2216, 2008. (Cited on pages 7, 28, 55 and 77.)

[Vincent 1993] Luc Vincent. *Morphological Gray Scale Reconstruction in Image Analysis: Applications and efficient algorithms*. IEEE Transactions on Image Processing, vol. 2, no. 2, pages 176–201, 1993. (Cited on page 23.)

[Viterbi 1979] AJ Viterbi and JK Omura. *Principles of Digital Communication and Coding*. McGraw-HillNew York, pages 240–242, 1979. (Cited on page 44.)

[Welfer 2010] Daniel Welfer, Jacob Scharcanski, Cleyson M Kitamura, Melissa M Dal Pizzol, Laura WB Ludwig and Diane Ruschel Marinho. *Segmentation of the Optic Disk in Color Eye Fundus Images Using an Adaptive Morphological Approach.* Computers in Biology and Medicine, vol. 40, no. 2, pages 124–137, 2010. (Cited on pages 149 and 152.)

[Westenberg 2007] Michel A Westenberg, Jos BTM Roerdink and Michael HF Wilkinson. *Volumetric Attribute Filtering and Interactive Visualization Using the Max-tree representation.* IEEE Transactions on Image Processing, vol. 16, no. 12, pages 2943–2952, 2007. (Cited on pages 7, 51, 61 and 145.)

[Wilkinson 2001] Michael H. F. Wilkinson and Michel A. Westenberg. *Shape Preserving Filament Enhancement Filtering.* In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 770–777, 2001. (Cited on pages 7, 21, 45, 46, 49, 51 and 127.)

[Wilkinson 2008] Michael HF Wilkinson, Hui Gao, Wim H Hesselink, J-E Jonker and Arnold Meijster. *Concurrent Computation of Attribute Filters on Shared Memory Parallel Machines.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 10, pages 1800–1813, 2008. (Cited on page 145.)

[Xia 2010] Gui-Song Xia, Julie Delon and Yann Gousseau. *Shape-Based Invariant Texture Indexing.* International Journal of Computer Vision, vol. 88, no. 3, pages 382–403, 2010. (Cited on pages 7, 8 and 54.)

[Xu 2012] Yongchao Xu, Thierry Géraud and Laurent Najman. *Context-Based Energy Estimator: Application to Object Segmentation on the Tree of Shapes.* In IEEE International Conference on Image Processing, pages 1577–1580, 2012. (Cited on pages xv, 64, 79 and 235.)

[Xu 2013] Yongchao Xu, Thierry Géraud and Laurent Najman. *Salient Level Lines Selection Using the Mumford-Shah Functional.* In IEEE International Conference on Image Processing, pages 1227–1231, 2013. (Cited on page xvii.)

[Yang 2008] Allen Y Yang, John Wright, Yi Ma and S Shankar Sastry. *Unsupervised Segmentation of Natural Images via Lossy Data Compression.* Computer Vision, Graphics and Image Understanding, vol. 110, no. 2, pages 212–225, 2008. (Cited on page 192.)

[Zana 2001] Frederic Zana and J-C Klein. *Segmentation of Vessel-Like Patterns Using Mathematical Morphology and Curvature Evaluation.* IEEE Transac-

tions on Image Processing, vol. 10, no. 7, pages 1010–1019, 2001. (Cited on pages 148 and 157.)

[Zhu 2010]   Xiaolu Zhu, Rangaraj M Rangayyan and Anna L Ells. *Detection of the Optic Nerve Head in Fundus Images of the Retina Using the Hough Transform for Circles.* Journal of Digital Imaging, vol. 23, no. 3, pages 332–341, 2010. (Cited on page 149.)