



A graph-based mathematical morphology reader

Laurent Najman, Jean Cousty

*Université Paris-Est, Laboratoire d'Informatique Gaspard-Monge, Équipe A3SI, ESIEE Paris.
E-mail: {laurent.najman, jean.cousty}@esiee.fr*

The two authors contributed equally to this work which received funding from the Agence Nationale de la Recherche, contract ANR-2010-BLAN-0205-03.

arXiv:1404.7748v1 [cs.CV] 30 Apr 2014

ARTICLE INFO

Article history:

Graphs
Mathematical Morphology
Computer Vision
Image Analysis
Filtering
Segmentation

ABSTRACT

This survey paper aims at providing a “literary” anthology of mathematical morphology on graphs. It describes in the English language many ideas stemming from a large number of different papers, hence providing a unified view of an active and diverse field of research.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

Mathematical morphology was born almost 50 years ago (Serra, 1982), initially an evolution of a continuous probabilistic framework (Matheron, 1975). Historically, this was the first consistent non-linear image analysis theory which from the very start included not only theoretical results, but also many practical aspects, including algorithmic ones (Soille, 1999). Despite its continuous origin, it was soon recognized that the roots of this theory were in algebraic theory, notably the framework of complete lattices (Heijmans, 1994). This allows the theory to be completely adaptable to non-continuous spaces, such as graphs. For a survey of the state of the art in mathematical morphology, we recommend (Najman and Talbot, 2010).

Graphs are generic data structures that have a long history in mathematics and have been applied in almost every scientific and engineering field, notably image analysis and computer vision (Lézoray and Grady, 2012; Grady and Polimeni, 2010). Because of their many interesting properties, a current trend is to develop the classical continuous tools from signal processing onto this kind of structures (Shuman et al., 2013).

The usefulness of graphs for mathematical morphology has long been recognized (Vincent, 1989), and the same trend as in the signal processing community can be observed here (Najman and Meyer, 2012). The objective of this paper is to offer an overview of the advantages of graphs for mathematical morphology. To reach a wider audience, we decided to express all

the ideas with the least possible mathematical jargon, if possible without any equation whatsoever. We emphasize that point by using the word *reader* in the title. This paper aims at being a “literary” anthology of papers using graph in the field of mathematical morphology, describing in the English language the main ideas of many papers, pointing out where the interested researcher can find more details.

This paper is organized as follows. Section 2 describes what is a graph, what type of graphs can be encountered, and how we can build them. Section 3 explains the basis of algebraic morphology and what are the adjunctions that are used on graphs for defining elementary morphological operators. One of the most basic problem in graphs is finding paths, and section 4 gives an overview of what has been done with paths in the field. The next section 5 is divided in three parts. In the first part (section 5.1), two major morphological tools for segmentation, namely the watershed and the flat zone approach, are reviewed. The second part (section 5.2) deals with their close cousin, connective filtering. Combining these two parts together provide hierarchical segmentation and filtering, which is the object of section 5.3. Section 6 exhibits some links between graph-morphology and discrete calculus. Before concluding the paper, a penultimate section 7 describes several interesting structures that generalize graphs.

2. What is a graph and some examples of graphs for morphological processing

A graph is a representation of a set of data where some pairs of data are connected by links. Once a graph representation is adopted, the (abstraction of) interconnected data are called vertices or nodes of the graph and the links that connect vertices are called edges. An edge of the graph is then simply a pair of connected vertices. Thus, a graph is made of a set of vertices and of a set of edges. If needed, we can also associate to each vertex and/or to each edge a weight that represents some kind of measure on the data, leading to weighted graphs. Once a graph is specified, the neighbors of a data point can be obtained by considering the edges that link this data point to others in the graph. Conversely, if we know the neighbors of each data point, then we can obtain edges by considering all pairs of neighbors. Thus, another common (and equivalent) way to define a graph is to consider the sets of neighbors of each vertex instead of a set of edges; in this case, the neighbourhood relation is symmetrical.

In image processing, the first (historically) example is the case of an image itself: indeed, an image is a set of pixels with integer coordinates and color information. These pixels are often structured in a grid thanks to the classical pixel adjacency relation (*i.e.*, 4- or 8- adjacency in 2D (Kong and Rosenfeld, 1989), see Figs. 1(a) and (b)). For example, a pixel is connected to the 4 or 8 closest pixels according to the Euclidean distance between the integer coordinates. In the associated graph representation, pixels are vertices, and if two pixels are connected for the given grid-adjacency, they are linked by an edge of the graph. In the literature, the set of vertices is often denoted by V (for vertices) or N (for nodes); the set of edges is generally denoted by E . The weight of a vertex can be as simple as the gray value of the corresponding pixel, or as complex as a measure combining color information and other cues, *etc.*, taken on a patch around the pixel. The weight of an edge is generally a kind of distance between the data of the pixels linked by the given edge. For example, in the case of a gray-scale image, the edge weight can be a gradient of intensity such as the absolute difference between pixel intensities.

Some important topological properties cannot be recovered when only the 4- (or the 8-) adjacency graph is considered (Kong and Rosenfeld, 1989). The Jordan curve theorem, which states that a closed curve separates the 2D space into two regions (interior and exterior) does not hold true in this setting (see *e.g.* Fig. 1(b)). This has lead researchers to explore other adjacency relations (Aharoni et al., 1996) such as the 6-adjacency grid (also known as the hexagonal grid, see *e.g.* (Serra, 1982, Chapter VI) and Fig. 1(c)) or grids derived from the Khalimsky plane (Khalimsky et al., 1990) (see Fig. 1(d)) for which a discrete analog of the Jordan curve theorem can be expressed. This, as well as better isotropic properties, explains the popularity of the hexagonal grid for morphological processing. However, in contrast to other grids, the hexagonal grid cannot be easily extended to 3D or higher dimensional spaces (see *e.g.* (Stelldinger and Strand, 2006)). Another problem, which can be encountered with any of the 4-, 6- or 8- adjacency grid, is related to the thickness of frontiers or contours made of vertices:

a contour can contain an arbitrary number of interior points (*i.e.* points in the contour that are not adjacent to the complement of the contour) (Cousty et al., 2008a,b). With the perfect fusion grid (see Fig. 1(e)) studied in (Cousty and Bertrand, 2009) a contour is always thin. This thinness property of contours is related (by an equivalence theorem) to an interesting properties dealing with the merging of adjacent regions (Cousty et al., 2008a). This latter property, which is indeed satisfied in perfect fusion grids, gave its name to this adjacency relation.

The graphs obtained with the adjacency relations presented in the previous paragraphs are “regular”. For instance, with the 4-adjacency relation, each vertex has 4 neighbors and the patterns given by the neighborhoods of the vertices are all the same. Thus, the graph is invariant under translation, *i.e.* if one translates the original pixel coordinates, then one still obtains the same graph¹. The operator acting on the images through this kind of graphs are then called spatially invariant and were historically the first ones to be considered in mathematical morphology. Since 2005, spatially variant morphology has become increasingly interesting (Lerallut et al., 2005, 2007). The idea is to adapt the local configuration around a point to the image content: a pixel is no more adjacent to its 4- or 8-neighbors but to a pattern that locally corresponds to the image content. The local patterns can be obtained by removing some edges of a spatially invariant graph. In this case, one can threshold some edge weights to determine the edges that are kept (see *e.g.* (Cousty et al., 2013a)). One can also apply a non-local selection procedure such as keeping only the edges of a minimum spanning tree (whose definition is given in Section 5.1) of the initial graph (Stawiaski and Meyer, 2009). In these cases one obtains a graph that has less edges than the initial spatially invariant graph. It can also be interesting to have more edges or to connect pixels whose coordinates are far from each other. To this end, one may find, for each pixel of the image the closest pixels for some distance that is not only based on the coordinates. The distance can be a geodesic distance in a weighted graph (see more details in the next Section 4) or can be a distance related to a continuous feature space onto which the vertices are mapped. Therefore, the distance between two pixels with the same color can be low even if the pixels are localized far from each other. Then, the neighbors of a pixel can be all pixels at a distance less than a predefined value (Lerallut et al., 2005; Curic et al., 2012) or can be the k closest pixels for the chosen distance, where k is a predefined value that sets up the size of the neighborhood in the resulting graph (see *e.g.* (Felzenszwalb and Huttenlocher, 2004)). In the framework of second generation connectivity (Serra, 1988; Ronse, 1998; Heijmans, 1999), operators from mathematical morphology itself have been used (Ouzounis and Wilkinson, 2007) to determine the pairs of (long-distance) nodes that should be connected. Finally, in the framework of non-local means image filtering (Buades et al., 2005), a complete graph is considered to structure the image pixels (*i.e.*, any two pixels are linked by an edge). Each edge is then weighted by a similarity measure be-

¹According to Burkhardt and Siggelkow (2001), one should say *equivariant* when the operator commutes with translation

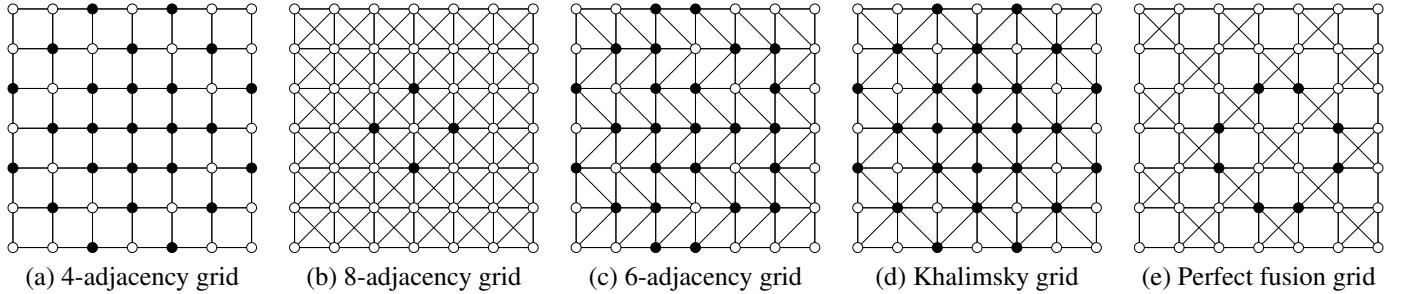


Fig. 1. Examples of pixel adjacency graphs used in image processing. Vertices are represented by dots and edges are represented by line segments. In sub-figures (b) and (e), the black vertices form the discrete analog of a Jordan curve but they do not separate the white vertices into two connected regions. In sub-figures (a,c,d), the white vertices form regions separated by frontiers made of black vertices; the frontiers are thick but they cannot be further “thinned” or “reduced by black point removal” while leaving unchanged the number of white regions.

tween small patches centered in the pixels corresponding to the extremities of the given edge.

Apart from regular grids, one of the first kind of graphs used for morphological processing was probably the family of region adjacency graphs (Pavlidis, 1977; Vincent, 1989; Beucher, 1994; Meyer, 1994). The nodes of the graph, often called superpixels, are the faces of a tessellation (or, using words from the digital world, the regions of a segmentation) of the space. Two faces are linked by an edge if they are neighbor of each other for a certain predefined adjacency. In mathematical morphology, the faces are often obtained as the flat zones of an image or the catchment basins of a watershed of the gradient magnitude of the image (see Section 5.1 for more details on these methods). However, any pixel classification method can lead to such a region adjacency graph.

Besides image analysis, graphs are often used in computer graphics. Indeed, a triangular mesh (or a triangulation), which is a very common representation for the surface of a 3D object, can be processed as a graph (see *e.g.*, Fig. 2). A triangular mesh is composed of triangles, sides (line segments) and corners (points) glued together according to certain rules (*e.g.* two triangles can have a common side or a common corner). Given a triangular mesh one can consider the graph whose vertices are the corners of the triangles and whose edges are the pairs of corners that are the extremities of a same side. When the triangular mesh satisfies the additional rule of a pseudomanifold (*i.e.* when each side belongs to exactly two triangles), a dual graph can also be built: each triangle is a vertex of the graph and two vertices are linked by an edge if the corresponding triangles share a common side. The vertices and edges of these graphs can be weighted with an information relative to the mesh: this can be a colorimetric or a geometric information. For instance, it is possible to weight these graphs with a function related to the curvature of the surface (Mangan and Whitaker, 1999; Philipp-Foliguet et al., 2011). Another possibility is to weight each edge of the dual graph with the face angle between the corresponding two triangles.

In computer graphics, unstructured cloud points are also often available. In order to build a graph over this data, one can again consider the closest neighbors of each point for a given distance. Another interesting possibility consists of building the Delaunay triangulation of the cloud points and to derive a

graph from this triangulation. In the computer graphics community, this leads to a multiscale hierarchical representation of the data, called the α -shapes (Edelsbrunner and Mücke, 1994), that were later considered in morphology by Loménie and Stamenon (2008); Loménie and Racoceanu (2012).

To finish this section, let us mention two applications of mathematical morphology in original graphs. In the first one, morphological segmentation operators are used as an image classifier (Papa et al., 2012). To this end, a given image database is structured by a weighted graph before applying morphological operators: each vertex is an image and two related images are connected by an edge that is weighted by a similarity measure. In the second application (Xu et al., 2012), graph based morphology is used for regularizing the features associated to a shape space representing an image. The shape space is a weighted graph called the component tree of the image (see more details in Section 5.2). The nodes are the shapes (components) appearing in the images and there is an edge between two shapes if they are included in each other. The weight of the nodes are provided by the shape descriptors.

3. Adjunctions and basic morphological operators

The algebraic basis of mathematical morphology is the lattice structure and the morphological operators act on lattices (Serra, 1988; Heijmans and Ronse, 1990; Ronse and Serra, 2010). In other words, the morphological operators map the elements of a first lattice to the elements of second one (which is not always the same as the first one). A lattice is a partially ordered set such that for any family of elements, we can always find a least upper bound and a greatest lower bound (called a supremum and an infimum). The supremum (*resp.*, infimum) of a family of elements is then the smallest (greatest) element among all elements greater (smaller) than every element in the considered family.

The classical lattice for binary image processing contains all shapes which can be drawn in the considered image, namely it is the family of all subsets of image pixels. The supremum is given by the union and the infimum by the intersection. A morphological operator is then a mapping that associates to any subset of pixels (a shape) another subset of pixels. Similarly, given a graph, one can consider the lattice of all sub-

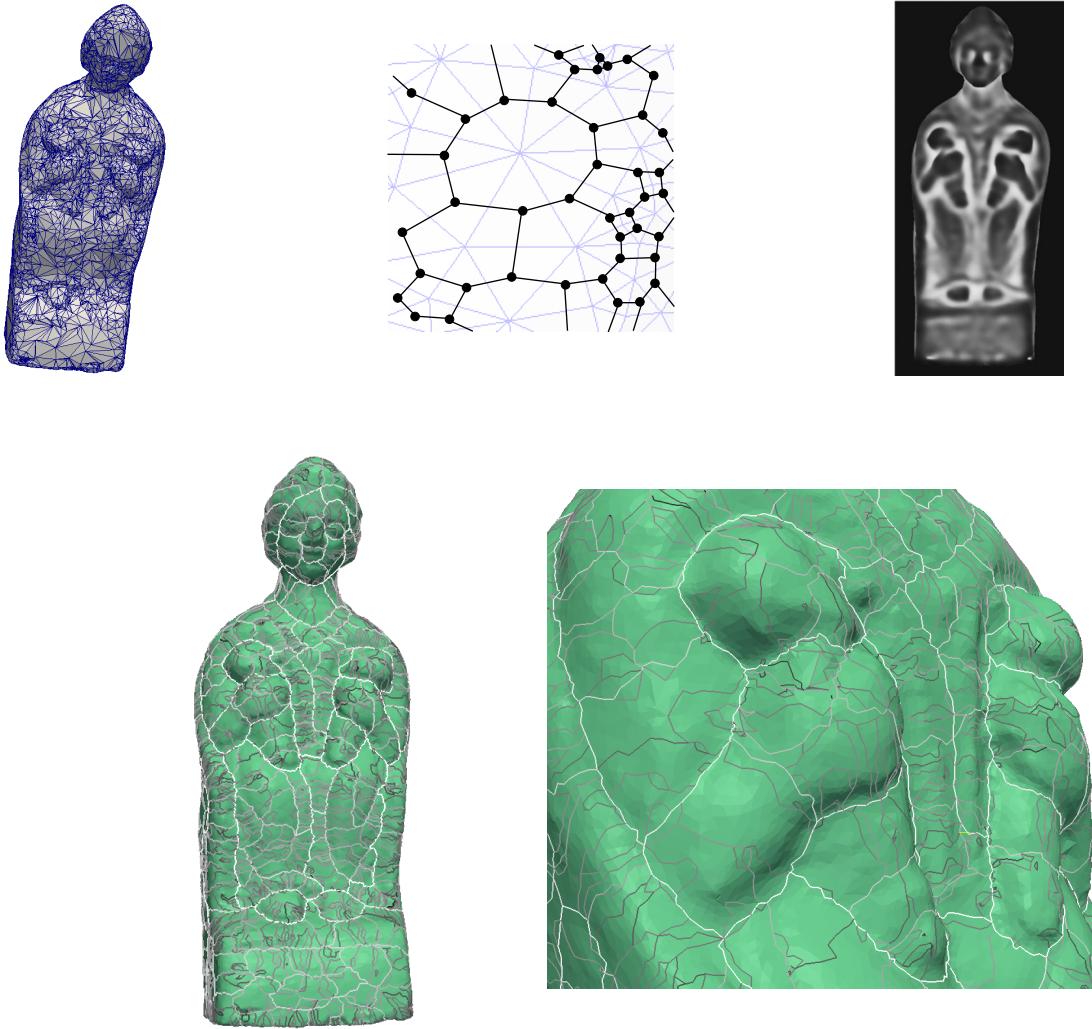


Fig. 2. Illustration of the segmentation of the surface of a 3D object. First row: a triangular mesh, a crop on its associated dual graph, and its pseudo-inverse curvature. Second row: a saliency map representing a hierarchical segmentation of the surface. A framework for the indexing and retrieval of ancient artwork 3D models, using shape descriptors adapted to the surface regions of the segmentations, is detailed in (Philipp-Foliguet et al., 2011). The mesh is provided by the French Museum Center for Research and Restoration (C2RMF, Le Louvre, Paris).

sets of vertices (Vincent, 1989) and the lattice of all subsets of edges (Cousty et al., 2009b, 2013a). The supremum and infimum in these lattices are also the union and intersection. In some cases, it is also interesting to consider a lattice whose elements are graphs, so that the inputs and outputs of the operators are graphs. In particular, when the workspace is a graph (*e.g.* a pixel adjacency graph defined from an image), it is interesting to consider the lattice of all its subgraphs (Cousty et al., 2009b, 2013a): a graph is a subgraph of another when both the vertex and edge sets of the two graphs are included in each other. In the lattice of subgraphs, the supremum or union (*resp.*, the infimum or intersection) of two graphs is defined by the union (*resp.*, intersection) of the vertex and edge sets.

The algebraic framework of morphology relies mostly on a relation between operators called adjunction (Serra, 1988; Heijmans and Ronse, 1990). This relation is particularly interesting, because it extends single operators to a whole family of other

interesting operators: having a dilation (*resp.*, an erosion), an (adjunct) erosion (*resp.*, a dilation) can always be derived, then by applying successively these two adjunct operators a closing and an opening are obtained in turn (depending which of the two operators is first applied), and finally composing this opening and closing leads to alternating filters. Each of these operators satisfy a set of remarkable properties that are interesting in particular in the context of noise cleaning (more details on the use of morphological operators for image denoising are provided in the next paragraphs and illustrated in Fig. 3). Firstly, they are all increasing, meaning that if we have two ordered elements, then the results of the operator applied to these elements are also ordered, so the morphological operators preserve order. Additionally the following important properties hold true:

- the dilation (*resp.*, erosion) commutes under supremum (*resp.*, infimum);

- the opening, closing and alternating filters are indeed morphological filters, which means that they are both increasing and idempotent (after applying a filter to an element of the lattice, applying it again does not change the result);
- the closing (*resp.*, opening) is extensive (*resp.*, anti-extensive), which means that the result of the operator is always larger (*resp.*, smaller) than the initial object;

In binary morphology on a graph, as initially proposed by Vincent (1989), a “natural” dilation maps any subset of vertices to the vertices that are neighbors of a vertex in that subset. The adjunct erosion is then the set of all vertices whose neighborhood is included in the initial set. Intuitively, one can guess that dealing also with the edges of a graph can help for reaching a better precision (Meyer and Angulo, 2007; Meyer and Lerallut, 2007; Cousty et al., 2009b, 2013a). This was the motivation for defining the analog “natural” dilation of a subset of edges (Cousty et al., 2009b, 2013a): it contains all edges which are adjacent to (*i.e.* which share a common vertex with) an edge in the initial subset. The adjunct erosion of a subset of edges contains each edge whose neighborhood (*i.e.* the set of all edges adjacent to a given edge) is included in the initial subset. Interestingly, when one applies simultaneously the vertex and edge natural dilations to the vertex and edge sets of a subgraph, the resulting pair of edge and vertex sets is still a subgraph, thus defining a natural dilation on subgraphs (Cousty et al., 2009b, 2013a). The adjunct erosion is obtained by the simultaneous applications of the vertex and edge erosions.

From a methodological viewpoint, in the usual framework of mathematical morphology, one has to choose a structuring element that parametrizes the operator. With morphology on graphs, the choice of a structuring element is, in general, replaced by the choice of the edge set that indicates which data are connected (see (Heijmans et al., 1992; Heijmans and Vincent, 1992) for a framework of morphology on graphs where one must choose both an edge set and a second “graph” that plays the role of a structuring element). In the digital setting, there is a direct correspondence between these two approaches. However, the use of graphs opens the door to the processing of many kind of data (as seen in Section 2) and to new operators such as those described in the next paragraphs.

The natural operators described above can be redefined and enriched through the use of four elementary operators that are building blocks (introduced in (Meyer and Angulo, 2007; Meyer and Lerallut, 2007) and further studied in (Cousty et al., 2009b, 2013a)) for morphology on graphs:

1. the vertex-edge dilation is a dilation that maps any set of vertices to the set of edges that contain at least one of these vertices;
2. the edge-vertex erosion, which is the adjunct erosion of the previous vertex-edge dilation, maps any set of edges to the set of vertices completely surrounded by edges of this set of edges (*i.e.*, vertices whose adjacent edges all belong to this set of edges);
3. the edge-vertex dilation is a dilation that maps any set of edges to the set of vertices which are contained in one of these edges; and

4. the vertex-edge erosion, which is the adjunct erosion of the previous edge-vertex dilation, maps any set of vertices to the set of edges whose two extremities lie in the initial set of vertices.

The natural dilation on vertices (*resp.*, edges) is simply the composition of the vertex-edge (*resp.*, edge-vertex) dilation and the edge-vertex (*resp.*, vertex-edge) dilation, whereas the associated erosion on vertices (*resp.*, edges) is the composition of the vertex-edge (*resp.*, edge-vertex) erosion and the edge-vertex (*resp.*, vertex-edge) erosion. Since the four operators defined above can be grouped as pair of adjunct operators, they also lead to openings and closings. For instance, the successive application of the vertex-edge dilation and the edge-vertex erosion is the closing which, given a set of vertices, fills in the points which do not belong to the set but which are completely surrounded by that set (*i.e.* the points whose (strict) neighborhood is completely included in that set). Note that this closing is not the same as the one obtained by composition of the natural dilation and erosion. In fact, one can prove that the results of the two closings are ordered (when applied to the same subset of vertices the result of the first one is always included in the result of the second one). This leads to interesting granulometries and alternating sequential filters.

The composition of any two dilations is still a dilation. Hence, by successive applications of elementary dilations (a same dilation can possibly be applied several times), one obtains series of dilations, adjunct erosions, openings and closings. When the dilations used in the compositions are those described in the previous paragraphs (*i.e.*, the natural dilations or the vertex-edge and edge-vertex dilations), the associated series of closings (*resp.*, openings) is ordered: when applied to a same object, the result obtained with one closing (*resp.*, opening) of the series is always smaller (*resp.*, greater) than the result obtained with the next closings of the series. These series of openings and closings, called granulometries, are interesting for studying size distributions of subsets of vertices, subsets of edges and subgraphs of a graph (see *e.g.* (Ronse and Serra, 2010; Couprise and Talbot, 2010)). Furthermore, from granulometries, series of alternating sequential filters can be derived: each of them is a sequence of intermixed openings and closings of increasing size. These operators (which, contrarily to openings and closings, are not extensive or anti-extensive) progressively filter the objects in a balanced and progressive way. They constitute interesting tools for simplifying subsets of vertices, subsets of edges and subgraphs of a graph. Fig. 3 (top row) presents the result of such a filtering procedure for a subset of pixels considered in the 4-adjacency graph. In this illustration, the edge-vertex and vertex-edge dilations were used to obtain the alternating sequential filters. As detailed in (Cousty et al., 2013a), if, instead of the edge-vertex and vertex-edge dilations, the natural dilations were used, then the resulting filter would be less performing.

The morphological operators presented in the previous paragraphs are all increasing. As such, they all induce stack operators acting on functions weighting the vertices and/or edges of a graph (see (Wendt et al., 1986) for stack operators, (Serra, 1982; Maragos and Schafer, 1987; Heijmans, 1991; Ronse, 2006) for

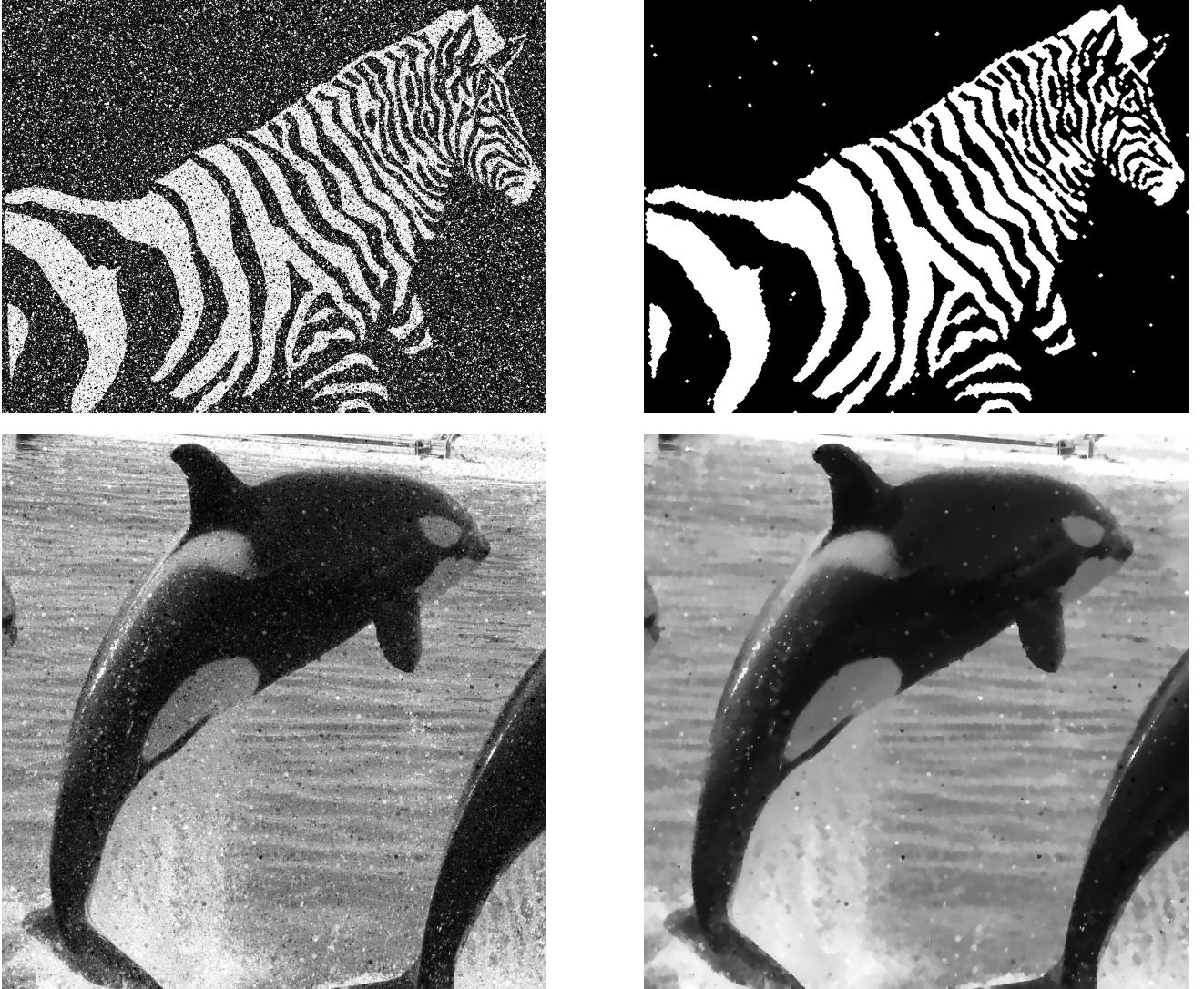


Fig. 3. Illustration of morphological alternating sequential filters on graphs. The alternating sequential filters are obtained thanks to the vertex-edge and edge-vertex dilations presented in Section 3. Top (*resp.*, bottom) row: the filtering (right) is applied to a binary (*resp.*, grayscale) image (left) considered in the 4-adjacency graphs (*resp.*, in a spatially variant adjacency graph). The corresponding filterings in the usual pixel-based framework of structuring elements (*i.e.* the filters obtained on graphs from the natural dilation) are less performing (see details in (Cousty et al., 2013a)).

stack operators in the context of flat mathematical morphology, and (Bertrand, 2005, 2007b) for stack operators in the context of watershed image segmentation). This allows for the definition of morphological operators for weighted graphs, and thus for grayscale images, to be systematically inferred from the ones on non-weighted graphs (see (Cousty et al., 2013a)). The idea is to decompose a function into level-sets by thresholding, then to apply a same operator to each level-set, before reconstructing a resulting function by “stacking” these results. Fig. 3 (bottom row) presents the results obtained with the grayscale extension of the graph alternating sequential filters presented in the previous paragraph. Here the operator is applied to a grayscale image structured by a spatially variant graph obtained by removing the edges of the 4 adjacency graph connecting two pixels with a high difference of intensity.

4. Paths and shortest paths

A classical problem in graph theory is to find a shortest path linking two points (Dijkstra, 1959) (Note that there may exist several such shortest paths). It is not surprising that paths and shortest paths find many applications in image processing and computer vision (Peyré et al., 2010).

In a graph, a path is a sequence of vertices such that any two successive vertices are linked by an edge. Depending of the applicative context, several notions of length can be associated to paths. The simplest one, when weights are not considered, consists of counting the number of edges in the path. When weights are associated to edges, one can for instance sum the edge weights along the path or consider the maximum edge weights of the path (Pollack, 1960; Udupa and Samarasekera, 1996). Similar strategies can be adapted for vertex-weighted graph. An optimal or shortest path between two points is then

a path of minimal length among all the paths linking these two points. In graph theory, finding the length of the shortest paths from a given vertex to all other vertices of the graph is a well-studied problem. When the weights are always positive², the algorithm proposed by Dijkstra (1959) provides an efficient solution.

An elementary use of paths is the computation of a distance map: from any pixel of an image, one can compute the distance (length of the shortest path) to the nearest obstacle vertex; labeling every vertex with this distance provides what is called a *distance transform* (Rosenfeld and Pfaltz, 1968; Fabbri et al., 2008). A common obstacle vertex is a pixel of an object in a binary image. An interesting property of distance maps is the following: a thresholding of a distance map for a given value m yields a dilation of size m of the object. If the graph is unweighted, then the dilation is exactly the natural dilation on vertices described in the previous section (Vincent, 1989). If the graph is weighted, then we still get an algebraic dilation, however with a different geometric outcome. In general, a binary object dilated of size $m + n$ on a weighted graph is not equal to the dilated of size m of the same object dilated of size n . As a special case, this composition law holds true for the dilations on non-weighted graphs.

A notable use of paths is for morphological filtering (Heijmans et al., 2005) of images that depict thin objects of interest. Path openings and closings are algebraic morphological operators using families of paths. Indeed, paths are thin and oriented structuring elements that are not necessarily perfectly straight. Hence, paths openings and closings offer more flexibility than line-based openings and closings. Several variations around that notion have been explored in the literature (Talbot and Appleton, 2007; Cokelaer et al., 2012; Morard et al., 2014) and in applications (Valero et al., 2010; Tankyevych, 2010; Morard, 2012).

Many other usages of paths can be found in the literature. A pioneering work (Vincent, 1998) aims at finding linear features in images as optimal paths. A more recent and popular contribution, called *seam carving* (Avidan and Shamir, 2007), is aimed at content-aware image resizing. A seam is an optimal path connecting two image borders, either from top to bottom (vertical) or from left to right (horizontal). The length of a seam is given by a measure of contrast of the pixels along the path. Removing the least important seams removes redundant part of the image, and thus makes it possible to resize the image without distorting its content. Other applications of the very same idea include contour extraction (Falcão et al., 1998) (optimal paths between two seed points are good contour candidates) and segmentation and matting³ (Saha and Udupa, 2001; Falcão et al., 2004; Bai and Sapiro, 2007) (specifying several user-provided seeds, each region of the segmentation is given by the vertices that are closest to one of the seeds with respect to all the others seeds).

More generally, one can compute for any pixel of the image,

²This condition can be relaxed, see (Falcão et al., 2004); see also the Bellman-Ford algorithm (Cormen et al., 2001).

³Matting refers to the problem of accurate foreground estimation.

an optimal path of a given length. By selecting several seeds, one obtain an image of paths that has many applications (Cohen and Kimmel, 1997). Choosing the correct seed pixels is in general application-dependent (Rouchdy and Cohen, 2008). An interesting choice is to choose as seeds all image pixels (Bismuth et al., 2012). We can also add some regularity constraints on the paths, for example, we can request them to be polygonal: indeed, polygonal paths are less tortuous than usual optimal paths. Polygonal Path Images (Bismuth et al., 2012) (PPI) are useful tools for enhancing thin objects in images: for example, one can count the number of paths of the PPI that run through a given pixel; the higher this number, the higher the probability of presence of an actual thin object (see Fig. 4 for an example). Other uses of such maps are described in (Bismuth, 2012).

As seen in this section, a great variety of powerful image operators can be implemented using optimal paths. In the context of graph-based image processing applications, this approach has been promoted notably under the name of image foresting transform (Falcão et al., 2004; Falcão and Bergo, 2004; Papa et al., 2012) (IFT). In particular, IFT allows the implementation of operators based on connectivity: region growing, ordered propagation, watershed, flooding, geodesic dilation, morphological reconstruction, etc. The IFT framework is thus a first unifying framework for presenting such operators. In the next section, we detail another framework for connected operators, based on optimum spanning forests.

5. Connected filters, watersheds and hierarchies

In this section, we review morphological segmentation (Section 5.1) and filtering methods (Section 5.2) that rely on the notion of connected components. These segmentation and filtering methods are deeply related: in general, the filtering methods lead to interesting segmentation in (quasi) flat zones whereas the segmentation methods lead to a cartoon (filtered) image where all vertices of a region take a constant value such as the mean of the original values in the region. In many cases, when the results depend on a scale parameter, the set of all possible results are organized as a hierarchy (Section 5.3). We conclude the section with an important practical point, the design of criteria adapted to the task (Section 5.4).

5.1. Segmentation: flat zones, watersheds and minimum spanning forest

Image segmentation is the task of delineating objects of interest that appear in an image, or more generally in a graph. In many cases, the result of such a process, also called a segmentation, is a set of connected regions which are composed of vertices, and are separated by a frontier. Depending on the applicative context, the frontier set can be made of vertices or can be an inter-vertices separation made of edges. In the first case, a formal notion of frontier is the one given by a binary watershed or cleft (Bertrand, 2005; Cousty et al., 2008a) and in the second case graph cuts (Diestel, 1997; Boykov et al., 2001; Cousty et al., 2009a) are considered as frontiers. In all cases, a region or a set of vertices is connected if there exists a path that is included in the region and that links any two of its vertices. A connected set is furthermore a connected component of

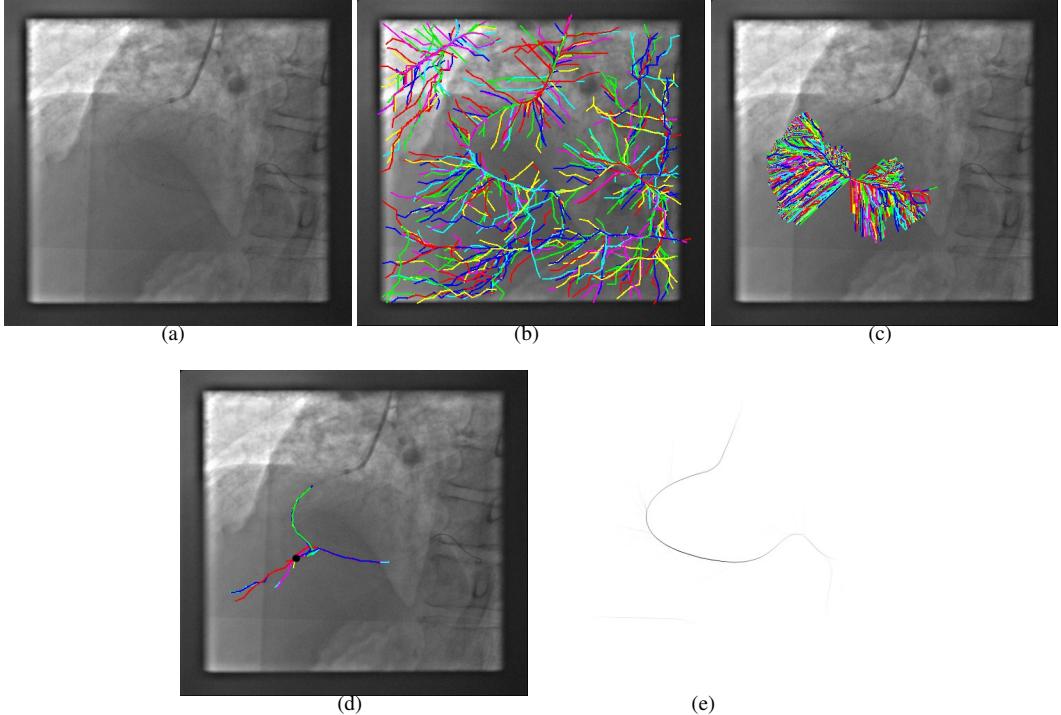


Fig. 4. (a) X-ray fluoroscopy image from an angioplasty exam illustrating a guide-wire, with a long smooth curve appearance and low contrast to noise ratio (Bismuth et al., 2012). (b): 500 locally optimal paths originating from random locations. Observe their tendency to converge to the linear structures of the image and especially to the guide-wire. (c,d) The set of paths intersecting at one given point (belonging to the guide-wire, in (c), and to the background in (d), in this case the point is indicated by the dark spot). (e) The result of path voting perfectly finds the elongated structure. See (Bismuth et al., 2012; Bismuth, 2012) for more details.

the graph if none of its proper supersets is still connected. The notion of a connected component in a graph is fundamental for defining two basic morphological segmentation methods: the quasi-flat zones and the watersheds.

The flat zones segmentation partitions the vertices of a non-negative edge-weighted graph. The partition is obtained as the set of connected components of the graph whose vertices are those of the weighted graph and whose edges are those with a null weight in the weighted graph. When the weight function is the gradient (see Section 2) of a grayscale image, the gray level in each flat zone is constant, and the flat zones are the maximal connected sets satisfying this property. In many cases, the flat zones segmentation is too fine (*i.e.*, contains too many small regions) and quasi-flat zones may be better adapted (see *e.g.* (Nagao et al., 1979; Meyer and Maragos, 1999; Soille, 2008)). To this end, the connected components are considered in the graph whose edges are those with weight below a given positive value. As we will see later in this section, (quasi-) flat zones are the basis for powerful hierarchical segmentation and filtering methods.

The watershed transform introduced by Beucher and Lantuéjoul (1979) for morphological segmentation and later popularized by Vincent and Soille (1991) is used as a fundamental step in many image segmentation procedures. A grayscale image, or more generally a function, is seen as a topographic surface: the gray values become the elevations, the basins and valleys correspond to dark areas whereas the mountains and crest lines correspond to light areas. Intuitively, the

watershed is a subset of the domain, located on the ridges of the topographic surface, that delineates its catchment basins. It may be thought of as a separating line-set from which a drop of water can flow down towards several minima. For applications to image segmentation, the watershed is often computed from the gradient magnitude of an image. Therefore, the resulting contours are located on high gradient contours of the image, which often correspond to the borders of the objects of interest (see *e.g.*, Figs. 5 and 6).

Following the intuitive drop of water principle presented in the previous paragraph, the watershed cuts, a notion of a watershed in edge-weighted graphs, were introduced in (Cousty et al., 2009a). A watershed cut is indeed a graph cut: it is only made of edges and it partitions the vertex set of the underlying graph. The consistency of watershed cuts was established by Cousty et al. (2009a): they can be equivalently characterized by their catchment basins (through a steepest descent property) or by their dividing lines (through the drop of water principle). In a discrete framework, watershed cuts are the first watershed definition that satisfies this natural consistency property. Furthermore, a global optimality property of watershed cuts is provided in (Cousty et al., 2009a) by an equivalent characterization in terms of minimum spanning forests.

The minimum spanning tree (MST) problem (Cormen et al., 2001) is one of the most typical and well known problems of combinatorial optimization: given a connected edge-weighted graph, find a connected subgraph that is spanning (*i.e.* whose vertex set is the same as the given edge weighted graph) and

whose weight is minimal, the weight of a subgraph being the sum of the weights of its edges. A minimum spanning tree of an edge-weighted graph can be computed by efficient and easy to implement algorithms (Nešetřil et al., 2001; Kruskal, 1956; Prim, 1957). For tackling image segmentation problems, we are interested by optimal structures that are not necessarily connected since we look for segmentations made of several connected regions. In this case, minimum spanning forests (MSF) are adapted: given a set of “root” vertices, a MSF is a minimum weight subgraph among the family of all spanning subgraphs such that each connected component contains exactly one root. The first links between watershed segmentations and MSFs were drawn by Meyer (1994). Later Cousty et al. (2009a) proved that the catchment basins provided by watershed cuts and the connected components of the MSFs rooted in the regional minima of the weight map are the same. As we will see in the next section, minimum spanning trees are also deeply related to hierarchical segmentations or more generally to hierarchical representations of data.

Additionally, watershed cuts have also been characterized in terms of shortest paths (Cousty et al., 2010a), drawing a link with the IFT framework described in the previous section (see also (Audigier and Lotufo, 2007b) for a link between minimum spanning forest and IFT) and in terms of flooding (Cousty et al., 2010a), making a link with the watershed presentation popularized in the 90’s (Vincent and Soille, 1991; Meyer and Beucher, 1990; Meyer, 1991; Beucher and Meyer, 1992). Links between watershed cuts and other popular graph based segmentation methods such as min-cuts or random walks were established in (Allène et al., 2010) and (Couprie et al., 2011b) respectively. As far as we know, similar properties have not been obtained in other discrete settings. In particular, when one wants to obtain as a watershed a separation made of vertices, this results in weaker properties (see counter examples in (Najman et al., 2005; Cousty, 2007)). Among the watershed definitions or algorithms producing a separation made of vertices, the topological watershed (Couprie and Bertrand, 1997), which is defined for vertex-weighted graphs, can be characterized by interesting properties of contrast preservation (Bertrand, 2005, 2007b). It was shown in (Cousty et al., 2010a) that these properties are also satisfied by watershed cuts. Given a weight function, it must be noted that there exist, in general, several watersheds. The choice of one of these watersheds can be arbitrary or based on a (optimal) criterion (see discussions related to this subject in (Meyer and Najman, 2010; Audigier and Lotufo, 2007a; Couprie et al., 2011b; Straehle et al., 2013)).

Efficient algorithms for computing watersheds is an intense subject of research since its introduction in the late 70’s. Vincent and Soille (1991), followed by Meyer (1991), were the first to propose linear-time complexity watershed algorithms relying on sorting the pixels according to their gray level and on a hierarchical priority queue, respectively. The topological watershed (Couprie et al., 2005) can be computed in quasi-linear time thanks to the min tree (see Section 5.2) of the function. The watershed cuts can be obtained in linear time (Cousty et al., 2009a, 2010a), without any sorting or auxiliary data structures such as a hierarchical queue or a component tree. The interesting trade-

off between the precision of the watershed contours and the low computational costs is an important reason for the popularity of watersheds in applications.

When the methods described in this section are applied for analyzing an image, they often produce an over-segmentation: the obtained partitions are too fine and contain more regions than objects of interest appearing in the image. Marker-based (or seed-based) segmentation is a usual procedure to prevent this over-segmentation. Given a set of “seed” or “root” vertices, which mark regions of interest in the image, the idea is to obtain a cut or a partition of the vertices such that each region contains exactly one seed. In mathematical morphology, this methodology is presented and developed in (Meyer and Beucher, 1990; Beucher and Meyer, 1992) under the name of watershed from markers. Given a set of seeds, one can modify an image or function so that after this filtering, the segmentation of the transformed function is a partition associating exactly one region to each seed. For instance, in a seeded watershed procedure, one needs a function such that regional minima correspond to seeds. Connected filters, which are described in the next section, allow this kind of filtering to be performed. They also allow for producing functions such that the associated segmentations are made of exactly k regions, where k is a predefined value. The obtained regions are then the most significant according to a certain criterion used for the filtering step.

5.2. Connected filtering

In binary morphology, connected filters act by removing specific connected components of a graph, while leaving the remaining connected components perfectly preserved. The extension to weighted graphs is straightforward when we consider stacks, as described in section 3. For example, if we want to remove all round white objects from the graph, we first design an attribute or a (numerical) criterion that states how round is a component; then we consider the family C of all the connected components of all the upper level sets of the weighted graph, and we remove the components that are not round enough for the criterion. We can then reconstruct a filtered weighted graph with the remaining components. From an algorithmic standpoint, an efficient implementation relies on the fact that the family C can be structured in a tree, called the max-tree in the literature (Salembier et al., 1998). Indeed, any two connected components of C are either disjoint or nested. There exists fast algorithms for computing this max-tree (Najman and Couprie, 2006; Berger et al., 2007; Wilkinson, 2011), see (Carlinet and Géraud, 2013) for a survey and a comparison.

From a high-level standpoint, such a filtering is equivalent to a thresholding of the max-tree, seen as a node-weighted graph whose nodes are the components and weights are given thanks to the criterion. When the criterion is increasing (meaning that if a connected component A is included in another component B , then its attribute is lower than the attribute of B), the thresholding amounts to cutting branches in the tree (see Fig 7 for an example). However, the majority of useful criteria are not increasing. Thresholding then removes nodes within a branch, and thus, as classical image thresholding that does not take the pixel context into account, is not very robust to noise: although

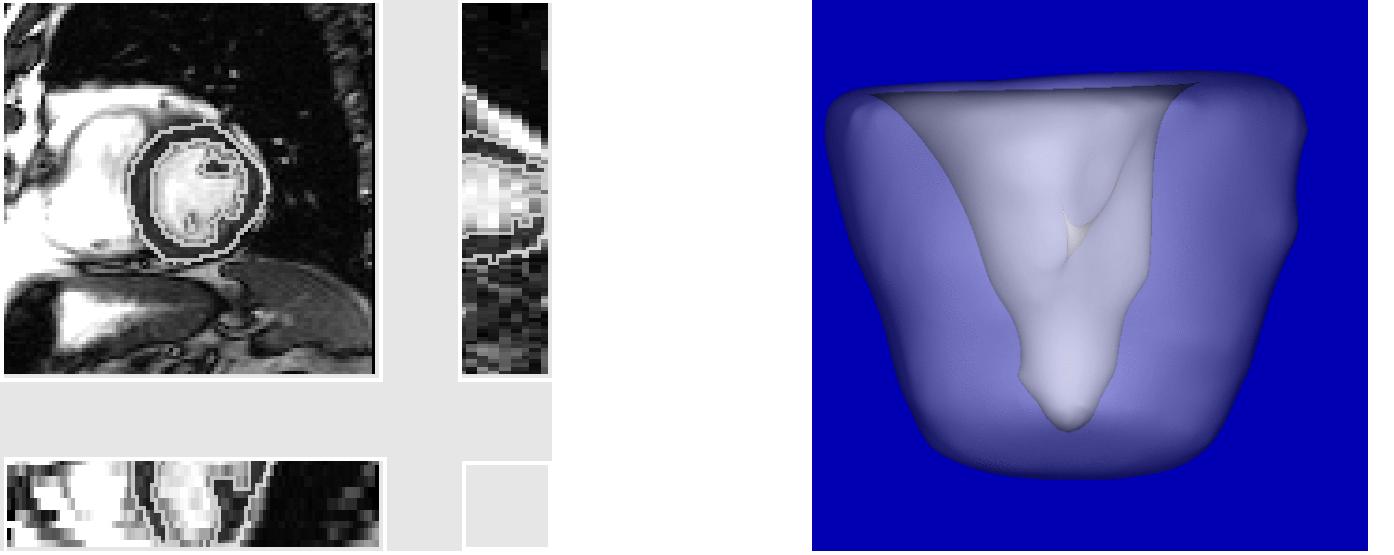


Fig. 5. Example of 3D+t (time+space) left ventricular myocardium segmentation by watershed cuts. Left: three orthogonal sections of a cardiac 3D MRI superimposed with the internal border of the segmented left ventricular myocardium. Right: a three dimensional rendering of the segmented object. The watershed cut is computed in 4D (considering time as a supplementary dimension to the space) from markers obtained by a series of morphological operators and the resulting regions are smoothed by alternating sequential filters (see more details in (Cousty et al., 2010b)). This method has been validated by comparisons with manual segmentation performed by cardiologists (Cousty et al., 2010b) and by comparisons with other state-of-the-art methods (Lebenberg et al., 2012).

two nodes in two different branches of the tree can appear visually very similar, the criterion can identify them as being very different (see Fig. 8). Several strategies have been proposed to robustify filters (Salembier et al., 1998; Urbach et al., 2007; Salembier and Wilkinson, 2009; Salembier, 2010), they all amount to cutting whole branches of the tree: if a specific node has to be removed, then all its descendants are also removed. A fruitful and seminal idea, called *shaping* (Xu et al., 2012, 2013), is to apply a connected filter on the tree itself, seen as a weighted graph whose neighborhood relationship is given by the parenthood relationship: a node is neighbor of its parent and its children, and the weight is given thanks to the criterion. We can then build a max-tree on this graph, and use an increasing criterion on this second tree to robustly remove components.

Other trees are possible, for example the min-tree, which is made from all the connected components of the lower-level sets. The min-tree helps dealing with dark components. Both the max-tree and the min-tree are also known as the component tree (Jones, 1999; Breen and Jones, 1996; Najman and Couplie, 2006). Another tree example is the so-called *tree of shapes* (Monasse and Guichard, 2000; Caselles and Monasse, 2010; Géraud et al., 2013; Najman and Géraud, 2013), which is intuitively the tree of all the level lines of a graph. The tree of shapes deals with both white and black components at the same time, and thus is useful in producing self-dual filters. There are numerous topological issues at play here, and this line of work is intimately linked to what is done in (discrete) Morse theory (Forman, 2002), algebraic topology and persistent homology (Edelsbrunner et al., 2000) (see also section 7).

5.3. Hierarchies of partitions and optimum spanning forests

In the previous section 5.2, we did not pay strict attention to the type of graph under scrutiny. Indeed, the ideas can be applied to any weighted graph, whether it is a vertex-weighted graph or an edge-weighted graph. However, traditionally, connected filters have been applied to vertex-weighted graph. But the very same ideas can be applied to edge-weighted graphs. This has been a common practice for at least 20 years, without always a clear realization that this was indeed done. The main example has been mentioned before: the quasi-flat zones hierarchy (Salembier and Serra, 1995). Any hierarchy can be represented as a tree, called a dendrogram (see Fig. 9.b). In fact several trees can be used to represent a given hierarchy (Cousty et al., 2013b; Najman et al., 2013). As described in section 5.1, the quasi-flat zones hierarchy is obtained by thresholding an edge-weighted graph, the weights being a gradient of intensity. Two connected components of two threshold levels are either disjoint or nested, hence the tree structure. It has been shown in (Cousty et al., 2013b) that the connected components of all the thresholds (organized with the inclusion relationship) can be obtained from the min-tree of the edge-weighted graph, which can be computed by efficient algorithms. But components with exactly the same vertices can be obtained by considering only a minimum spanning tree of the edge-weighted graph (Cousty et al., 2013b), which uses less memory than the original graph and is easier to handle because it contains less redundancy (Najman et al., 2013). The min-tree of the minimum spanning tree is called the alpha-tree in the literature, and specific algorithms for computing it can be designed (Najman et al., 2013; Havel et al., 2013).

Filtering the min-tree with an increasing criterion is a process that is known as a *flooding* in the watershed literature (Meyer

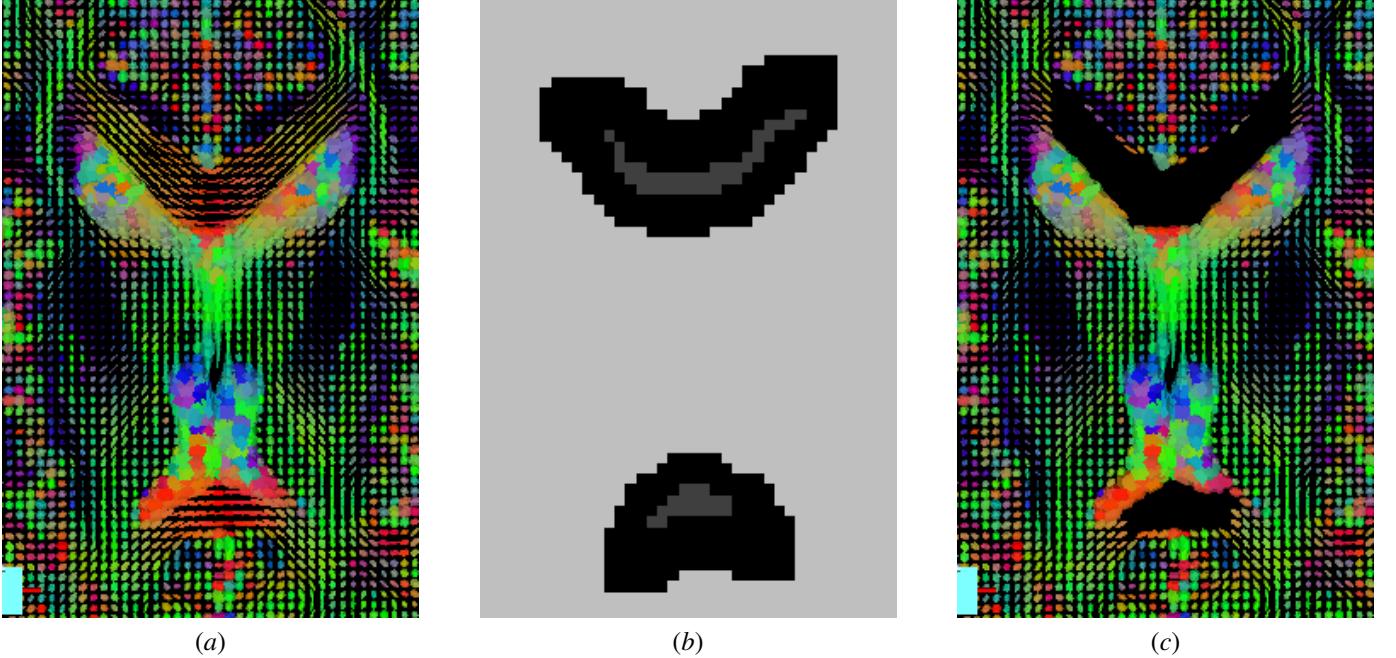


Fig. 6. Illustration of Diffusion Tensor Images (DTIs) segmentation. (a): A close-up on a cross-section of a 3D brain DTI. (b): Image representation (in the same cross-section as (a)) of the markers, obtained from a statistical atlas, for the corpus callosum (in dark gray) and for its background (in light gray) (c): Segmentation of the corpus callosum by a marker based watershed cut. The tensors belonging to the region corresponding to the seed labeled “corpus callosum” are removed from the initial DTI and thus the corresponding voxels appear black (see more details about this illustration in (Cousty et al., 2010a) and about DTI morphological segmentation in (Rittner and Lotufo, 2008)).

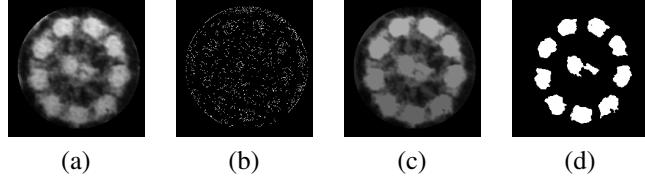


Fig. 7. (a) Original image. (b) Maxima of image (a), in white. (c) Image filtered with an increasing criterion (volume) on the max-tree. (d) Maxima of image (c), which correspond to the ten most significant lobes of the image (a).

and Beucher, 1990; Meyer and Najman, 2010; Cousty et al., 2008c). The very same process has been done on the alpha-tree in the constrained connectivity framework (Soille, 2008) in the literature (albeit without the link to the min-tree of the MST we just mentioned). The reason to restrict ourselves to increasing criteria is for transforming a hierarchy into another hierarchy. Indeed, filtering a hierarchy amounts to do a *non-horizontal cut* (Guigues et al., 2006; Meyer and Najman, 2010) in the hierarchy (see Fig. 9.b). If a criterion (depending on a parameter) is increasing, all the possible non-horizontal cuts (for all the possible values of the parameter) stack, hence providing a novel hierarchy.

Hierarchies have been exploited in image processing and computer vision since the beginning (Zahn, 1971; Morris et al., 1986; Pavlidis, 1977). However, many criteria used in practice are not increasing. A current popular example of a non-increasing criterion is proposed in (Felzenszwalb and Huttenlocher, 2004); the criterion is based on measuring the dissimilarity between elements along the boundary of the two components relative to a measure of the dissimilarity among neighboring elements within each of the two components. The algo-

rithm proposed in (Felzenszwalb and Huttenlocher, 2004) extracts from the hierarchy of quasi-flat zones a segmentation that is neither too coarse nor too fine. Several attempts to produce a hierarchy based on the same criterion can be found in the literature (Haxhimusa and Kropatsch, 2004; Guimarães et al., 2012; Xu et al., 2013). The idea of doing a shaping, *i.e.* a connected filter on the dendrogram of the hierarchy, seen as an node-weighted graph whose weight is given by the criterion of (Felzenszwalb and Huttenlocher, 2004), is explored in (Xu et al., 2013). A different approach is proposed in (Guimarães et al., 2012): the idea is to relax one of the two constraints, for example one can extract, from the hierarchy of quasi flat-zones, a (largest) hierarchy of segmentations that are not too coarse, but these segmentations can be too fine.

To conclude this section, let us mention an interesting representation of hierarchy of segmentations: it consists in stacking all the contours of the segmentations, or equivalently, in valuating each contour by the number of times it appears in the hierarchy (see Fig. 9.c and Fig 2). This notion has been introduced under the name of *geodesic saliency* of watershed contours in (Najman and Schmitt, 1996), has been independently rediscovered

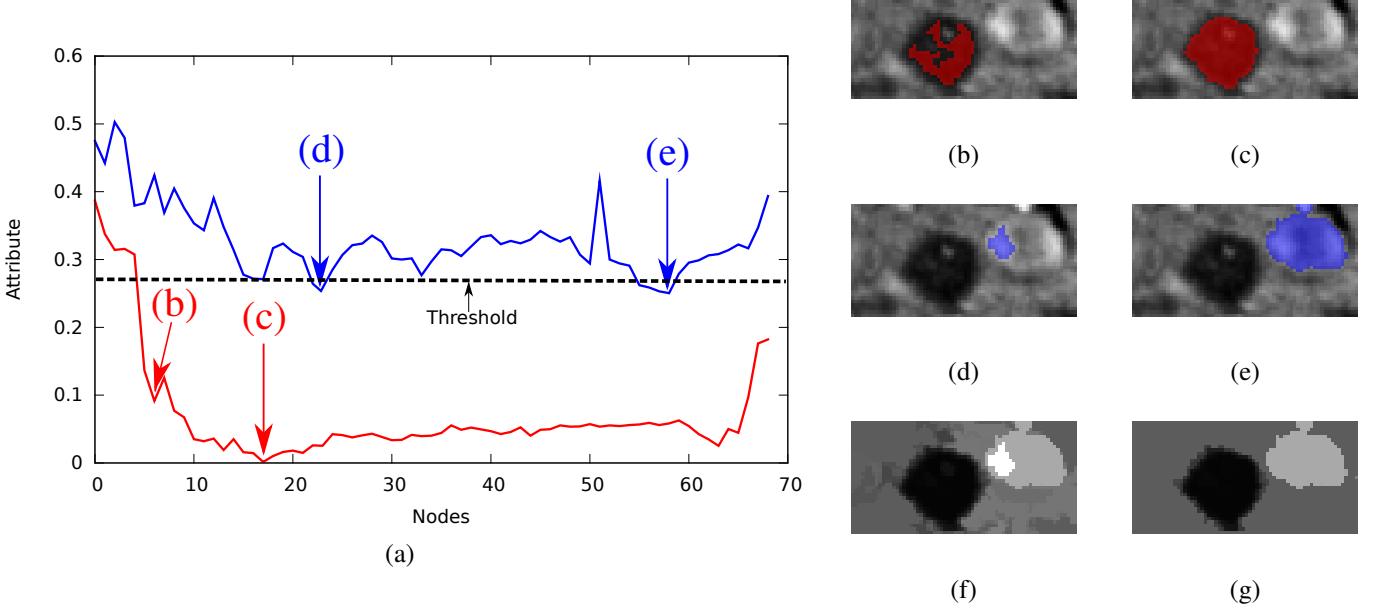


Fig. 8. (a) Evolution of a “circularity” criterion on two branches of a tree of shapes (Xu et al., 2013); (b to e): Some shapes; (f) Attribute thresholding; (g) A morphological shaping.

ered by Guigues et al. (2006), and is extensively used in (Arbelaez et al., 2011), where it is the main entry point for evaluating the quality of a given hierarchy. It has been proved in (Najman, 2011) that such a representation is formally equivalent both to a specific watershed (called *ultrametric*) and to a dendrogram, hence to a hierarchy of segmentations. Efficient algorithms to produce saliency maps are the subject of several studies. A basic algorithm (non-dedicated) can be found in (Najman, 2011), but we rather recommend using the more efficient one proposed in (Cousty and Najman, 2011), with the tree structure proposed in (Najman et al., 2013).

5.4. Design of criteria

In the previous sections 5.2 and 5.3, we briefly describe several criteria. In applications, the design of a criterion adapted to the task at hand is fundamental. In mathematical morphology, the first criteria proposed in the literature were of a geometrical nature, such as the measure of the area (Serra and Vincent, 1992; Vincent, 1994) of the component, or the depth (Grimaud, 1992) or the volume (Vachier and Meyer, 1995) of the blob corresponding to the component. Stochastic criteria were developed in (Angulo and Jeulin, 2007; Meyer and Stawiaski, 2010), with an efficient algorithm relying on watershed cuts in (Malmborg and Luengo Hendriks, 2014). Optimisation of energy-type criteria that make a balance between a data-attachment term and a regularization term, were introduced latter (Salembier and Garrido, 2000), and a formalization has been proposed under the name of *scale-set* theory (Guigues et al., 2006). A recent review paper is available in (Salembier and Wilkinson, 2009). A generalization of the scale-set theory is proposed in (Kiran et al., 2014).

Most of the previous criteria are increasing, allowing to transform from a hierarchy into another hierarchy. Non-increasing criteria are frequent in the literature (Zahn, 1971; Morris et al., 1986; Felzenszwalb and Huttenlocher, 2004), a simple geometrical example being the various moments (Westenberg et al., 2007). A generic framework for dealing with non-increasing criteria has been proposed in (Xu, 2013; Xu et al., 2012, 2013). Finally, we would like to mention other approaches based on classical classification tools (Guigues et al., 2003) or on the *Helmotz principle* (popularized in Computer Vision under the term *Number of False Alarms*) (Cardelino et al., 2013).

6. A little further with graphs: discrete calculus

We have not reviewed in this paper numerous other interesting graph-based approaches. Differential equations is one of them. Indeed, discrete settings are recently becoming the subject of numerous studies (Grady and Polimeni, 2010; Desbrun et al., 2005): the main idea is that one can write on graphs an exact discrete version of differential equations, and efficiently solve many problems. For example, some graph generalizations of the partial differential equations of mathematical morphology (Alvarez et al., 1993) can be written (Ta et al., 2011; Drakopoulos and Maragos, 2012; Purkait and Chanda, 2012), offering a greater flexibility than the continuous framework (notably, an easy integration of patch-based processing and novel applications).

We would like to mention the popular graph-based optimization approaches, such as the max-flow/min-cut one (Ford and Fulkerson, 1962; Cormen et al., 2001) (known in the computer vision community under the name of graph-cut (Boykov et al., 2001)). These methods can be used to solve a wide variety of

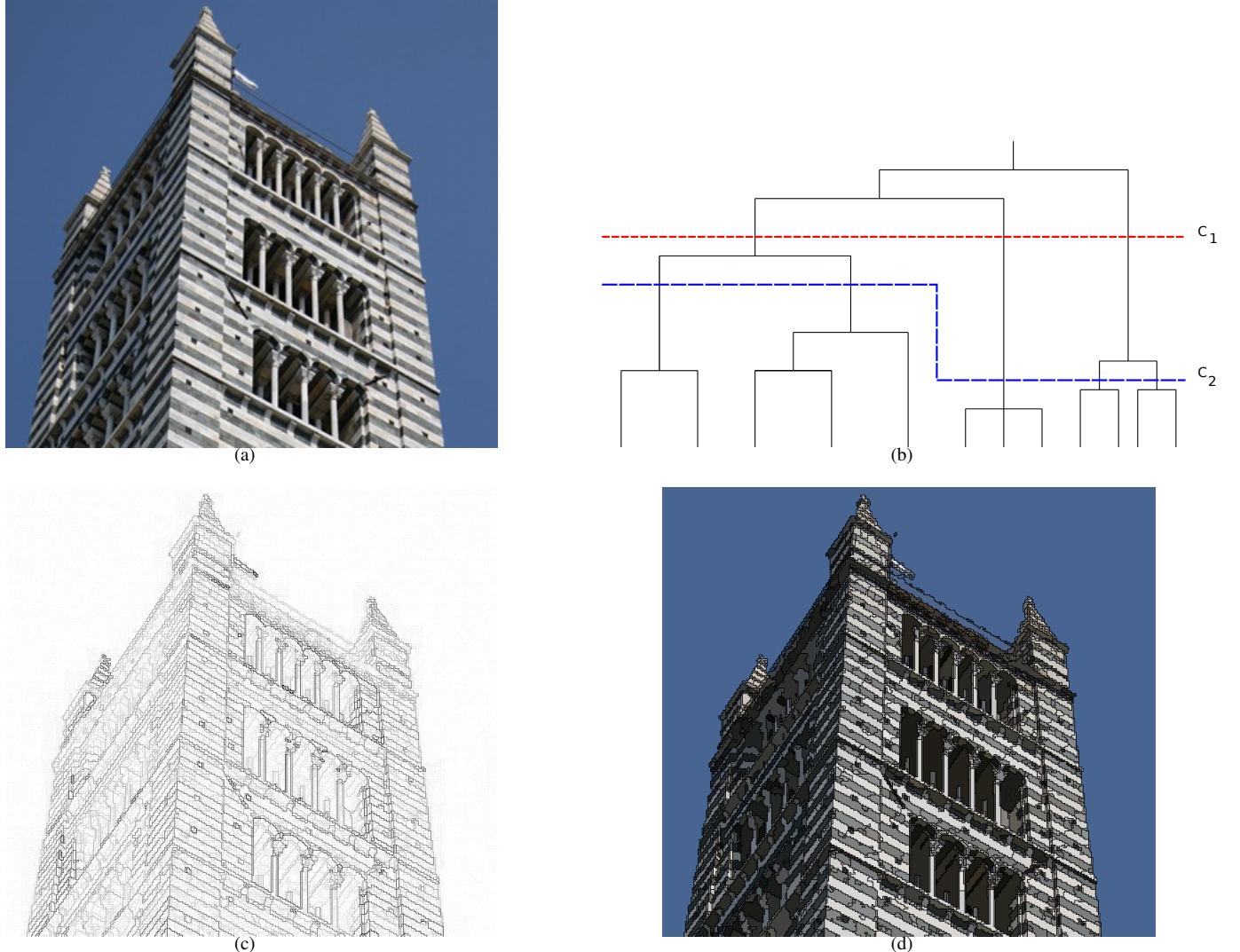


Fig. 9. Hierarchical segmentation and filtering. (a) A color image. (b) A hierarchy of flat zones (Salembier and Serra, 1995) of (a), represented by its dendrogram (min-tree of the minimum spanning tree of a color distance (Cousty et al., 2013b; Cousty and Najman, 2011; Najman et al., 2013)). The two cuts C_1 and C_2 correspond to two different flat-zone segmentations of (a), C_1 being a horizontal cut and C_2 being a non-horizontal cut (Guigues et al., 2006) (called a flooding in the morphological literature (Meyer and Najman, 2010)). (c) A saliency map (Najman and Schmitt, 1996), theoretically equivalent to the dendrogram (Najman, 2011), but with better visualisation properties. (d) A segmentation of (a) in which each region has been colorized by the mean color of the pixels forming the region. Such a coloring is a filtering of (a). The segmentation (d) is obtained equivalently by either a thresholding of (c) or by a horizontal cut of (b). Other saliency maps can be obtained through floodings of (d) or, equivalently, through non-horizontal cuts of (b).

problems that can be formulated in terms of energy minimization. Although energy minimization approaches seem hardly related to the morphological approach based on lattice theory (Serra, 2006), there exists a framework (called the power-watershed framework (Couprie et al., 2011b)) in which graph-cuts (Boykov et al., 2001), shortest paths (Falcão et al., 2004), random walks (Grady, 2006) and watershed cuts (Cousty et al., 2009a), can all be unified together, and in which we can study their links and differences. Many applications can be designed thanks to this framework, including some that are surprising for morphology: for example the (power) watershed can now be used to perform the anisotropic diffusion process (Couprie et al., 2010) or to produce a surface reconstruction from unstructured cloud points (Couprie et al., 2011a) (see Fig. 10).

We believe that many other links with seemingly unrelated methods can be searched and found: for example, the popular mean-shift approach (Cheng, 1995; Comaniciu and Meer, 2002) can be seen (Paris and Durand, 2007) as computing a max-tree in the feature space, and filtering this max-tree with a depth criterion. Exploring, detailing and emphasizing such links with other methods is indeed a promising research direction.

7. Beyond graphs: other interesting structures

Several problems related to image processing cannot be handled with undirected graphs as presented in this article.

The set of all connected sets of vertices in a graph form an algebraic structure called a connection which was introduced in (Serra, 1988, Chapter 2) and further studied notably in (Ronse, 1998; Braga-Neto and Goutsias, 2003; Ronse, 2008). The structure of a connection is a basis for studying the algebraic properties related to connectivity in many frameworks. Whereas the notion of a graph hardly extends to the case of a continuous plane, a continuous setting can be studied through a connection. Furthermore, even in the case of a finite set of vertices, the notion of a connection is more versatile than the one of a graph: for instance, with a connection, we can consider the situation where a set of three points is connected whereas any pair made of two of these three points is disconnected (in a graph at least two of these three possible pairs must be connected by an edge if the whole triple is connected). Such a connection could be obtained using, for instance, an hypergraph.

A study of morphological operators on hypergraphs was recently initiated by Bloch and Bretto (2013); Bloch et al. (2013). This framework allows higher order information to be taken into account by grouping any number of vertices into an hyper-edge. In particular, new similarity measures between images were proposed based on morphological operators in hypergraphs.

Asymmetric links between pairs of data cannot be considered in the presented framework of undirected graphs. This information can be taken into account in the framework of directed graphs. Image processing, including in particular morphological processing, in this kind of space is currently an emerging research topic (Takyevych et al., 2013; Perret et al., 2013; Miranda and Mansilla, 2014; Ronse, 2014).

For a complete topological characterization of geometrical objects, graphs (as well as connections, hyper-graphs or directed graphs) are, in general, not sufficient. Indeed, in a graph, we can make the difference between a 0-dimensional element (a vertex) and a 1-dimensional element (an edge) but the distinction with a 2-dimensional element (*i.e.* a patch of surface) cannot be made without any further information. Moreover, whereas the “cavities” of an object can be well identified with graphs as connected components of the complement of an object, characterizing a hole such as the one appearing in a torus is not feasible. Simplicial and cubical complexes generalize graphs to higher dimensions in the sense that a graph is a complex of dimension 1; furthermore, they allow the topological issues mentioned above to be tackled (Bertrand, 2007a; Couprie and Bertrand, 2009). Intuitively, a simplicial complex may be thought of as a set of elements having various dimension (*e.g.* tetraedra, triangles, edges, vertices) glued together according to certain rules. Recent studies investigated mathematical morphology in this framework, leading to morphological operators that can filter noise with respect to its dimension (Dias et al., 2011) and to links between the notions of watershed and of homotopy (Cousty et al., 2014). The framework of combinatorial maps, which provides another topology-endowed representation of discrete objects, has also been used to perform morphological filters of an image along watershed contours before building a hierarchy of segmentation (Brun et al., 2005).

8. Conclusion

As can be seen from this paper, graphs have been and currently are a prominent topic in image analysis and computer vision. With the advent of the so-called *Big Data*, we expect this trend to be extremely persistent (Lum et al., 2013) and promising for opening novel research directions. Indeed, there is no reason to restrict the application of the very same ideas we have described here to images. Any kind of data can be processed with these techniques, notably, social graph models (Grady and Polimeni, 2010) (allowing fine-grained prediction of human behavior), but also energy, transportation, sensor and neural networks to name a few.

Most of the tools presented in this paper are readily available in Pink, an open-source library (Couprie, 2014; Couprie et al., 2011c). In this library, one can find various implementations of the very same operators, according to the type of data (images, graphs, complexes, etc.) and the value type (integer, float, color, etc.) that has to be processed. A promising research direction is to write an algorithm once, and let the compiler translate the resulting code to any type of data one wants to deal with. This direction is pursued with the Olena platform (Géraud, 2014; Levillain et al., 2009), an open source framework for generic data processing.

Acknowledgements

The work of the authors on graph-based mathematical morphology owed much to the collaboration with Gilles Bertrand and Michel Couprie and, therefore, the authors feel indebted to

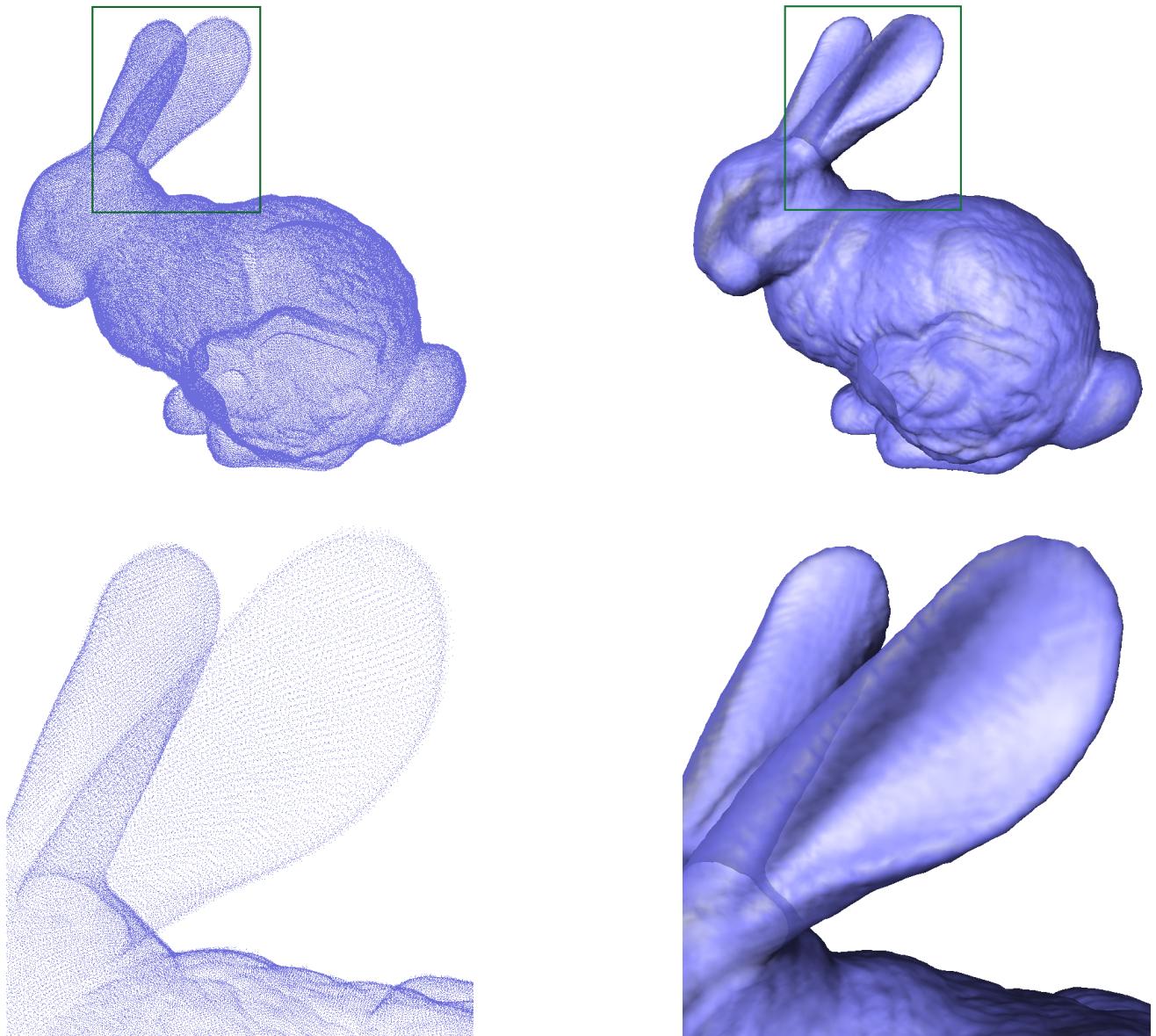


Fig. 10. Surface reconstruction with the power watershed. From a noisy set of point measurements (left), a dedicated watershed algorithm with global optimality properties computes a smooth surface (right). The algorithm is fast, robust to seed placements, and compares favorably with existing algorithms (Couprie et al., 2011a)

them. The authors would also like to thank Hugues Talbot and Christian Ronse for their careful reading of the paper.

References

- Aharoni, R., Herman, G.T., Loeb, M., 1996. Jordan graphs. *Graphical Models and Image Processing* 58, 345–59.
- Allène, C., Audibert, J.Y., Couprise, M., Keriven, R., 2010. Some links between extremum spanning forests, watersheds and min-cuts. *Image and Vision Computing* 28, 1460–71.
- Alvarez, L., Guichard, F., Lions, P.L., Morel, J.M., 1993. Axioms and fundamental equations of image processing. *Archive for Rational Mechanics and Analysis* 123, 199–257.
- Angulo, J., Jeulin, D., 2007. Stochastic watershed segmentation, in: Banon, G.J.F., Barrera, J., Braga-Neto, U.d.M. (Eds.), *Mathematical Morphology and its Application to Signal and Image Processing (ISMM 2007)*, INPE/MCT, pp. 265–76.
- Arbelaez, P., Maire, M., Fowlkes, C., Malik, J., 2011. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 898–916.
- Audigier, R., Lotufo, R., 2007a. Uniquely-determined thinning of the tie-zone watershed based on label frequency. *Journal of Mathematical Imaging and Vision* 27, 157–73.
- Audigier, R., Lotufo, R.d.A., 2007b. Watershed by image foresting transform, tie-zone, and theoretical relationships with other watershed definitions. *Mathematical Morphology and its Applications to Signal and Image Processing (ISMM)*, 277–88.
- Avidan, S., Shamir, A., 2007. Seam carving for content-aware image resizing. *ACM Transactions on graphics* 26, 10.
- Bai, X., Sapiro, G., 2007. A geodesic framework for fast interactive image and video segmentation and matting, in: *IEEE 11th International Conference on Computer Vision (ICCV 2007)*, IEEE, pp. 1–8.
- Berger, C., Géraud, T., Levillain, R., Widynski, N., Baillard, A., Bertin, E., 2007. Effective component tree computation with application to pattern recognition in astronomical imaging, in: *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, IEEE, pp. IV–41.
- Bertrand, G., 2005. On topological watersheds. *Journal of Mathematical Imaging and Vision* 22, 217–30.
- Bertrand, G., 2007a. On critical kernels. *Comptes Rendus de l'Académie des Sciences, Série Math.* I 345, 363–7.
- Bertrand, G., 2007b. On the dynamics. *Image and Vision Computing* 25, 447–54. *International Symposium on Mathematical Morphology (ISMM 2005)*.
- Beucher, S., 1994. Watershed, hierarchical segmentation and waterfall algorithm, in: *Mathematical Morphology and its Applications to Image Processing*. Springer, pp. 69–76.
- Beucher, S., Lantuéjoul, C., 1979. Use of watersheds in contour detection, in: *Proc. of the International Workshop on Image Processing Real-Time Edge and Motion Detection/Estimation*.
- Beucher, S., Meyer, F., 1992. The morphological approach to segmentation: the watershed transformation. *Optical Engineering* 34, 433–.
- Bismuth, V., 2012. Image processing algorithms for the visualization of interventional devices in X-ray fluoroscopy. *These. Université Paris-Est*.
- Bismuth, V., Vaillant, R., Talbot, H., Najman, L., 2012. Curvilinear structure enhancement with the Polygonal Path Image: application to guide-wire segmentation in X-ray fluoroscopy, in: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012*. Springer Berlin Heidelberg, pp. 9–16.
- Bloch, I., Bretto, A., 2013. Mathematical morphology on hypergraphs, application to similarity and positive kernel. *Computer Vision and Image Understanding* 117, 342–54.
- Bloch, I., Bretto, A., Leborgne, A., 2013. Similarity between hypergraphs based on mathematical morphology, in: Luengo Hendriks, C., Borgefors, G., Strand, R. (Eds.), *Mathematical Morphology and Its Applications to Signal and Image Processing*. Springer Berlin Heidelberg, volume 7883 of *Lecture Notes in Computer Science*, pp. 1–12.
- Boykov, Y., Veksler, O., Zabih, R., 2001. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23, 1222–39.
- Braga-Neto, U., Goutsias, J., 2003. A theoretical tour of connectivity in image processing and analysis. *Journal of Mathematical Imaging and Vision* 19, 5–31.
- Breen, E., Jones, R., 1996. Attribute openings, thinnings, and granulometries. *Computer Vision and Image Understanding* 64, 377–89.
- Brun, L., Mokhtari, M., Meyer, F., 2005. Hierarchical watersheds within the combinatorial pyramid framework, in: *Discrete Geometry for Computer Imagery*. Springer, pp. 34–44.
- Buades, A., Coll, B., Morel, J.M., 2005. A non-local algorithm for image denoising, in: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, IEEE, pp. 60–5.
- Burkhardt, H., Sigelkow, S., 2001. Invariant features in pattern recognition - fundamentals and applications, in: *Nonlinear Model-Based Image/Video Processing and Analysis*, John Wiley & Sons, pp. 269–307.
- Cardelino, J., Caselles, V., Bertalmio, M., Randall, G., 2013. A contrario selection of optimal partitions for image segmentation. *SIAM Journal on Imaging Sciences* 6, 1274–317.
- Carlinet, E., Géraud, T., 2013. A comparison of many max-tree computation algorithms, in: *Mathematical Morphology and Its Applications to Signal and Image Processing*. Springer Berlin Heidelberg, pp. 73–85.
- Caselles, V., Monasse, P., 2010. The tree of shapes of an image, in: *Geometric Description of Images as Topographic Maps*. Springer, pp. 9–34.
- Cheng, Y., 1995. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17, 790–9.
- Cohen, L.D., Kimmel, R., 1997. Global minimum for active contour models: A minimal path approach. *International Journal of Computer Vision* 24, 57–78.
- Cokelaer, F., Talbot, H., Chanussot, J., 2012. Efficient robust d-dimensional path operators. *IEEE Journal of Selected Topics in Signal Processing* 6, 830–9.
- Comaniciu, D., Meer, P., 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 603–19.
- Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C., 2001. *Introduction to algorithms*. The MIT Press.
- Couprise, C., Bresson, X., Najman, L., Talbot, H., Grady, L., 2011a. Surface reconstruction using power watershed, in: *Mathematical Morphology and Its Applications to Image and Signal Processing*. Springer, pp. 381–92.
- Couprise, C., Grady, L., Najman, L., Talbot, H., 2010. Anisotropic diffusion using power watersheds, in: *Image Processing (ICIP), 2010 17th IEEE International Conference on*, IEEE, pp. 4153–6.
- Couprise, C., Grady, L., Najman, L., Talbot, H., 2011b. Power watershed: A unifying graph-based optimization framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 1384–99.
- Couprise, M., 2014. The Pink Image Processing Library. <http://pinkhq.com>. Accessed: 2014-04-28.
- Couprise, M., Bertrand, G., 1997. Topological grayscale watershed transform, in: *Proc. of SPIE Vision Geometry V*, pp. 136–46.
- Couprise, M., Bertrand, G., 2009. New characterizations of simple points in 2D, 3D, and 4D discrete spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 637–48.
- Couprise, M., Marak, L., Talbot, H., 2011c. The pink image processing library, in: *EuroSciPy*. Poster and Demo Session.
- Couprise, M., Najman, L., Bertrand, G., 2005. Quasi-linear algorithms for the topological watershed. *Journal of Mathematical Imaging and Vision* 22, 231–49.
- Couprise, M., Talbot, H., 2010. Distance, granulometry and skeleton, in: Najman, L., Talbot, H. (Eds.), *Mathematical Morphology: From Theory to Applications*, ISTE / J. Wiley & Sons, pp. 265–89.
- Cousty, J., 2007. Lignes de partage des eaux discrètes : théorie et application à la segmentation d'images cardiaques. Ph.D. thesis. Université de Marne-la-Vallée, France.
- Cousty, J., Bertrand, G., 2009. Uniqueness of the perfect fusion grid on \mathbb{Z}^d . *Journal of Mathematical Imaging and Vision* 34, 291–306.
- Cousty, J., Bertrand, G., Couprise, M., Najman, L., 2008a. Fusion graphs: Merging properties and watersheds. *Journal of Mathematical Imaging and Vision* 30, 87–104.
- Cousty, J., Bertrand, G., Couprise, M., Najman, L., 2014. Collapses and watersheds in pseudomanifolds of arbitrary dimension. *Journal of Mathematical Imaging and Vision*, URL: <http://hal.archives-ouvertes.fr/hal-00871498>. to appear - DOI 10.1007/s10851-014-0498-z.
- Cousty, J., Bertrand, G., Najman, L., Couprise, M., 2009a. Watershed cuts: Minimum spanning forests and the drop of water principle. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 1362–74.
- Cousty, J., Bertrand, G., Najman, L., Couprise, M., 2010a. Watershed cuts:

- Thinnings, shortest path forests, and topological watersheds. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, 925–39.
- Cousty, J., Couprivé, M., Najman, L., Bertrand, G., 2008b. Weighted fusion graphs: Merging properties and watersheds. *Discrete Applied Mathematics* 156, 3011–27.
- Cousty, J., Najman, L., 2011. Incremental algorithm for hierarchical minimum spanning forests and saliency of watershed cuts, in: Soille, P., Pesaresi, M., Ouzounis, G.K. (Eds.), *Mathematical Morphology and Its Applications to Image and Signal Processing*. Springer. volume 6671 of *Lecture Notes in Computer Science*, pp. 272–83.
- Cousty, J., Najman, L., Couprivé, M., Clément-Guinaudeau, S., Goissen, T., Garot, J., 2010b. Segmentation of 4d cardiac MRI: Automated method based on spatio-temporal watershed cuts. *Image and Vision Computing* 28, 1229–43.
- Cousty, J., Najman, L., Dias, F., Serra, J., 2013a. Morphological filtering on graphs. *Computer Vision and Image Understanding* 117, 370–85.
- Cousty, J., Najman, L., Perret, B., 2013b. Constructive links between some morphological hierarchies on edge-weighted graphs, in: *Mathematical Morphology and Its Applications to Signal and Image Processing (ISMM 2013)*. Springer Berlin Heidelberg, pp. 86–97.
- Cousty, J., Najman, L., Serra, J., 2008c. Raising in watershed lattices, in: *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, IEEE. pp. 2196–9.
- Cousty, J., Najman, L., Serra, J., 2009b. Some morphological operators in graph spaces, in: Wilkinson, M.H., Roerdink, J.B. (Eds.), *Mathematical Morphology and Its Application to Signal and Image Processing*. Springer Berlin Heidelberg. volume 5720 of *Lecture Notes in Computer Science*, pp. 149–60.
- Curic, V., Luengo Hendriks, C., Borgerfors, G., 2012. Salience adaptive structuring elements. *IEEE Journal of Selected Topics in Signal Processing* 6, 809–19.
- Desbrun, M., Hirani, A.N., Leok, M., Marsden, J.E., 2005. Discrete exterior calculus. Technical Report arXiv preprint math/0508341. CalTech, Applied Geometry lab.
- Dias, F., Cousty, J., Najman, L., 2011. Some morphological operators on simplicial complex spaces, in: Deblie-Rennesson, I., Domenjoud, E., Kerautret, B., Even, P. (Eds.), *Discrete Geometry for Computer Imagery*. Springer Berlin Heidelberg. volume 6607 of *Lecture Notes in Computer Science*, pp. 441–52.
- Diestel, R., 1997. *Graph Theory*. Graduate Texts in Mathematics, Springer.
- Dijkstra, E., 1959. A note on two problems in connexion with graphs. *Numerische Mathematik* 1, 269–71.
- Drakopoulos, K., Maragos, P., 2012. Active contours on graphs: Multiscale morphology and graphcuts. *IEEE Journal of Selected Topics in Signal Processing* 6, 780–94.
- Edelsbrunner, H., Letscher, D., Zomorodian, A., 2000. Topological persistence and simplification, in: *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, IEEE. pp. 454–63.
- Edelsbrunner, H., Mücke, E.P., 1994. Three-dimensional alpha shapes. *ACM Transactions on Graphics* 13, 43–72.
- Fabbri, R., Costa, L.D.F., Torelli, J.C., Bruno, O.M., 2008. 2D Euclidean distance transform algorithms: A comparative survey. *ACM Computing Surveys (CSUR)* 40, 2.
- Falcão, A.X., Bergo, F.P., 2004. Interactive volume segmentation with differential image foresting transforms. *IEEE Transactions on Medical Imaging* 23, 1100–8.
- Falcão, A.X., Stolfi, J., de Alencar Lotufo, R., 2004. The image foresting transform: Theory, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26, 19–29.
- Falcão, A.X., Udupa, J.K., Samarasekera, S., Sharma, S., Hirsch, B.E., Lotufo, R.d.A., 1998. User-steered image segmentation paradigms: Live wire and live lane. *Graphical Models and Image Processing* 60, 233–60.
- Felzenszwalb, P.F., Huttenlocher, D.P., 2004. Efficient graph-based image segmentation. *International Journal of Computer Vision* 59, 167–81.
- Ford, L., Fulkerson, D., 1962. *Flows in networks*. Princeton University Press (Princeton, NJ).
- Forman, R., 2002. A user's guide to discrete Morse theory. *Sém. Lothar. Combin* 48, B48c.
- Géraud, T., 2014. The Olena Image Processing Platform. <http://olena.1rde.epita.fr/>. Accessed: 2014-04-28.
- Géraud, T., Carlinet, E., Crozet, S., Najman, L., 2013. A quasi-linear algorithm to compute the tree of shapes of nD images, in: Luengo Hendriks, C., Borgerfors, G., Strand, R. (Eds.), *Mathematical Morphology and Its Applications to Signal and Image Processing*. Springer. volume 7883 of *Lecture Notes in Computer Science*, pp. 98–110.
- Grady, L., 2006. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 1768–83.
- Grady, L.J., Polimeni, J.R., 2010. *Discrete Calculus: Applied Analysis on Graphs for Computational Science*. Springer.
- Grimaud, M., 1992. New measure of contrast: the dynamics, in: San Diego'92, International Society for Optics and Photonics. pp. 292–305.
- Guigues, L., Cocquerez, J.P., Men, H.L., 2006. Scale-sets image analysis. *International Journal of Computer Vision* 68, 289–317.
- Guigues, L., Le Men, H., Cocquerez, J.P., 2003. The hierarchy of the cocoons of a graph and its application to image segmentation. *Pattern Recognition Letters* 24, 1059–66.
- Guimarães, S.J.F., Cousty, J., Kenmochi, Y., Najman, L., 2012. A hierarchical image segmentation algorithm based on an observation scale, in: Gimel'farb, G.L., Hancock, E., Imiya, A. (Eds.), *Structural, Syntactic, and Statistical Pattern Recognition*. Springer. pp. 116–25.
- Havel, J., Merciol, F., Lefèvre, S., 2013. Efficient schemes for computing α -tree representations, in: Luengo Hendriks, C., Borgerfors, G., Strand, R. (Eds.), *Mathematical Morphology and Its Applications to Signal and Image Processing (ISMM 2013)*. Springer Berlin Heidelberg. volume 7883 of *Lecture Notes in Computer Science*, pp. 111–22.
- Haxhimusa, Y., Kropatsch, W., 2004. Segmentation graph hierarchies, in: Fred, A., Caelli, T., Duin, R.P.W., Campilho, A., de Ridder, D. (Eds.), *Structural, Syntactic, and Statistical Pattern Recognition*. Springer. pp. 343–51.
- Heijmans, H., 1991. Theoretical aspects of gray-level morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13, 568–82.
- Heijmans, H., 1994. *Morphological Image Operators*. Advances in Electronics and Electron Physics Series, Academic Press, Boston.
- Heijmans, H., Buckley, M., Talbot, H., 2005. Path openings and closings. *Journal of Mathematical Imaging and Vision* 22, 107–19.
- Heijmans, H., Nacken, P., Toet, A., Vincent, L., 1992. Graph morphology. *Journal of Visual Communication and Image Representation* 3, 24 – 38.
- Heijmans, H., Vincent, L., 1992. Graph morphology in image analysis, in: Dougherty, E. (Ed.), *Mathematical Morphology in Image Processing*, Marcel-Dekker. pp. 171–203.
- Heijmans, H.J., 1999. Connected morphological operators for binary images. *Computer Vision and Image Understanding* 73, 99–120.
- Heijmans, H.J.A.M., Ronse, C., 1990. The algebraic basis of mathematical morphology. I Dilations and erosions. *Computer Vision, Graphics, and Image Processing* 50, 245–95.
- Jones, R., 1999. Connected filtering and segmentation using component trees. *Computer Vision and Image Understanding* 75, 215–28.
- Khalimsky, E., Kopperman, R., Meyer, P.R., 1990. Computer graphics and connected topologies on finite ordered sets. *Topology and its Applications* 36, 1 – 17.
- Kiran, B.R., Serra, J., et al., 2014. Global-local optimizations on hierarchies of segmentations. *Pattern Recognition* 47, 12–24.
- Kong, T., Rosenfeld, A., 1989. Digital topology: Introduction and survey. *Computer Vision, Graphics, and Image Processing* 48, 357–93.
- Kruskal, J.B., 1956. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society* 7, 48–50.
- Lebenberg, J., Buvat, I., Lalonde, A., Clarysse, P., Casta, C., Cochet, A., Constantinides, C., Cousty, J., De Cesare, A., Jehan-Besson, S., et al., 2012. Nonsupervised ranking of different segmentation approaches: Application to the estimation of the left ventricular ejection fraction from cardiac cine MRI sequences. *IEEE Transactions on Medical Imaging* 31, 1651–60.
- Lerallut, R., Decencière, E., Meyer, F., 2005. Image filtering using morphological amoebas, in: Ronse, C., Najman, L., Decencière, E. (Eds.), *Mathematical Morphology 40 Years on, ISMM 2005*. Springer. pp. 13–22.
- Lerallut, R., Decencière, E., Meyer, F., 2007. Image filtering using morphological amoebas. *Image and Vision Computing* 25, 395–404.
- Levillain, R., Géraud, T., Najman, L., 2009. Milena: Write generic morphological algorithms once, run on many kinds of images, in: Wilkinson, M.H.F., Roerdink, J.B.T.M. (Eds.), *Mathematical Morphology and Its Application to Signal and Image Processing*. Springer. volume 5720 of *Lecture Notes in Computer Science*, pp. 295–306.
- Lézoray, O., Grady, L. (Eds.), 2012. *Image Processing and Analysis with Graphs: Theory and Practice*. Digital Imaging and Computer Vision, CRC Press / Taylor and Francis.

- Loménie, N., Racoceanu, D., 2012. Point set morphological filtering and semantic spatial configuration modeling: Application to microscopic image and bio-structure analysis. *Pattern Recognition* 45, 2894–911.
- Loménie, N., Stamon, G., 2008. Morphological mesh filtering and α -objects. *Pattern Recognition Letters* 29, 1571–9.
- Lum, P., Singh, G., Lehman, A., Ishkanov, T., Vejdemo-Johansson, M., Alagappan, M., Carlsson, J., Carlsson, G., 2013. Extracting insights from the shape of complex data using topology. *Nature Scientific reports* 3, 1236–.
- Malmberg, F., Luengo Hendriks, C., 2014. An efficient algorithm for exact evaluation of stochastic watersheds. *Pattern Recognition Letters* .
- Mangan, A.P., Whitaker, R.T., 1999. Partitioning 3D surface meshes using watershed segmentation. *IEEE Transactions on Visualization and Computer Graphics* 5, 308–21.
- Maragos, P., Schafer, R., 1987. Morphological filters—Part I: Their set-theoretic analysis and relations to linear shift-invariant filters. *IEEE Trans. on Acoustics, Speech and Signal Processing* 35, 1153–69.
- Matheron, G., 1975. Random Sets and Integral Geometry. John Wiley & Sons, New York.
- Meyer, F., 1991. Un algorithme optimal de ligne de partage des eaux, in: Procs. of 8ème Congrès RFIA, AFCET, Lyon-Villeurbanne, France, pp. 847–59.
- Meyer, F., 1994. Minimum spanning forests for morphological segmentation, in: Mathematical Morphology and its Applications to Image Processing. Springer, pp. 77–84.
- Meyer, F., Angulo, J., 2007. Micro-viscous morphological operators, in: Banon, G.J.F., Barrera, J., Braga-Neto, U.d.M. (Eds.), Mathematical Morphology and its Application to Signal and Image Processing (ISMM 2007), INPE/MCT, pp. 165–76.
- Meyer, F., Beucher, S., 1990. Morphological segmentation. *Journal of Visual Communication and Image Representation* 1, 21–46.
- Meyer, F., Lerallut, R., 2007. Morphological operators for flooding, leveling and filtering images using graphs, in: Graph-based Representations in Pattern Recognition (GbRPR'07), pp. 158–67.
- Meyer, F., Maragos, P., 1999. Morphological scale-space representation with levelings, in: Nielsen, M., Johansen, P., Olsen, O., Weickert, J. (Eds.), Scale-Space Theories in Computer Vision. Springer Berlin Heidelberg, volume 1682 of *Lecture Notes in Computer Science*, pp. 187–98.
- Meyer, F., Najman, L., 2010. Segmentation, minimum spanning tree and hierarchies, in: Najman, L., Talbot, H. (Eds.), Mathematical Morphology: from Theory to Applications. ISTE / J. Wiley & Sons, pp. 229–61.
- Meyer, F., Stawiaski, J., 2010. A stochastic evaluation of the contour strength, in: Pattern Recognition. Springer, pp. 513–22.
- Miranda, P., Mansilla, L., 2014. Oriented image foresting transform segmentation by seed competition. *IEEE Transactions on Image Processing* 23, 1057–7149.
- Monasse, P., Guichard, F., 2000. Fast computation of a contrast-invariant image representation. *IEEE Transactions on Image Processing* 9, 860–72.
- Morard, V., 2012. Détection de structures fines par traitement d’images et apprentissage statistique : application au contrôle non destructif. Ph.D. thesis. Mines ParisTech.
- Morard, V., Dokládal, P., Decencière, E., 2014. Parsimonious path openings and closings. *IEEE Transactions on Image Processing* , To appear.
- Morris, O.J., Lee, M.d.J., Constantinides, A.G., 1986. Graph theory for image analysis: an approach based on the shortest spanning tree. *IEE Proc. on Communications, Radar and Signal* 133, 146–52.
- Nagao, M., Matsuyama, T., Ikeda, Y., 1979. Region extraction and shape analysis in aerial photographs. *Computer Graphics and Image Processing* 10, 195–223.
- Najman, L., 2011. On the equivalence between hierarchical segmentations and ultrametric watersheds. *Journal of Mathematical Imaging and Vision* 40, 231–47.
- Najman, L., Couplie, M., 2006. Building the component tree in quasi-linear time. *IEEE Transactions on Image Processing* 15, 3531–9.
- Najman, L., Couplie, M., Bertrand, G., 2005. Watersheds, mosaics, and the emergence paradigm. *Discrete Applied Mathematics* 147, 301–24.
- Najman, L., Cousty, J., Perret, B., 2013. Playing with Kruskal: algorithms for morphological trees in edge-weighted graphs, in: Luengo Hendriks, C., Borgefors, G., Strand, R. (Eds.), Mathematical Morphology and Its Applications to Signal and Image Processing (ISMM 2013). Springer Berlin Heidelberg, volume 7883 of *Lecture Notes in Computer Science*, pp. 135–46.
- Najman, L., Géraud, T., 2013. Discrete set-valued continuity and interpolation, in: Luengo Hendriks, C., Borgefors, G., Strand, R. (Eds.), Mathematical Morphology and Its Applications to Signal and Image Processing. Springer Berlin Heidelberg, volume 7883 of *Lecture Notes in Computer Science*, pp. 37–48.
- Najman, L., Meyer, F., 2012. A short tour of mathematical morphology on edge and vertex weighted graphs, in: Lezoray, O., Grady, L. (Eds.), *Image Processing and Analysis with Graphs: Theory and Practice*. CRC Press. Digital Imaging and Computer Vision, pp. 141–74.
- Najman, L., Schmitt, M., 1996. Geodesic saliency of watershed contours and hierarchical segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18, 1163–73.
- Najman, L., Talbot, H. (Eds.), 2010. *Mathematical Morphology: from Theory to Applications*. ISTE-Wiley.
- Nešetřil, J., Milková, E., Nešetřilová, H., 2001. Otakar Borůvka on Minimum Spanning Tree problem: Translation of both the 1926 papers, comments, history. *DMATH: Discrete Mathematics* 233, 3–36.
- Ouzounis, G.K., Wilkinson, M.H., 2007. Mask-based second-generation connectivity and attribute filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 990–1004.
- Papa, J.P., Falcão, A.X., de Albuquerque, V.H.C., Tavares, J.M.R., 2012. Efficient supervised optimum-path forest classification for large datasets. *Pattern Recognition* 45, 512–20.
- Paris, S., Durand, F., 2007. A topological approach to hierarchical segmentation using mean shift, in: *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, IEEE, pp. 1–8.
- Pavlidis, 1977. *Structural Pattern Recognition*. Springer, Berlin.
- Perret, B., Cousty, J., Tankyevych, O., Talbot, H., Passat, N., 2013. Directed connected operators: asymmetric hierarchies for image filtering and segmentation. Technical Report. Laboratoire d’Informatique Gaspard-Monge - LIGM , Laboratoire Image, Signaux et Systèmes Intelligents - LISSI - EA 3956 , Centre de Recherche en Sciences et Technologies de l’Information et de la Communication - CRESTIC. URL: <http://hal.archives-ouvertes.fr/hal-00869727>. submitted.
- Peyré, G., Pechaud, M., Keriven, R., Cohen, L., 2010. Geodesic methods in computer vision and graphics. *Foundations and Trends in Computer Graphics and Vision* 5, 197–397.
- Philipp-Foliguet, S., Jordan, M., Najman, L., Cousty, J., 2011. Artwork 3D model database indexing and classification. *Pattern Recognition* 44, 588–97.
- Pollack, M., 1960. The maximum capacity through a network. *Operations Research* 8, 733–6.
- Prim, R.C., 1957. Shortest connection networks and some generalizations. *Bell System Technical Journal* 36, 1389–401.
- Purkait, P., Chanda, B., 2012. Super resolution image reconstruction through Bregman iteration using morphologic regularization. *IEEE Transactions on Image Processing* 21, 4029–39.
- Rittner, L., Lotufo, R., 2008. Diffusion tensor imaging segmentation by watershed transform on tensorial morphological gradient, in: *Computer Graphics and Image Processing, 2008. SIBGRAPI'08. XXI Brazilian Symposium on*, IEEE, pp. 196–203.
- Ronse, C., 1998. Set-theoretical algebraic approaches to connectivity in continuous or digital spaces. *Journal of Mathematical Imaging and Vision* 8, 41–58.
- Ronse, C., 2006. Flat morphology on power lattices. *Journal of Mathematical Imaging and Vision* 26, 185–216.
- Ronse, C., 2008. Partial partitions, partial connections and connective segmentation. *Journal of Mathematical Imaging and Vision* 32, 97–125.
- Ronse, C., 2014. Axiomatics for oriented connectivity. *Pattern Recognition Letters* , To appear – DOI 10.1016/j.patrec.2014.03.020.
- Ronse, C., Serra, J., 2010. Algebraic foundations of morphology, in: Najman, L., Talbot, H. (Eds.), *Mathematical Morphology: From Theory to Applications*. ISTE / J. Wiley & Sons, pp. 35–80.
- Rosenfeld, A., Pfaltz, J.L., 1968. Distance functions on digital pictures. *Pattern Recognition* 1, 33–61.
- Rouchdy, Y., Cohen, L.D., 2008. Image segmentation by geodesic voting. Application to the extraction of tree structures from confocal microscope images, in: *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, IEEE, pp. 1–5.
- Saha, P.K., Udupa, J.K., 2001. Relative fuzzy connectedness among multiple objects: theory, algorithms, and applications in image segmentation. *Computer Vision and Image Understanding* 82, 42–56.
- Salembier, P., 2010. Connected operators based on tree pruning strategies, in: Najman, L., Talbot, H. (Eds.), *Mathematical Morphology:from Theory to Applications*. ISTE-Wiley, pp. 179–98.

- Salembier, P., Garrido, L., 2000. Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. *IEEE Transactions on Image Processing* 9, 561–76.
- Salembier, P., Oliveras, A., Garrido, L., 1998. Anti-extensive connected operators for image and sequence processing. *IEEE Transactions on Image Processing* 7, 555–70.
- Salembier, P., Serra, J., 1995. Flat zones filtering, connected operators, and filters by reconstruction. *IEEE Transactions on Image Processing* 4, 1153–60.
- Salembier, P., Wilkinson, M.H., 2009. Connected operators. *Signal processing magazine, IEEE* 26, 136–57.
- Serra, J., 1982. *Image Analysis and Mathematical Morphology*. volume 1. Academic Press.
- Serra, J., 1988. *Image Analysis and Mathematical Morphology, Volume 2: Theoretical Advances*. Academic Press.
- Serra, J., 2006. A lattice approach to image segmentation. *Journal of Mathematical Imaging and Vision* 24, 83–130.
- Serra, J., Vincent, L., 1992. An overview of morphological filtering. *Circuits, Systems and Signal Processing* 11, 47–108.
- Shuman, D.I., Narang, S.K., Frossard, P., Ortega, A., Vandergheynst, P., 2013. *The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains*. *Signal Processing Magazine, IEEE* 30, 83–98.
- Soille, P., 1999. *Morphological Image Analysis*. Springer-Verlag.
- Soille, P., 2008. Constrained connectivity for hierarchical image partitioning and simplification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 1132–45.
- Stawiaski, J., Meyer, F., 2009. Minimum spanning tree adaptive image filtering, in: *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pp. 2245–8.
- Stelldinger, P., Strand, R., 2006. Topology preserving digitization with FCC and BCC grids, in: Reulke, R., Eckardt, U., Flach, B., Knauer, U., Polthier, K. (Eds.), *Combinatorial Image Analysis, 11th International Workshop, IWCIA 2006*. Springer, pp. 226–40.
- Straehle, C., Peter, S., Köthe, U., Hamprecht, F.A., 2013. K-smallest spanning tree segmentations, in: Weickert, J., Hein, M., Schiele, B. (Eds.), *Pattern Recognition*. Springer Berlin Heidelberg. volume 8142 of *Lecture Notes in Computer Science*, pp. 375–84.
- Ta, V.T., Elmoataz, A., Lézoray, O., 2011. Nonlocal PDEs-based morphology on weighted graphs for image and data processing. *IEEE Transactions on Image Processing* 20, 1504–16.
- Talbot, H., Appleton, B., 2007. Efficient complete and incomplete path openings and closings. *Image and Vision Computing* 25, 416–25.
- Takyevych, O., 2010. Filtering of thin objects: applications to vascular image analysis. Ph.D. thesis. Université Paris-Est.
- Takyevych, O., Talbot, H., Passat, N., 2013. Semi-connections and hierarchies, in: Luengo Hendriks, C., Borgefors, G., Strand, R. (Eds.), *Mathematical Morphology and Its Applications to Signal and Image Processing (ISMM 2013)*. Springer. volume 7883 of *Lecture Notes in Computer Science*, pp. 159–70.
- Udupa, J.K., Samarasekera, S., 1996. Fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. *Graphical Models and Image Processing* 58, 246–61.
- Urbach, E.R., Roerdink, J.B., Wilkinson, M.H., 2007. Connected shape-size pattern spectra for rotation and scale-invariant classification of gray-scale images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 272–85.
- Vachier, C., Meyer, F., 1995. Extinction value: a new measurement of persistence, in: *IEEE Workshop on nonlinear signal and image processing*, pp. 254–7.
- Valero, S., Chanussot, J., Benediktsson, J.A., Talbot, H., Waske, B., 2010. Advanced directional mathematical morphology for the detection of the road network in very high resolution remote sensing images. *Pattern Recognition Letters* 31, 1120–7.
- Vincent, L., 1989. Graphs and mathematical morphology. *Signal Processing* 16, 365–88.
- Vincent, L., 1994. Morphological area openings and closings for grey-scale images, in: Ying-Lie, O., Toet, A., Foster, D., Heijmans, H.J.A.M., Meer, P. (Eds.), *Shape in Picture*. Springer. volume 126 of *NATO ASI*, pp. 197–208.
- Vincent, L., 1998. Minimal path algorithms for the robust detection of linear features in gray images. *Computational Imaging And Vision* 12, 331–8.
- Vincent, L., Soille, P., 1991. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13, 583–98.
- Wendt, P., Coyle, E., Gallagher, N., J., 1986. Stack filters. *IEEE Trans. on Acoustics, Speech and Signal Processing* 34, 898 – 911.
- Westenberg, M.A., Roerdink, J.B., Wilkinson, M.H., 2007. Volumetric attribute filtering and interactive visualization using the max-tree representation. *IEEE Transactions on Image Processing* 16, 2943–52.
- Wilkinson, M.H.F., 2011. A fast component-tree algorithm for high dynamic-range images and second generation connectivity, in: Macq, B., Schelkens, P. (Eds.), *ICIP, IEEE*. pp. 1021–4.
- Xu, Y., 2013. Tree-based shape spaces for applications in image processing and computer vision. Ph.D. thesis. Université Paris-Est.
- Xu, Y., Géraud, T., Najman, L., 2012. Morphological filtering in shape spaces: Applications using tree-based image representations, in: *Pattern Recognition (ICPR), 21st International Conference on, IEEE*. pp. 485–8.
- Xu, Y., Géraud, T., Najman, L., 2013. Two applications of shape-based morphology: blood vessels segmentation and a generalization of constrained connectivity, in: Luengo Hendriks, C., Borgefors, G., Strand, R. (Eds.), *Mathematical Morphology and Its Applications to Signal and Image Processing*. Springer. volume 7883 of *Lecture Notes in Computer Science*, pp. 390–401.
- Zahn, C., 1971. Graph-theoretical methods for detecting and describing Gestalt clusters. *IEEE Transactions on Computers C-20*, 99–112.