J Math Imaging Vis (2009) 35: 128–142
DOI 10.1007/s10851-009-0159-9

Links Between Image Segmentation Based on Optimum-Path Forest and Minimum Cut in Graph

Paulo A.V. Miranda · Alexandre X. Falcão

Published online: 16 June 2009

Abstract Image segmentation can be elegantly solved by optimum-path forest and minimum cut in graph. Given that both approaches exploit similar image graphs, some comparative analysis is expected between them. We clarify their differences and provide their comparative analysis from the theoretical point of view, for the case of binary segmentation (object/background) in which hard constraints (seeds) are provided interactively. Particularly, we formally prove that some optimum-path forest methods from two distinct region-based segmentation paradigms, with internal and external seeds and with only internal seeds, indeed minimize some graph-cut measures. This leads to a proof of the necessary conditions under which the optimum-path forest algorithm and the min-cut/max-flow algorithm produce exactly the same segmentation result, allowing a comparative analysis between them.

Keywords Image segmentation based on optimum-path forest · Graph-cut segmentation · Image foresting transform · Graph-based image segmentation

1 Introduction

Discrete Mathematics provides an elegant framework for image processing, rich of efficient algorithms with proofs of correctness. As a consequence, many image segmentation methods have been modeled as graph-search problems.

P.A.V. Miranda · A.X. Falcão (✉)
LIV, Institute of Computing, UNICAMP, C.P. 6176, 13083-970, Campinas, SP, Brazil
e-mail: afalcao@ic.unicamp.br

P.A.V. Miranda
e-mail: paulo.miranda@ic.unicamp.br

Two popular approaches exploit undirected and weighted image graphs, where the pixels are the nodes, the arcs are defined by an adjacency relation, and the arc weights are similarity values computed based on image properties. The first approach minimizes a functional of the arc weights leading to a cut in the graph that separates object and background [9, 10, 41, 49]. The second approach [21], called image foresting transform (IFT), can reduce segmentation to the computation of an *optimum-path forest* according to a *connectivity function*, which assigns a value to any path in the graph, including *trivial* paths formed by a single node. That is, considering the maximum value among all possible paths with terminus at each node, the optimum path is trivial for some nodes, called *roots*, and the remaining nodes will have an optimum path coming from their most strongly connected root, partitioning the graph into an optimum-path forest (disjoint sets of optimum-path trees). Two distinct region-based segmentation paradigms, with internal and external seeds (i.e., roots by imposition) and with only internal seeds, can be solved by IFT. In the first paradigm, internal and external seeds compete with each other for their most strongly connected pixels, such that the image is partitioned into two optimum-path forests—one rooted at the internal seeds, defining the object, and the other rooted at the external seeds, representing the background [16, 30]. The second paradigm solves segmentation by computing one optimum-path forest from only internal seeds and applying some other criterion to cut optimum paths such that the remaining forest defines the object [6, 20, 22, 31]. Indeed, both paradigms can be easily extended to multiple objects, but we will focus on binary image segmentation (object/background) with interactive seed selection. Some variants have also shown the importance of a hybrid paradigm [31].

These approaches based on optimum-path forest and minimum cut in graph are usually regarded to as unrelated.

Recently, some links between them were clarified for a particular case, in which an increasing transformation is applied to all arc weights [1]. The present work advances the state of the art in graph-based image segmentation by better clarifying the relation between these approaches. We theoretically prove that some IFT-based methods from both paradigms, with internal and external seeds and with only internal seeds, indeed minimize some graph-cut measures. These IFT-based methods are closely related to popular segmentation approaches, such as *absolute-fuzzy connectedness* (AFC) [43], *relative-fuzzy connectedness* (RFC) [39, 45], *iterative relative-fuzzy connectedness* (IRFC) [13] and *watershed transforms from markers* (WT) [7]. We clarify their differences and the advantages of the IFT-based approach. These methods have been successfully used in many applications [16, 25, 28, 34, 35, 42, 44], which validates the importance of these theoretical results. In view of that, we also extend the theorem stated in [1] by establishing the necessary conditions to its converse. The results provide better understanding of the methods and help the selection of the best algorithm for a given application.

Section 2 presents the basic notions on image graphs and Sect. 3 presents the concepts about the IFT, which will be used for image segmentation based on optimum-path forest in Sect. 4. The approach based on minimum cut in graph is briefly described in Sect. 5. Sections 6, 7 and 8 present the theorems and their corresponding proofs. Taking into account the theoretical results, some comparative analysis involving methods from both approaches is carried out along the text, but in Sect. 9 the analysis takes into account only IFT with internal and external seed competition and the min-cut/max-flow segmentation. Our conclusions are stated in Sect. 10.

2 Basic Concepts on Image Graphs

A multi-dimensional and multi-spectral image $\hat{I}$ is a pair $(\mathcal{I}, \mathbf{I})$ where $\mathcal{I} \subset Z^n$ is the image domain and $\mathbf{I}(t)$ assigns a set of m scalars $I_i(t)$, $i = 1, 2, \ldots, m$, to each pixel $t \in \mathcal{I}$. The subindex i is removed when $m = 1$.

An *adjacency relation* $\mathcal{A}$ is a binary relation on $\mathcal{I}$. We use $t \in \mathcal{A}(s)$ and $(s, t) \in \mathcal{A}$ to indicate that t is adjacent to s. Once the adjacency relation $\mathcal{A}$ has been fixed, the image $\hat{I}$ can be interpreted as a graph $(\mathcal{I}, \mathcal{A})$ whose nodes (or vertices) are the image pixels in $\mathcal{I}$ and whose arcs are the pixel pairs (s, t) in $\mathcal{A}$. We are interested in irreflexive and symmetric relations. For example, one can take $\mathcal{A}$ to consist of all pairs of pixels (s, t) in the Cartesian product $\mathcal{I} \times \mathcal{I}$ such that $d(s, t) \leq \rho$ and $s \neq t$, where $d(s, t)$ denotes the Euclidean distance and ρ is a specified constant (e.g., 4-neighborhood, when $\rho = 1$, and 8-neighborhood, when $\rho = \sqrt{2}$, in case of 2D images).

Each arc $(s, t) \in \mathcal{A}$ has a fixed weight $w(s, t) \geq 0$ which may be computed from local image and object properties extracted from $\mathbf{I}$ and some global information (e.g., markers [33]). Graph-cut segmentation methods [10, 41, 49] usually assign lower weights to arcs across the object's boundary (i.e., an affinity measure between pixels s and t) while some methods based on optimum-path forest [30] work with higher arc weights across the object's boundary (i.e., a dissimilarity measure). However, the latter may be easily adapted to the first scheme [4] and, therefore, lower arc weights across the object's boundary will be considered without loss of generality. For example, one may use the complement of a gradient magnitude (i.e., $I_{max} - \frac{I(s)+I(t)}{2}$ for a gradient image $\hat{I}$ with maximum value I_{max}). In this work we consider only undirected and weighted graphs. That is, the adjacency relation is symmetric and $w(s, t) = w(t, s)$ for all $(s, t) \in \mathcal{A}$.

For a given image graph $(\mathcal{I}, \mathcal{A})$, a path $\pi_t = \langle t_1, t_2, \ldots, t \rangle$ is a sequence of adjacent pixels with terminus at a pixel t. A path is *trivial* when $\pi_t = \langle t \rangle$. A path $\pi_t = \pi_s \cdot \langle s, t \rangle$ indicates the extension of a path π_s by an arc (s, t) (Fig. 1a). All paths considered in this work are simple paths, that is, paths with no repeated vertices (pixels).

A *predecessor map* is a function P that assigns to each pixel t in $\mathcal{I}$ either some other adjacent pixel in $\mathcal{I}$, or a distinctive marker nil not in $\mathcal{I}$—in which case t is said to be a *root* of the map. A *spanning forest* is a predecessor map which contains no cycles—i.e., one which takes every pixel to nil in a finite number of iterations (Figs. 1b and 1c, where $R(\pi_t)$ is a root node and $P(t)$ is the predecessor node of t in the path π_t). For any pixel $t \in \mathcal{I}$, a spanning forest P defines a path π_t recursively as $\langle t \rangle$ if $P(t) = nil$, and $\pi_s \cdot \langle s, t \rangle$ if $P(t) = s \neq nil$.

3 Image Foresting Transform (IFT)

A *connectivity function* computes a value $f(\pi_t)$ for any path π_t, usually based on arc weights. Let $\Pi(\mathcal{I}, \mathcal{A}, t)$ be the set of all paths in the graph $(\mathcal{I}, \mathcal{A})$ with terminus at t. In this work, a path is *optimum* according to the following definition.

Definition 1 (Optimum path) A path π_t is *optimum* if $f(\pi_t) \geq f(\tau_t)$ for any other path $\tau_t \in \Pi(\mathcal{I}, \mathcal{A}, t)$.

By taking to each pixel $t \in \mathcal{I}$ one optimum path with terminus t, we obtain the optimum-path value $V(t)$, which is uniquely defined by

$$V(t) = \max_{\forall \pi_t \in \Pi(\mathcal{I}, \mathcal{A}, t)} \{f(\pi_t)\}. \quad (1)$$

The *image foresting transform* (IFT) algorithm solves the above optimization problem by dynamic programming [21].

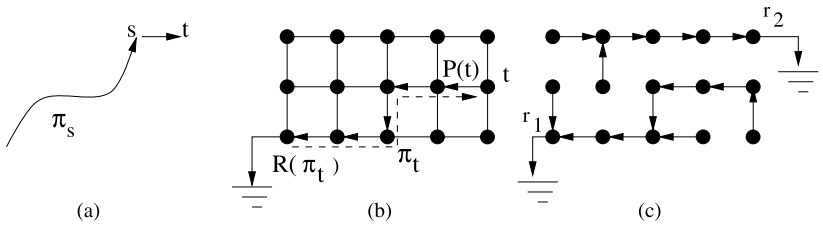

Fig. 1 (**a**) Path $\pi_t = \pi_s \cdot \langle s,t \rangle$ indicates the extension of path π_s by an arc (s,t). (**b**) A 4-neighborhood graph showing a path π_t (*dashed line*) represented in backwards, where $P(t)$ is the predecessor node of t and $R(\pi_t)$ is the root pixel. (**c**) A spanning forest P with two root nodes, r_1 and r_2

The IFT takes an image $\hat{I}$, a path-value function f and an adjacency relation $\mathcal{A}$; and assigns one optimum path π_t to every pixel $t \in \mathcal{I}$ such that an *optimum-path forest* P is obtained—i.e., a spanning forest where all paths are optimum. However, f must be *smooth*, that is, satisfy Definition 2, as demonstrated in [21]. The attributes of the forest include the map V, the roots $R(\pi_t)$, root labels $L(t)$, and the predecessor $P(t)$ of t in the optimum path. The image operators are then reduced to a local processing of these attributes [21]. If we consider $\leq$ instead of $\geq$ in Definition 1, and minimize $V(t)$ in (1), we obtain a dual definition of optimality. To convert a problem to its dual form, we simply have to invert the sign of the path-value function (i.e., $g(\pi_t) = -f(\pi_t)$). It is important to state that the IFT was originally presented in this equivalent dual form [21]. In this paper we consider the first schema (Definition 1 and (1)).

Definition 2 (Smooth path-value function) A path-value function f is *smooth* if for any pixel $t \in \mathcal{I}$, there is an optimum path π_t which either is trivial, or has the form $\tau_s \cdot \langle s,t \rangle$ where

(C1) $f(\tau_s) \geq f(\pi_t)$,
(C2) τ_s is optimum,
(C3) for any optimum path τ'_s, $f(\tau'_s \cdot \langle s,t \rangle) = f(\pi_t)$.

An interesting property of an optimum-path forest is that any path starting in a root node is also a complete optimum path (path-value function must be *smooth*), according to the following definition.

Definition 3 (Complete optimum path) A path $\pi_{t_n} = \langle t_1, t_2, \ldots, t_n \rangle$ is *complete optimum* if all paths $\pi_{t_i} = \langle t_1, t_2, \ldots, t_i \rangle$, $i = 1, 2, \ldots, n$ are optimum paths.

Note that, the applications for the image foresting transform (IFT) go beyond region-based image segmentation [17–19]. In this work, however, we are only interested in region-based image segmentation by IFT. It is also important to state that although we are restricting our analysis to undirected graphs, the IFT can also handle segmentation using directed graphs [21].

4 Region-based Segmentation Using IFT

The segmentation of an image is represented by a labeled image $\hat{L} = (\mathcal{I}, L)$, where $L(t) = 1$ for object pixels and $L(t) = 0$ for background pixels. Each segmentation defines an *induced cut boundary* $\mathcal{C}$ in the graph.

Definition 4 (Induced cut boundary) The segmentation given by a labeled image $\hat{L}$ defines an *induced cut boundary* in the graph, which is the set $\mathcal{C}$ of arcs (s,t) such that $L(s) = 1$ and $L(t) = 0$.

We consider image segmentation from two seed sets, $\mathcal{S}_1$ and $\mathcal{S}_0$ ($\mathcal{S}_1 \cap \mathcal{S}_0 = \emptyset$), containing pixels interactively selected inside and outside the object, respectively. A feasible segmentation must satisfy these sets of hard constraints.

Definition 5 (Feasible segmentation) The segmentation given by a labeled image $\hat{L}$ is *feasible* if $L(t) = 1$ for all $t \in \mathcal{S}_1$ and $L(t) = 0$ for all $t \in \mathcal{S}_0$.

We are interested in the particular case of smooth path-value functions, the monotonically decremental path-value function $f_{\min}$. This function basically assigns to any path π_t the minimum arc-weight along π_t. Equation (2) presents it in the recursive form.

$$f_{\min}(\langle t \rangle) = \begin{cases} w_{\max} + 1 & \text{if } t \in \mathcal{S}_1 \cup \mathcal{S}_0 \\ -\infty & \text{otherwise} \end{cases}$$
$$f_{\min}(\pi_s \cdot \langle s,t \rangle) = \min\{f_{\min}(\pi_s), w(s,t)\}, \tag{2}$$

where $w_{\max}$ represents the maximum arc-weight in the graph and the search for optimum paths is constrained to start in $\mathcal{S}_1 \cup \mathcal{S}_0$ (roots by imposition).

There are basically two distinct region-based segmentation paradigms by optimum-path forest, with internal $\mathcal{S}_1$ and external $\mathcal{S}_0$ seeds and with only internal seeds $\mathcal{S}_1$ (i.e., $\mathcal{S}_0 = \emptyset$).

4.1 IFT with Only Internal Seeds

This region-based segmentation paradigm solves segmentation by computing one optimum-path forest from only in-

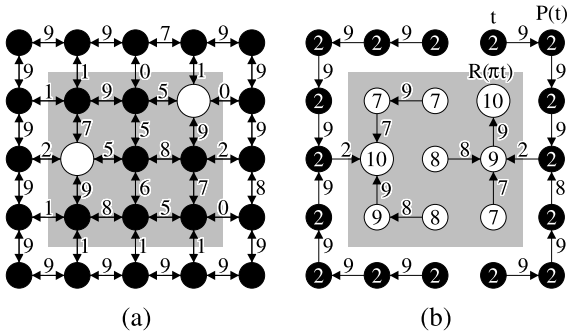

Fig. 2 (**a**) A 4-neighborhood graph, where the numbers indicate the arc weights and the object is the *shaded square*. Two seeds are selected inside the object (*bigger white dots*). (**b**) An optimum-path forest for the path-value function defined via (2). The numbers inside the nodes indicate the values of the optimum paths (1). The object is obtained as the *white dots* after using a threshold $\kappa = 3$

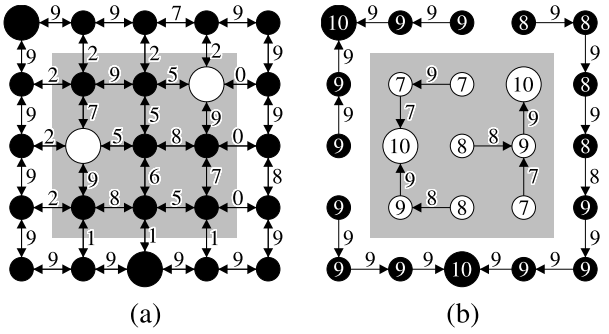

Fig. 3 (**a**) A 4-neighborhood graph, where the numbers indicate the arc weights and the object is the *shaded square*. Four seeds are selected, two are inside the object (*white dots*) and two are in the background (*bigger black dots*). (**b**) An optimum-path forest for the path-value function defined by (2). The numbers inside the nodes indicate the values of the optimum paths (1). The label $L(s) = 0$ (*black*), or $L(s) = 1$ (*white*) of each seed s is propagated to all pixels within its respective optimum-path tree

ternal seeds and applying some pruning criterion to cut optimum paths such that the remaining forest defines the object [6, 20, 22, 31]. We focus on a particular method of this paradigm denoted by IFT-CT (*IFT segmentation by Connectivity Threshold*), which is related to the classical method called *absolute-fuzzy connectedness* (AFC) [43].

In IFT-CT, seeds are specified inside the object and the optimum-path value from the seed set is computed to each pixel, such that the object is obtained by thresholding the resulting connectivity map $V(t)$ (1). The resulting segmentation is defined as the maximal subset of $\mathcal{I}$, leading to a feasible segmentation (Definition 5 with $\mathcal{S}_0 = \emptyset$), wherein all pixels t are reached by optimum paths whose values $V(t)$ are greater than or equal to a given threshold κ (see Fig. 2).

In AFC [43], the graph is implicitly defined by arcs with fixed weights computed by a fuzzy affinity relation, which encodes both the adjacency relation $\mathcal{A}$ and the arc weights $w(s,t)$. While in IFT-CT, the adjacency relation and arc weights are defined separately. For multiple seeds, a single linear-time execution of the IFT-CT is enough to compute the connectivity map $V(t)$. However, given that AFC does not allow connectivity through zero-weighted arcs, their results may differ. Now, if we allow different arc weights for each seed, function $f_{\min}$ is no longer *smooth* (Definition 2) [21]. In this case, the object must be defined as the union of all individual IFT-CT segmentation results, computed for each seed separately [31].

Several other methods exist [6, 20, 22, 31] with different pruning criteria, aiming to improve IFT-CT. In [31], a variant is proposed with multiple κ automatic thresholds leading to greater flexibility. Given that, the IFT algorithm computes optimum paths progressively from the seeds by extending optimum paths already computed [21], we have an ordered region growing from the internal seeds. The analysis of the induced cut boundaries along this ordered region growing is exploited as pruning criterion in [22], with the advantage of being independent of parameters. However, this region growing of the IFT from internal seeds may invade the background (i.e., the leaking problem) before filling the entire object, causing the methods [22] and IFT-CT to fail and requiring more seeds for correction. With regard to this issue, the methods [6, 20] are more robust. In these latter methods, a combinatorial property of the forest is exploited to automatically identify the *leaking pixels* (boundary parts which are crossed by optimum paths) and eliminate their subtrees, such that the remaining forest defines the object [6, 20].

4.2 IFT with Internal and External Seeds

The method IFT-SC (*IFT segmentation by Seed Competition*) consists of the computation of an optimum-path forest P using internal and external seeds, $\mathcal{S}_1$ and $\mathcal{S}_0$. Its segmentation $\hat{L}$ is defined as follows, where π_t is the optimum path with terminus t obtained from P.

$$L(t) = \begin{cases} 1 & \text{if } R(\pi_t) \in \mathcal{S}_1, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

From an *optimum-path forest* we obtain an image partition, where each seed is root of an *optimum-path tree* composed by pixels more strongly connected to that seed than to any other, and ties are broken by some tie-breaking policy as discussed next. For a given optimum-path forest, the object is defined as the union of all optimum-path trees rooted at the internal seeds (see Fig. 3).

As observed in [21], the optimum-path forest may not be unique. For example, if all paths have the same value, then any spanning forest will be optimum. Since path values are usually discrete, multiple solutions are common when the arc weights are defined in a short range (Figs. 4a–c). Ties between paths π_t and τ_t from seeds $s_1 = R(\pi_t)$ and

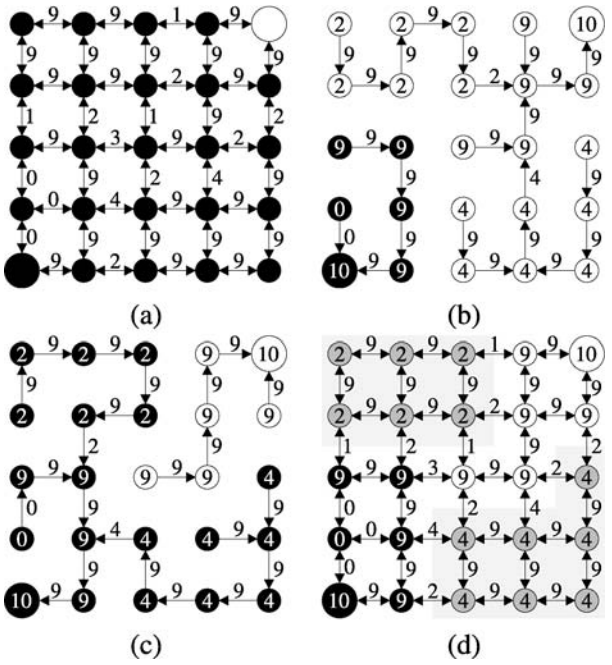

Fig. 4 (**a**) A 4-neighborhood graph with one object seed (*white dot*) and one background seed (*bigger black dot*). (**b**, **c**) Two possible optimum-path forests using (2). The numbers inside the nodes indicate the values of the optimum paths (1). (**d**) Two tie zones having values 2 and 4 are shown in *gray*. The pixel (0, 3) with optimum value 0 is not a tie-zone pixel, since it is unambiguously surrounded by a black influence zone. Note that, any path leading to this node from the *white dot* also has value 0, but these paths are disregarded since they are not complete optimum

$s_2 = R(\tau_t)$ with the same label ($\{s_1, s_2\} \subset \mathcal{S}_1$ or $\{s_1, s_2\} \subset \mathcal{S}_0$) are never a problem, since they lead to exactly the same final segmentation result $\hat{L}$. Hence, any solution in this case is satisfactory. However, a special care has to be taken in the case of seeds with different labels, which constitute the basis of the real *tie zones* as follows.

Definition 6 (Tie-zone pixel) A pixel t is a *tie-zone pixel* if there exist two complete optimum paths π_t and τ_t such that $R(\pi_t) \in \mathcal{S}_0$ and $R(\tau_t) \in \mathcal{S}_1$.

The *tie zones* are informally defined as maximal connected components of *tie-zone pixels*, while some works just consider the union of all *tie-zone pixels* [2]. The following more precise and formal definition of *tie zone* for path-value function $f_{\min}$ will be adopted through the rest of this paper.

Definition 7 (Tie zone) A *tie zone* is a maximal set $\mathcal{T}$ of *tie-zone pixels*, which forms a subtree in some optimum-path forest.

Figure 4d shows an example of *tie zones*, where the *tie zone pixels* are highlighted in gray. Every node t in an optimum-path forest can be seen as the root of a subtree, which is composed by all nodes that are reached by optimum paths that pass through t in this optimum-path forest. Consider the upper left node as coordinate (0, 0) in Fig. 4a such that x-coordinates increase from left to right; and y-coordinates increase from top to bottom. In Fig. 4b, the nodes (2, 1) and (3, 3) are roots of optimum-path subtrees composed by *tie-zone pixels*. Therefore, we have two *tie zones* in Fig. 4d. In this work we assume optimum-path forests such that each tie zone, that may be reached from distinct labels (in different optimum-path forests), will receive just one of the labels. This may be accomplished either by using a *LIFO* tie-breaking policy [21], or simply by assigning a fixed label (1 or 0) to all tie zones (Figs. 4b–c). Thus, when the partition of ambiguous regions is important, this should be treated by means of a better arc-weight estimation.

The IFT-SC formulation [16, 30] captures the essential features of the *watershed transform from markers* (WT) [7], although there is no unique and precise definition for a watershed transform in the literature [37]. Indeed, it was proven that the tie zones of the IFT-SC include all solutions predicted by many discrete definitions of WT [3]. An extensive discussion of the theoretical relations among several different types of watersheds, and also their relation to minimum spanning forests can be found in [14].

The method IFT-SC is also closely related to relative-fuzzy connectedness (RFC) [39, 45]. However, in RFC the tie zones are left unassigned to either background or foreground. Apart from this way to handle ties in RFC, the main difference between these approaches resides in the fact that in RFC the optimum-path values (defined as *strength of connectedness* in [45]) from the internal and external seeds are computed independently for each seed. Hence, it is not possible to guarantee that the computed paths from one seed are complete optimum (Definition 3) in relation to the other seeds (i.e., in RFC an optimum path from a seed s_1 to a target pixel t may pass through a region having pixels more strongly connected to some other seed s_2). This significantly increases the number of ties, causing holes within the objects, and the iterative relative-fuzzy connectedness (IRFC) [13] was proposed to circumvent this problem. It is basically an iterative refinement strategy that imposes additional constraints based on the results from previous iterations. What it essentially does is to penalize paths that are not complete optimum in relation to the other seeds. Thus, disregarding the divergences in the treatment of ties, IFT-SC and IRFC should give similar segmentation results, but being the IFT-SC simpler and faster due to the simultaneous label propagation from all seeds.

In fact, the main motivation for IFT-SC was to eliminate the choice of κ in IFT-CT, favoring the simultaneous segmentation of multiple objects. Actually RFC can be viewed as AFC with automatically calculated thresholds, one for

each object [12]. On the other hand, some variants have shown the importance of a hybrid IFT-based method which includes competition among internal and external seeds, and simultaneous automatic computation of multiple κ values per object [31]. The IFT can also extend these methods, by using a more general definition for strength of connectedness based on *smooth* path-value functions [21] (e.g., additive function [5, 36]) or even based on *non-smooth* path-value functions (e.g., non-fixed arc weights [26]). However, in the last case, the IFT results into a spanning forest which may not be optimal [21].

5 Graph-cut Segmentation

Approaches for graph-cut segmentation are based on objective functions that measure some global property of the object's boundary from the arc-weight assignment (Sect. 2). The idea is to assign weights to the arcs such that the minimum of this objective function (a graph-cut measure) corresponds to the desired segmentation (i.e., a *cut boundary* whose arcs connect the nodes between object and background).

Wu and Leahy [50] were the first to introduce a solution for graph cut using as measure the sum of the arc weights in the cut boundary (4). Their cut measure has the bias toward small boundaries and other objective functions, such as *average cut* [15], *mean cut* [48], *average association* [40], *normalized cut* [41], *ratio cut* [49], and *energy functions* [9, 10, 27] have been proposed to circumvent this problem.

$$E_1(\hat{L}) = \sum_{\forall (s,t)\in\mathcal{A} \mid L(s)=1, L(t)=0} w(s,t) \tag{4}$$

Unfortunately, the problem of segmenting a desired object in a given image cannot be simply reduced to finding a minimum of an objective function in the entire search space, since false-cut boundaries due to similarities between object and background are very common in practice. Indeed it was verified that even in a reduced search space that includes the desired cut from the user's point of view, it does not always correspond to the minimum cut [22].

In view of this, two different strategies have been proposed in the literature. One produces a hierarchical partition tree by recursively applying an unsupervised graph-cut algorithm inside each partition obtained from the previous iteration and then delegating the object recognition for a high-level method that analyzes this tree in a second step [41, 49]. The second strategy incorporates hard constraints (seed pixels) to reduce the search space [9, 10]. Although the second strategy is mostly used in interactive segmentation, where the hard constraints are provided by mouse clicks and drags, the hard constraints may be implemented by probabilistic models (e.g., brain atlases in MRI [25, 29]) in order to achieve automatic segmentation. Hence, the difference between these strategies, when used for single object segmentation, is basically when the recognition is made: *a posteriori* or *a priori*.

One of the main problems in the first strategy is that the minimum cut in a generic graph is NP-hard, when we consider the entire search space [41, 49]. Heuristic solutions still present poor computational performance [24] and their results are sometimes far from the desired one [11]. The second strategy can benefit from the rich set of fast recognition approaches available in the literature [32, 47, 51], which give the object's approximate whereabouts and location in the image, providing the hard constraints, and avoiding this expansive tree computation. But on the other hand, the results may be heavily affected by premature errors during this automatic setting of hard constraints. In fact, Sects. 6 and 7 prove that IFT-SC and IFT-CT (Sect. 4) are indeed graph-cut approaches under the second strategy.

In the second strategy, we have graph-cut methods [9, 10] that extend the work of Wu and Leahy [50] by adding two terminal nodes (*source* and *sink*) to the image graph, which represent object and background, respectively. These nodes are directly connected to all pixels by arcs whose weights reflect penalties for assigning a pixel to object and background based on region properties (*probability maps*). A min-cut/max-flow algorithm from source to sink [23] is then used to compute the minimum-cut boundary according to the following equation:

$$E_2(\hat{L}) = E_1(\hat{L}) + \lambda \left(\sum_{\forall s\in\mathcal{I} \mid L(s)=1} \overline{\mathcal{P}_o}(s) + \sum_{\forall t\in\mathcal{I} \mid L(t)=0} \overline{\mathcal{P}_b}(t) \right) \tag{5}$$

where $\overline{\mathcal{P}_o}(s)$ and $\overline{\mathcal{P}_b}(t)$ are the complement of the probability maps $\mathcal{P}_o(s)$ and $\mathcal{P}_b(t)$ that measure how well the intensities of pixels s and t fit into a known intensity model (i.e., histogram) of the object and background, respectively.

If the method fails in detecting the desired boundary, the user can impose the arc weights with source and sink by selecting seed pixels inside and outside the object [10]. The running time of these algorithms is still polynomial [9] (i.e., typically $O(mn^2)$ where m is the number of arcs and n is the number of *nodes*). It also has an ad-hoc parameter $\lambda \geq 0$ which specifies the relative importance of the region properties term versus the boundary properties term. If λ is low their cut becomes the same as in Wu and Leahy [50], which has a bias toward small boundaries, and if λ is high the method becomes very dependent on the probability maps (Figs. 5b, c). Moreover, the obtained segmentation (Fig. 5d) is not guaranteed to be connected with the seed sets in the

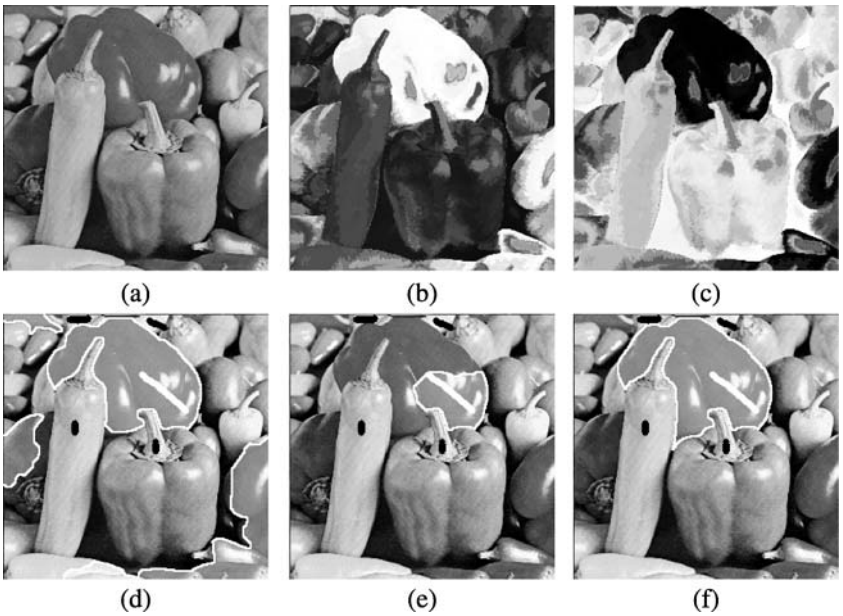

Fig. 5 (**a**) An image of peppers. (**b**, **c**) The probability maps $\mathcal{P}_o$ and $\mathcal{P}_b$ for the bigger pepper at the top. (**d**) The segmentation by graph-cut using (5) with $\lambda = 2$. (**e**, **f**) Graph-cut results using (6) with increasing power values, 2 and 5, respectively

image space, whenever object and background present regions with similar features (Fig. 5a). Therefore, this approach seems to be more useful in applications where reliable probability maps are available [38].

Other possible solution is to raise the arc weights to the power of n (6), considering $\tilde{w}(s,t) = [w(s,t)]^n$ instead of $w(s,t)$ in (4). As we increase the power value n, larger boundaries are favored (Figs. 5e, f), avoiding the need to compute the probability maps.

$$\tilde{E}_1(\hat{L}) = \sum_{\forall (s,t)\in\mathcal{A} \mid L(s)=1, L(t)=0} [w(s,t)]^n = \sum_{\forall (s,t)\in\mathcal{A} \mid L(s)=1, L(t)=0} \tilde{w}(s,t) \tag{6}$$

Therefore, from the discussion above, we have two possible solutions to fix the undesirable bias of E_1. One is to consider E_2 by changing the graph topology at the price of losing the connectivity notion in the original graph $(\mathcal{I}, \mathcal{A})$, which makes the method to behave like a threshold depending on λ [8]; and the second is to penalize arcs with high weights $w(s,t)$ by applying some increasing transformation, as the power of n in $\tilde{E}_1$ or as the exponential used in [29], but conserving the topology of the graph $(\mathcal{I}, \mathcal{A})$. In fact, it will be proven in Sect. 8 the formal conditions under which optimum-path forest by IFT-SC and graph-cut segmentation by (6) produce the same results. Next, Sects. 6 and 7 show that the methods IFT-SC and IFT-CT based on optimum-path forest are indeed graph-cut approaches, each with its own graph-cut measure.

6 IFT-SC as a Graph-cut Approach

The theorems assume an undirected graph, with fixed arc weights (Sect. 2) and will be proven for object/background segmentation.

From an optimum-path forest we obtain an image partition in two disjoint sets with distinct labels. Let $\mathcal{C}_{sc}$ be any cut boundary induced by an IFT-SC segmentation $\hat{L}$ with a single label for each tie zone. For any arc $(a,b) \in \mathcal{C}_{sc}$, at least one of the following inequalities is true with the left-hand side being strictly lower than the right-hand side (Fig. 6).

$$f_{\min}(\pi_a \cdot \langle a,b\rangle) < f_{\min}(\pi_b) \tag{7}$$

$$f_{\min}(\pi_b \cdot \langle b,a\rangle) < f_{\min}(\pi_a) \tag{8}$$

This is a consequence of path optimality (1) for $f_{\min}$ (2) when a single label is assigned to each tie zone. If all tie zones are assigned to the object then (7) holds. If all tie zones are assigned to the background then (8) holds. In the case of *LIFO* tie-breaking policy, at least one of these inequalities will be true for all arcs in the cut.

For example, assume that all tie zones were labeled to the background. If the extension of path π_b by the arc (b,a) has higher value than path π_a, then this path π_a is not optimum. Otherwise if the extension of π_b has the same value of path π_a, then pixel a is in a tie zone and should not be part of the object. Therefore, (8) is the only valid configuration left.

In fact we may conclude even more and consider the equations below instead of (7), (8):

$$w(a,b) < f_{\min}(\pi_b) \quad \text{and} \quad w(a,b) \le f_{\min}(\pi_a) \tag{9}$$

$$w(a,b) < f_{\min}(\pi_a) \quad \text{and} \quad w(a,b) \le f_{\min}(\pi_b) \tag{10}$$

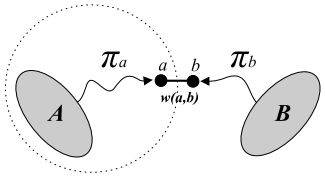

Fig. 6 The *dashed line* represents a segmentation by an optimum-path forest with the arc (a, b) in its cut boundary. The sets of nodes A and B represents internal and external seeds respectively

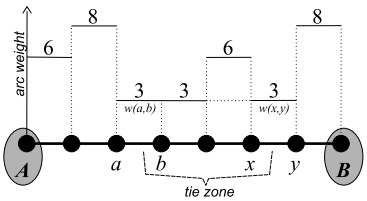

Fig. 7 A 1D example. All pixels in the tie zone are reached by paths from A and B with the same optimum value 3. If this tie zone is labeled to the background (set B) then a is reached from A with $f_{\min}(\pi_a) = 6$ and b is reached from B with $f_{\min}(\pi_b) = 3$. Note that $w(a, b) = 3$ and (10) is valid ($3 < 6$ and $3 \leq 3$)

When all tie zones are labeled to the background, the first inequality in (10) can be proved by contradiction as follows. If $w(a, b) \geq f_{\min}(\pi_a)$ then $f_{\min}(\pi_a \cdot \langle a, b\rangle) = f_{\min}(\pi_a)$. There are two possibilities, the extension of path π_a by arc (a, b) has value $f_{\min}(\pi_a) > f_{\min}(\pi_b)$, or it has value $f_{\min}(\pi_a) \leq f_{\min}(\pi_b)$. If the first case is true, then π_b is not optimum leading to a contradiction. In the second case, we have that $f_{\min}(\pi_b \cdot \langle b, a\rangle) = \min\{f_{\min}(\pi_b), w(b, a)\} \geq f_{\min}(\pi_a)$, since $w(a, b) \geq f_{\min}(\pi_a)$ and $f_{\min}(\pi_a) \leq f_{\min}(\pi_b)$, but this is invalid according to (8). Therefore, $w(a, b) < f_{\min}(\pi_a)$ is the only valid configuration. The second inequality in (10) may also be proved by contradiction. If $w(a, b) > f_{\min}(\pi_b)$ then $f_{\min}(\pi_b \cdot \langle b, a\rangle) = f_{\min}(\pi_b)$, and a should belong to the background, otherwise $f_{\min}(\pi_a) > f_{\min}(\pi_b)$ and b could not belong to the background since $f_{\min}(\pi_a \cdot \langle a, b\rangle) > f_{\min}(\pi_b)$. But a in the background leads to a contradiction proving the second inequality given in (10). See Fig. 7 for an example in 1D.

Equation (9) can be proven in a similar way when all tie zones are labeled to the object.

Theorem 1 (Optimum-path forest cut in IFT-SC) *Any segmentation $\hat{L}$ defined by an optimum-path forest with path-value function $f_{\min}$ and with a single label value for each tie zone (Definition 7) minimizes the graph-cut measure E_3 defined by (11) among all possible segmentation results (Definition 5).*

$$E_3(\hat{L}) = \max_{\forall (s,t)\in\mathcal{A} |\ L(s)=1, L(t)=0} w(s, t) \tag{11}$$

Proof We will prove the theorem in the case when all tie zones are labeled to the background, the other case having essentially identical proof. Let $E_{\min}$ be the minimum value of E_3 defined by (11) among all segmentation results satisfying the sets of hard constraints $\mathcal{S}_1$ and $\mathcal{S}_0$ (Definition 5). Let $\mathcal{C}_{\text{sc}}$ be the cut boundary induced (Definition 4) by a segmentation $\hat{L}$ obtained through IFT-SC with path-value function $f_{\min}$ and let (a, b) be the arc with maximum weight $w(a, b)$ in $\mathcal{C}_{\text{sc}}$ (that is, $E_3(\hat{L}) = w(a, b)$). For any path π_t, let $Arc(\pi_t)$ be the set of all arcs within π_t, or the empty set if π_t is a trivial path.

Any optimum cut $\mathcal{C}_{\min}$, which minimizes the graph-cut measure E_3 defined by (11) among all possible feasible segmentations, must contain at least one arc from the set $Arc(\pi_a) \cup \{(a, b)\} \cup Arc(\pi_b)$ in order to disconnect object and background seeds (Fig. 6). There are three possibilities:

1. $\mathcal{C}_{\min} \cap Arc(\pi_a)$ is not empty.
2. $\mathcal{C}_{\min} \cap Arc(\pi_b)$ is not empty.
3. $(a, b) \in \mathcal{C}_{\min}$.

In case 1, the path π_a will have one arc $(x, y) \in \mathcal{C}_{\min}$. From (11) we may conclude that $w(x, y) \leq E_{\min}$ and also that $f_{\min}(\pi_a) \leq E_{\min}$ by the definition of $f_{\min}$ (2). Putting everything together with (10) we have $w(a, b) < f_{\min}(\pi_a) \leq E_{\min}$, which implies that $E_3(\hat{L}) < E_{\min}$. But this is not possible according to the definition of $E_{\min}$.

In case 2, the path π_b will have one arc in $\mathcal{C}_{\min}$ and therefore we may conclude, in a way analogous to what was done in case 1, that $f_{\min}(\pi_b) \leq E_{\min}$.

$$f_{\min}(\pi_b) \leq E_{\min} \tag{12}$$

Assuming that Theorem 1 is false we have:

$$w(a, b) > E_{\min} \tag{13}$$

By combining the above hypothesis with (10), we have $f_{\min}(\pi_a) > E_{\min}$. Since $w(a, b) > E_{\min}$ and $f_{\min}(\pi_a) > E_{\min}$, from the definition of $f_{\min}$ (2), it is also easy to see that:

$$f_{\min}(\pi_a \cdot \langle a, b\rangle) = \min\{f_{\min}(\pi_a), w(a, b)\} > E_{\min} \tag{14}$$

From (12) we finally obtain:

$$f_{\min}(\pi_a \cdot \langle a, b\rangle) > E_{\min} \geq f_{\min}(\pi_b) \tag{15}$$

But from these results, we reach a contradiction because π_b will not be an optimum path if (15) holds. Therefore, hypothesis (13) with case 2 is false.

In case 3, from $(a, b) \in \mathcal{C}_{\min}$ we have that $w(a, b) \leq E_{\min}$ (11). But $w(a, b) < E_{\min}$ is not possible according to the definition of $E_{\min}$. Hence, we have that $w(a, b) = E_{\min}$ as we wanted to prove (Theorem 1). □

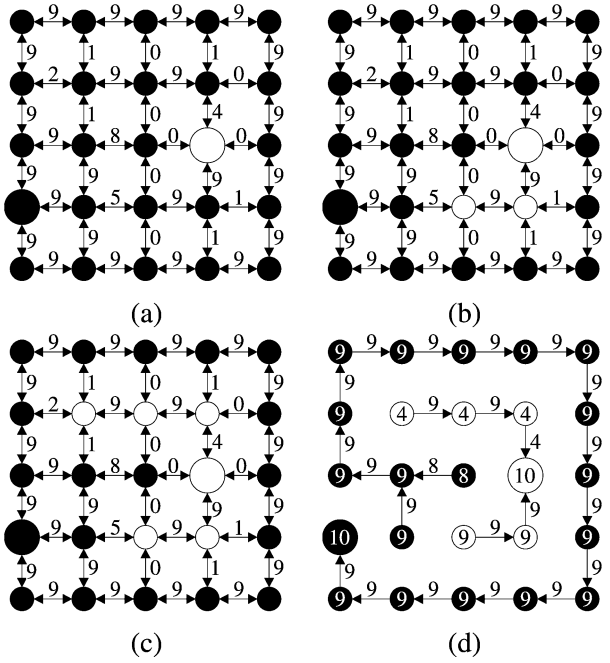

Fig. 8 (**a**) A 4-neighborhood graph with one internal seed (*white dot*) and one external seed (*bigger black dot*). (**b**, **c**) Two optimum segmentation candidates for E_3 (11) satisfying the hard constraints. (**d**) The chosen solution by optimum-path forest using (2), where the numbers inside the nodes indicate the values of the optimum paths (1)

For $\hat{L}$ to be a segmentation by IFT-SC under the stated conditions, Theorem 1 is necessary but not sufficient. Figures 8a–d illustrate this aspect. Figure 8a shows an image graph with two seed pixels. Figures 8b and 8c show two possible segmentation results, satisfying these hard constraints. Both have the same optimum cut $E_3(\hat{L}) = 5$ (11), but only Fig. 8c corresponds to a segmentation by IFT-SC (Fig. 8d). Fortunately, there is another cut property that gives a stronger characterization of an optimum-path forest segmentation by IFT-SC.

Theorem 2 (Piecewise optimum property in IFT-SC) *Let $\mathcal{C}_{\text{sc}}$ be any cut boundary induced by a segmentation $\hat{L}$ defined by an optimum-path forest with path-value function $f_{\min}$ and with a single label for each tie zone* (*Definition* 7). *Let E_3 be also defined for a set of arcs $\mathcal{X}$ as $E_3(\mathcal{X}) = \max_{\forall(s,t)\in\mathcal{X}}\{w(s,t)\}$. For any subset $\mathcal{C}_{\text{sub}} \subset \mathcal{C}_{\text{sc}}$, let $\overline{\mathcal{C}}_{\text{sub}}$ be defined as $\mathcal{C}_{\text{sc}} \backslash \mathcal{C}_{\text{sub}}$. Any non-empty subset $\mathcal{C}_{\text{sub}} \subset \mathcal{C}_{\text{sc}}$ minimizes $E_3(\mathcal{X})$ among all possible set of arcs $\mathcal{X}$ whose union $\mathcal{X} \cup \overline{\mathcal{C}}_{\text{sub}}$ defines an induced cut boundary of a feasible segmentation* (*Definition* 5).

For example in Fig. 8, if we consider all arcs in the cut except the arc with weight 5 as being the subset, then the next greatest arc has value 4 in Fig. 8b and value 2 in Fig. 8c. A valid IFT-SC segmentation will be one with the lower value (Fig. 8d).

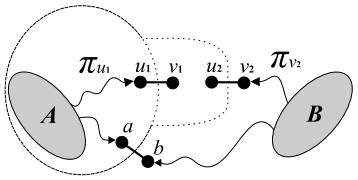

Fig. 9 Two possible optimum cut boundaries are shown, both with optimum value in (11) given by $w(a,b)$. The *dashed lines* represent the boundary pieces at issue, with greatest weights given by $w(u_1, v_1)$ and $w(u_2, v_2)$ respectively

Proof We will prove the theorem in the case when all tie zones are labeled to the background, the other case having essentially identical proof. Let $\overline{\mathcal{C}}_{\text{sub}}$ be a subset of the arcs of $\mathcal{C}_{\text{sc}}$. Let's consider two arbitrary sets of arcs $\mathcal{X}_1$ and $\mathcal{X}_2$ such that $\mathcal{X}_i \cup \overline{\mathcal{C}}_{\text{sub}}$ defines a feasible segmentation (Definition 5), for $i = 1, 2$. Let the arcs $(u_1, v_1) \in \mathcal{X}_1$ and $(u_2, v_2) \in \mathcal{X}_2$ be arcs with maximum weight within these sets (Fig. 9). Theorem 2 states that a set $\mathcal{X}_i$ with lower maximum-weight should be selected. There are two cases, $w(u_1, v_1) < w(u_2, v_2)$ or $w(u_1, v_1) > w(u_2, v_2)$.

In the first case we have that $w(u_1, v_1) < w(u_2, v_2)$ and according to the theorem $\mathcal{X}_1$ should be picked. Let's prove by contradiction and assume that $\mathcal{X}_2$ is the optimum-path forest result by IFT-SC. From (10) we know that the optimum path to pixel u_2 from the internal seeds satisfies $w(u_2, v_2) < f_{\min}(\pi_{u_2})$ and from the hypothesis $w(u_1, v_1) < w(u_2, v_2)$, we may conclude that $w(u_1, v_1) < f_{\min}(\pi_{u_2})$. But π_{u_2} must pass through $\mathcal{X}_1$ which implies a contradiction, since $f_{\min}(\pi_{u_2}) \leq w(u_1, v_1)$ by (2).

Similarly, in the second case we have that $w(u_1, v_1) > w(u_2, v_2)$ and according to the theorem $\mathcal{X}_2$ should be picked. By assuming its logical negation we get a contradiction like before by similar arguments. Hence, Theorem 2 holds as we wanted to prove. □

It is important to state that some graph-cut measures, such as the mean cut [48] (16), are not piecewise optimum (see Fig. 10). In these measures, changes in one part of the boundary have a global effect, changing the understanding of what is best in other parts. This instability is undesirable in interactive segmentation, because the user loses control over segmentation when local interventions for correction may modify other parts where the user was already satisfied with the segmentation results. Therefore, the piecewise optimum property is indeed very important.

$$E_4(\hat{L}) = \frac{E_1(\hat{L})}{\sum_{\forall(s,t)\in\mathcal{A}|\ L(s)=1, L(t)=0} 1} \tag{16}$$

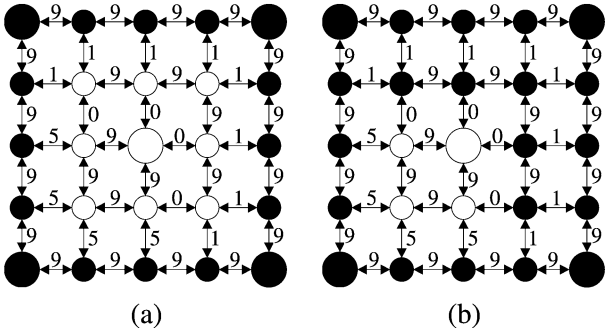

Fig. 10 Two possible segmentation results on the same 4-neighborhood graph, both satisfying the hard constraints (*bigger white* and *black dots*). (**a**) The optimum solution with minimum mean cut $\frac{(4\times5+8\times1)}{12} = 2.33$. (**b**) The second solution has worse mean cut $\frac{(4\times5+4\times0)}{8} = 2.50$. Since $\frac{4\times0}{4} < \frac{8\times1}{8}$ we may conclude that this graph-cut measure is not piecewise optimum

7 IFT-CT as a Graph-cut Approach

For a given image graph $(\mathcal{I}, \mathcal{A})$, the segmentation $\hat{L}$ obtained by the IFT-CT with a connectivity threshold κ, is feasible (Definition 5) with respect to the internal seeds in $\mathcal{S}_1$, and its induced cut $\mathcal{C}_{\text{ct}}$ has the following property:

$$E_3(\hat{L}) < \kappa \tag{17}$$

Equation (17) states that all arcs in the cut $\mathcal{C}_{\text{ct}}$ have values lower than κ. From the definition of IFT-CT, we have that $f_{\min}(\pi_a) \geq \kappa$ for any interior pixel a and optimum path π_a. If we assume that the above property (17) is false, then we have at least one arc $(a, b) \in \mathcal{C}_{\text{ct}}$ such that $w(a, b) \geq \kappa$. From the definition of $f_{\min}$ (2) we have that:

$$f_{\min}(\pi_a \cdot \langle a, b\rangle) = \min\{f_{\min}(\pi_a), w(a, b)\} \tag{18}$$

Since $f_{\min}(\pi_a) \geq \kappa$ and $w(a, b) \geq \kappa$, the above equation implies that $f_{\min}(\pi_a \cdot \langle a, b\rangle) \geq \kappa$. But from the definition of IFT-CT, this also implies that pixel b must belong to the object leading to a contradiction. Therefore, the property given in (17) is true.

Figure 11 shows that there may be more than one feasible segmentation (Definition 5) with induced cut boundary satisfying this property. Both results in Figs. 11b and 11d have $E_3(\hat{L}) = 3$ which is lower than $\kappa = 4$. In fact, the IFT-CT segmentation $\hat{L}$ is always given by the smallest region possessing the property given in (17) (Fig. 11b). To understand this, note that, the IFT-CT segmentation may be obtained by removing from the original image graph $(\mathcal{I}, \mathcal{A})$ all arcs whose weights are lower than κ, and by taking then the maximal connected components in the resulting graph, such that they contain at least one seed pixel. This can be accomplished by a simple breadth-first search from the internal seeds in the remaining graph. Since any cut satisfying property (17) will have all of its arcs removed, the breadth-first search will stop when the first cut with this property

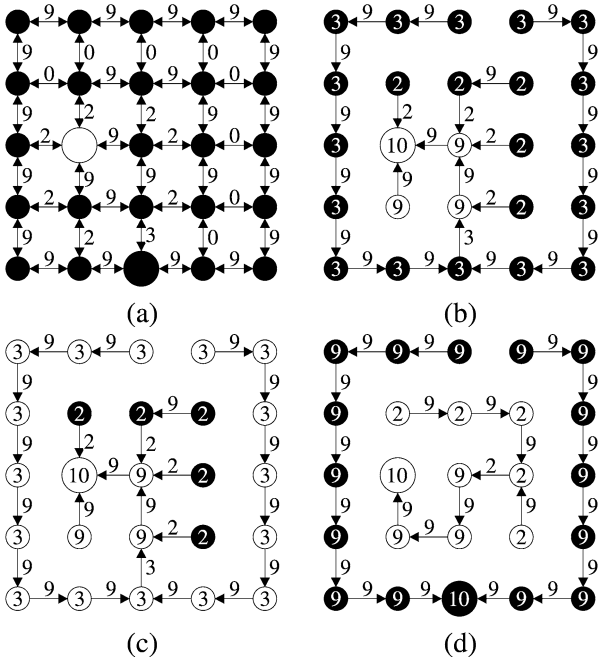

Fig. 11 (**a**) A 4-neighborhood graph with one internal seed (*white dot*) and one external seed (*bigger black dot*). (**b**, **c**) The IFT-CT results from the internal seed with $\kappa = 4$ and $\kappa = 3$, respectively. Note that, $\kappa = 4$ gives the IFT-CT result with lowest E_3 value satisfying all initial hard constraints ($E_3(\hat{L}) = 3$). (**d**) The result by IFT-SC also has $E_3(\hat{L}) = 3$. However, it is quite different from the IFT-CT due to the piecewise optimum property (Theorem 2)

is found. In fact, the implementation which thresholds the connectivity map $V(t)$ is more suitable for interactive segmentation, because the user may change the value of κ with real-time response. That is, the map $V(t)$ encodes all possible IFT-CT results for any value of κ (Figs. 11b, c).

Therefore, the cut $\mathcal{C}_{\text{ct}}$ can be finally defined using the characterization from Theorem 3.

Theorem 3 (Optimum-path forest cut in IFT-CT) *The segmentation $\hat{L}$ defined by the IFT-CT in a graph with fixed arc weights and using path-value function $f_{\min}$ minimizes the graph-cut measure E_5 defined by* (19) *among all segmentation results satisfying the internal hard constraints in $\mathcal{S}_1$* (*Definition 5 with $\mathcal{S}_0 = \emptyset$*).

$$E_5(\hat{L}) = \sum_{\forall s \in \mathcal{I} |\ L(s)=1} 1 + \mathcal{U}\left(E_3(\hat{L}) - \kappa\right) \cdot N \tag{19}$$

where $\mathcal{U}$ is the *unit step function* ($\mathcal{U}(x) = 1$ if $x \geq 0$, and 0 otherwise) and N is the total number of nodes/pixels. The second term in (19) acts as a penalty which makes impracticable cuts that do not satisfy property (17). Among all possible results satisfying property (17), the first term in (19) guarantees the selection of the one with smallest area.

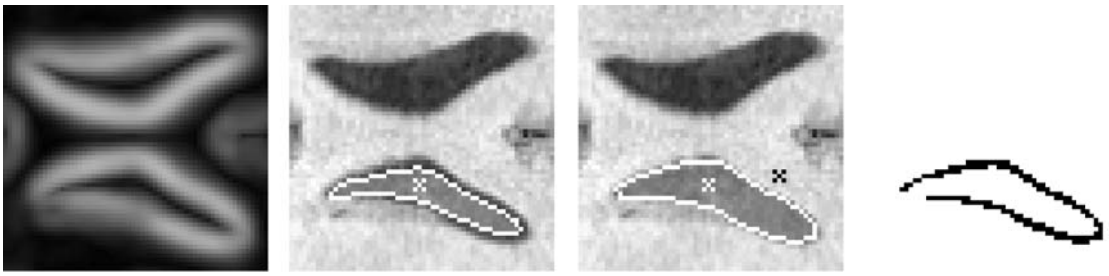

Fig. 12 (**a**) A gradient image of the ventricles with thick edges. (**b**) The result of IFT-CT by the selection of the best κ. (**c**) An improved result is obtained by IFT-SC. (**d**) The XOR operation of the labels (in *black*) indicates a considerable divergence

From Fig. 11 it is clear that, this cut $\mathcal{C}_{\text{ct}}$ does not have the piecewise optimum property with respect to E_3 (Theorem 2), since it always choose the smallest region. In real applications, if the arc weights are computed based on a thick gradient (Fig. 12a), this may result in the lost of a narrow band around the object (Figs. 12b–d). In this sense, it seems that a non-maximal suppression to thin the edges is an adequate preprocessing that should be adopted in IFT-CT. Therefore, the results of IFT-SC are in general different from the ones obtained by IFT-CT for any threshold κ (Figs. 11b–d and Figs. 12b, c). Another clear difference is regarding to the presence of holes inside the objects, which may appear in IFT-CT.

8 The Link with the Min-cut/Max-flow Algorithm

A min-cut/max-flow algorithm from source to sink [23], with the internal seeds in $\mathcal{S}_1$ connected to the source and the external seeds in $\mathcal{S}_0$ connected to the sink, computes a cut boundary which minimizes the graph-cut measure E_1 defined by (4) among all possible segmentation results (i.e., method [10] with parameter $\lambda = 0$). The same algorithm can be used to minimize $\tilde{E}_1$ (6).

The graph-cut measures E_1 (4) and $\tilde{E}_1$ (6) naturally lead to optimum solutions, that are also piecewise optimum. That is, for a given feasible segmentation $\hat{L}$, its induced cut boundary $\mathcal{C}$ is optimum, only if any of its subsets is also optimum. For example, for a cut $\mathcal{C}$, let $\mathcal{Y} \subset \mathcal{C}$ be a fixed subset that is known to be part of the optimum solution. Let K be the sum of all arc weights within this fixed set $\mathcal{Y}$. The cut $\mathcal{C}$ will be optimum only if $\mathcal{X} = \mathcal{C} \backslash \mathcal{Y}$ is also optimum:

$$E_{\min} = \min_{\forall \mathcal{C}} \left\{ \sum_{\forall (s,t) \in \mathcal{C}} w(s,t) \right\}$$

$$= \min_{\forall \mathcal{X}} \left[\sum_{\forall (s,t) \in \mathcal{X}} w(s,t) + \sum_{\forall (s,t) \in \mathcal{Y}} w(s,t) \right]$$

$$= \min_{\forall \mathcal{X}} \left\{ \sum_{\forall (s,t) \in \mathcal{X}} w(s,t) \right\} + K$$

Finally, assuming Theorems 1 and 2, the next theorem completes the link between IFT-SC and graph-cut segmentation based on the min-cut/max-flow algorithm, by establishing the necessary conditions under which they produce exactly the same result (i.e., the converse of a theorem presented in [1]). As shown in Sect. 5, the variant $\tilde{E}_1$ can circumvent the undesirable bias of the graph-cut measure E_1 toward small boundaries and avoid the need to compute probability maps (Figs. 5e, f). Indeed, this and other similar variants (increasing transformations) have been used in practice to improve graph-cut segmentation by emphasizing the differences between low and high arc weights [29]. However, the next theorem indicates that such transformations make the graph-cut segmentation to behave like IFT-SC, with more or less intensity, depending on which transformation is used.

Theorem 4 (Equivalence for increasing transformation) *Let $\hat{L}_{\text{sc}}$ be a segmentation by IFT-SC with path-value function $f_{\min}$ and let $\hat{L}_{\text{flow}}$ be a segmentation computed by a min-cut/max-flow algorithm, using the same sets of hard constraints $\mathcal{S}_1$ and $\mathcal{S}_0$ in the graph $(\mathcal{I}, \mathcal{A})$. In the absence of tie zones (Definition 7), there exists a real finite number m such that, for any $n \geq m$, the segmentation results $\hat{L}_{\text{sc}}$ and $\hat{L}_{\text{flow}}$ are exactly the same when $\hat{L}_{\text{flow}}$ is computed by using $\tilde{w}(s,t) = [w(s,t)]^n$ as arc weights.*

Proof By raising arc weights $w(s,t)$ to the power of n ($n > 1$), we obtain new arc weights $\tilde{w}(s,t)$, preserving the order (20) and emphasizing their differences (21).

$$w(s,t) > w(u,v) \quad \Leftrightarrow \quad \tilde{w}(s,t) > \tilde{w}(u,v) \tag{20}$$

$$\frac{w(s,t)}{w(u,v)} = K \quad \Leftrightarrow \quad \frac{\tilde{w}(s,t)}{\tilde{w}(u,v)} = K^n \tag{21}$$

Let $\tilde{E}_3$ be the graph-cut measure E_3 (11) obtained from Theorem 1 but with $\tilde{w}(s,t)$ in the place of $w(s,t)$:

$$\tilde{E}_3(\hat{L}) = \max_{\forall (s,t)\in\mathcal{A} \mid L(s)=1, L(t)=0} [w(s,t)]^n = \max_{\forall (s,t)\in\mathcal{A} \mid L(s)=1, L(t)=0} \tilde{w}(s,t) \quad (22)$$

For any two distinct segmentation results $\hat{L}_1$ and $\hat{L}_2$, it is easy to see that:

$$E_3(\hat{L}_1) < E_3(\hat{L}_2) \quad \Leftrightarrow \quad \tilde{E}_3(\hat{L}_1) < \tilde{E}_3(\hat{L}_2) \quad (23)$$

This is a direct consequence of the order being preserved (20) and, therefore, the optimum-path forest segmentation results with path-value function $f_{\min}$ are not affected by this increasing transformation [1].

As we increase the value of n, the arc-weight ratio changes exponentially (21). Clearly there will be a point ($n \geq m$) where any arc weight $\tilde{w}(s,t)$ will be greater than the sum of all other arc weights $\tilde{w}(u,v)$ having lower values (24) and (25).

$$\frac{\sum_{\forall (u,v) \mid \tilde{w}(u,v) < \tilde{w}(s,t)} \tilde{w}(u,v)}{\tilde{w}(s,t)} = \sum_{\forall (u,v) \mid \tilde{w}(u,v) < \tilde{w}(s,t)} \left[\frac{w(u,v)}{w(s,t)}\right]^n \quad (24)$$

Since $w(u,v)/w(s,t) < 1$ it is easy to see that:

$$\lim_{n\to\infty} \sum_{\forall (u,v) \mid \tilde{w}(u,v) < \tilde{w}(s,t)} \left[\frac{w(u,v)}{w(s,t)}\right]^n = 0 \quad (25)$$

Therefore, for any two distinct segmentation results $\hat{L}_1$ and $\hat{L}_2$, we may conclude an important result when $n \geq m$:

$$\tilde{E}_3(\hat{L}_1) < \tilde{E}_3(\hat{L}_2) \quad \Rightarrow \quad \tilde{E}_1(\hat{L}_1) < \tilde{E}_1(\hat{L}_2) \quad (26)$$

If the left-hand side of (26) is true, then there is an arc in the cut boundary of $\hat{L}_2$ with higher weight than all arcs in the cut of $\hat{L}_1$. This arc alone has greater value than the sum given in $\tilde{E}_1(\hat{L}_1)$ (6). Therefore, disregarding the cases when $\tilde{E}_3(\hat{L}_1) = \tilde{E}_3(\hat{L}_2)$, we may conclude that, under the declared conditions, minimizing $\tilde{E}_3$ or $\tilde{E}_1$ should lead to the same results.

The case $\tilde{E}_3(\hat{L}_1) = \tilde{E}_3(\hat{L}_2)$ always happens in the presence of tie zones. In Fig. 13, suppose that $\mathbf{C_1}$ has $\tilde{E}_3 = \tilde{w}(a_1,b_1)$ and $\mathbf{C_2}$ has $\tilde{E}_3 = \tilde{w}(a_2,b_2)$. If both boundaries have optimum $\tilde{E}_3$ values, then $\tilde{w}(a_1,b_1) = \tilde{w}(a_2,b_2)$ and the entire region comprised in between these two nested boundaries, $\mathbf{C_1}$ and $\mathbf{C_2}$, could be a tie zone. Although, both boundaries are optimum according to $\tilde{E}_3$ (22), in general only one is with respect to $\tilde{E}_1$ (6), depending on the values of the second greatest arcs $\tilde{w}(u_1,v_1)$ and $\tilde{w}(u_2,v_2)$. We

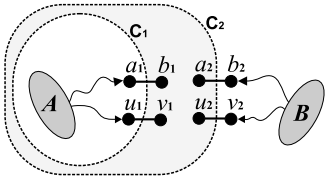

Fig. 13 Two nested cut boundaries $\mathbf{C_1}$ and $\mathbf{C_2}$, with the same optimum cut value $\tilde{E}_3$ (22) are shown. The weights $\tilde{w}(a_1,b_1)$ and $\tilde{w}(a_2,b_2)$ are the maximum within their respective boundaries. The *shaded area* is a tie zone since $\tilde{w}(a_1,b_1) = \tilde{w}(a_2,b_2)$

cannot guarantee that the best regarding to $\tilde{E}_1$ (6) will be chosen by the optimum-path forest algorithm. For example, if all tie zones are assigned to the background, then $\mathbf{C_1}$ will be always chosen. Therefore, the equivalence between these methods cannot be verified in the presence of tie zones. □

Even in the absence of tie zones, the case $\tilde{E}_3(\hat{L}_1) = \tilde{E}_3(\hat{L}_2)$ may still happen as occurred in Fig. 8. However, in this case, the optimum $\tilde{E}_1$ solution is guaranteed by the piecewise optimum property (Theorem 2). For instance, for $n = 2$, Fig. 8c has $\tilde{E}_1 = 34$ and Fig. 8b has $\tilde{E}_1 = 43$. In practice, tie zones are usually represented by a few pixels for suitable arc-weight assignment, which makes this result really relevant.

9 Comparative Analysis Between the Main Paradigms

Several comparisons among methods were already made along the previous sections. In this section, we restrict our attention to IFT-SC and the min-cut/max-flow segmentation. From the theoretical point of view, Theorem 4 states that, in the absence of tie zones, the IFT-SC (11) is a particular case of the min-cut by max-flow (6). Therefore, we have here an important intersection between the frameworks of the IFT and the max-flow approach.

As pointed by Allène et al. [1], the value of power n in (6) acts as a smoothing term. However, in practice, if we decrease n, many important saliences are lost, such as the cow's paws in Fig. 14. Note also that if smooth boundary is an important issue, it always can be done on a second step by a shape filtering [19]. Therefore, the main advantage of $\tilde{E}_1$ over E_3 concerns the E_3 worst case, when the desired cut has some arcs with maximum weight (i.e., gradient with perfect gaps (Fig. 15a)). In $\tilde{E}_1$ these arcs are avoided as much as possible by the selection of shortest cuts within the areas of low image contrast (Fig. 15b), while E_3 may fail if the seeds are provided closely inside and outside the gaps (Fig. 15c).

On the other hand, Theorem 4 indicates that, in the absence of tie zones, the tricks used to improve the results of the min-cut/max-flow algorithm by employing transformations which emphasize the weight differences [29], lead

in fact to approximations of the IFT-SC segmentation. The minimum cut by the max-flow algorithm has also its own implementation drawbacks. First, efficient implementations usually consider discrete arc weights and, therefore, the method may suffer from integer overflow as we increase the value of power n. For example, note that the best result of max-flow given in Fig. 14d required $n = 7$. A second issue is related to its efficiency, while IFT-based methods have linear-time implementations [21], the max-flow algorithm is still polynomial [9] and is not extensive to simultaneous segmentation of multiple objects [1]. Therefore, from the opposite point of view, the max-flow segmentation under the equivalence conditions (Theorem 4) is, in fact, a limited implementation of IFT-SC, which does not output neither the forest P nor the map $V(t)$; it only gives the label image $\hat{L}$.

Although the absence of tie zones (Definition 7) seems to be a very strong assumption, we have noted for several real images that the tie zones either do not appear or are represented by a few pixels, when we use a suitable arc-weight estimation [33]. In fact, an upper bound for the error of assuming Theorem 4 always true for a sufficiently high value of n is given by the tie zones, which can be computed as well. Note that a local optimization of $\tilde{E}_1$ restricted to these tie zones can further solve the problem.

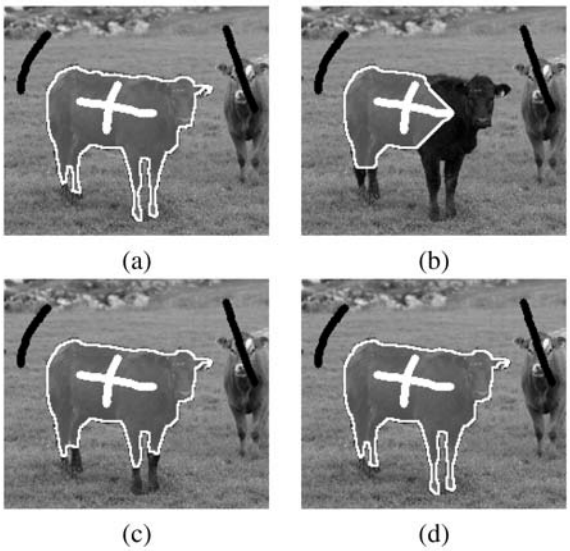

Fig. 14 (**a**) The segmentation by IFT-SC using (2). (**b**–**d**) Results of the min-cut/max-flow algorithm with increasing power values, 1, 3 and 7 respectively

10 Conclusions

We presented a self-contained paper with theoretical proofs of important existing connections between relevant segmentation methods of the literature. It was proven that some existing IFT-based methods indeed minimize the graph-cut measures E_3 and E_5 (Theorems 1, 2 and 3). It was also shown that the absence of tie zones is the necessary condition to the converse of the theorem stated in [1] (Theorem 4). However, it is important to note that the relations presented here only apply to the path-value function $f_{\min}$ and that the applications of the IFT go much beyond this particular class of operator [21].

Theorem 1 shows that IFT-SC provides optimum segmentation results from two points of view: as an optimum-path forest and as a minimum cut in the graph according to measure E_3, which can take into account both image and object properties by using the arc weights as described in [33]. Theorem 2 shows that this optimality is even stronger, given that it is also piecewise optimum. These results give the theoretical foundations to explain the great success and popularity of IFT-SC [16, 30] and its related methods (e.g., WT [7], RFC [39, 45], IRFC [13]). The connectivity between pixels is a key concept, which is naturally exploited by IFT-based methods. More recently, it has also been used to overcome graph-cut shortcomings in a combined approach, called DijkstraGC [46]. Apart from such hybrid approaches, it was shown that, to fix the undesirable bias of the min-cut algorithm toward small boundaries, we must change the graph topology or change the arc weights by employing some increasing transformation that penalizes arcs with high weights. The first solution comes with its own drawbacks, as discussed in Sects. 5 and 9, while the second solves the bias problem nicely and can be used even when it is hard to obtain good probability maps for object and background [29]. On the other hand, Theorem 4 indicates that these transformations are in fact making the min-cut segmentation to behave like IFT-SC, with more or less intensity, depending on which increasing function is used. This indicates that segmentation methods based on the min-cut

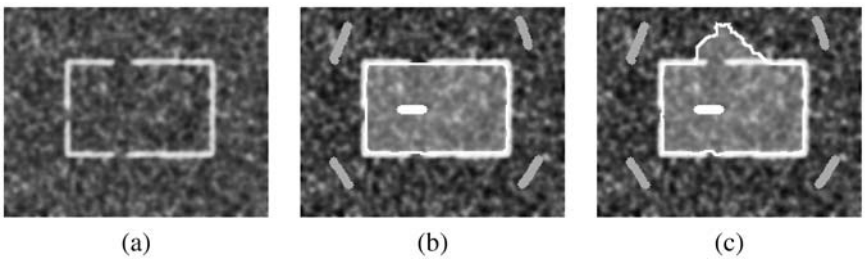

Fig. 15 (**a**) A synthetic gradient image with gaps. (**b**) The result of min-cut/max-flow algorithm. (**c**) The segmentation by IFT-SC using (2)

algorithm should compare their approaches with IFT-SC in order to justify the use of a more computationally expensive algorithm, especially for 3D applications, or at least mention IFT-SC as a related method. Our paper clarifies many of these aspects, presenting the definitions and theorems in a natural flow of evolution.

As future work, we intend to investigate ways to minimize tie zones by suitable arc-weight assignment and to incorporate shape and appearance model for automatic image segmentation.

Acknowledgements The authors thank FAPESP (Procs. 05/59808-0, 07/52015-0), CNPq (Procs. 302617/2007-8 and 472402/2007-2), and CAPES for the financial support.

References

1. Allène, C., Audibert, J.Y., Couprie, M., Cousty, J., Keriven, R.: Some links between min-cuts, optimal spanning forests and watersheds. In: Mathematical Morphology and its Applications to Signal and Image Processing (ISMM), pp. 253–264. MCT/INPE (2007)
2. Audigier, R., Lotufo, R.A.: The tie-zone watershed: Definition, algorithm and applications. In: Proceedings of IEEE International Conference on Image Processing (ICIP 05), pp. 654–657 (2005)
3. Audigier, R., Lotufo, R.A.: Watershed by image foresting transform, tie-zone, and theoretical relationships with other watershed definitions. In: Proc. of the 8th Intl. Symposium on Mathematical Morphology (ISMMâ07), pp. 277–288 (2007)
4. Audigier, R., Lotufo, R.A.: Seed-relative segmentation robustness of watershed and fuzzy connectedness approaches. In: XX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI), pp. 61–68. Belo Horizonte, MG Oct 2007. IEEE CPS
5. Bai, X., Sapiro, G.: Distance cut: interactive segmentation and matting of images and videos. In: IEEE Intl. Conf. on Image Processing (ICIP), vol. 2, pp. 249–252. San Antonio, Texas (2007)
6. Bergo, F.P.G., Falcão, A.X., Miranda, P.A.V., Rocha, L.M.: Automatic image segmentation by tree pruning. J. Math. Imaging Vis. **29**(2–3), 141–162 (2007)
7. Beucher, S., Meyer, F.: The morphological approach to segmentation: The watershed transformation. In: Mathematical Morphology in Image Processing, pp. 433–481. Dekker, New York (1993). Chap. 12
8. Boykov, Y., Funka-Lea, G.: Graph cuts and efficient N-D image segmentation. Int. J. Comput. Vis. **70**(2), 109–131 (2006)
9. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans. Pattern Anal. Mach. Intell. **26**(9), 1124–1137 (2004)
10. Boykov, Y.Y., Jolly, M.-P.: Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. In: International Conference on Computer Vision (ICCV), vol. 1, pp. 105–112 (2001)
11. Carballido-Gamio, J., Belongie, S.J., Majumdar, S.: Normalized cuts in 3D for spinal MRI segmentation. IEEE Trans. Med. Imag. **23**(1), 36–44 (2004)
12. Ciesielski, K.C., Udupa, J.K.: Affinity functions: recognizing essential parameters in fuzzy connectedness based image segmentation. In: Proc. of SPIE on Medical Imaging: Image Processing, vol. 7259 (2009)
13. Ciesielski, K.C., Udupa, J.K., Saha, P.K., Zhuge, Y.: Iterative relative fuzzy connectedness for multiple objects with multiple seeds. Comput. Vis. Image Underst. **107**(3), 160–182 (2007)
14. Cousty, J., Bertrand, G., Najman, L., Couprie, M.: Watersheds, minimum spanning forests, and the drop of water principle. Technical Report IGM2007-01, Université de Marne-la-Vallée (2007)
15. Cox, I.J., Rao, S.B., Zhong, Y.: Ratio regions: a technique for image segmentation. In: Intl. Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 557–564 (1996)
16. Falcão, A.X., Bergo, F.P.G.: Interactive volume segmentation with differential image foresting transforms. IEEE Trans. Med. Imag. **23**(9), 1100–1108 (2004)
17. Falcão, A.X., Udupa, J.K., Miyazawa, F.K.: An ultra-fast user-steered image segmentation paradigm: Live-wire-on-the-fly. IEEE Trans. Med. Imag. **19**(1), 55–62 (2000)
18. Falcão, A.X., da Cunha, B.S., Lotufo, R.A.: Design of connected operators using the image foresting transform. In: Proc. of SPIE on Medical Imaging, vol. 4322, pp. 468–479 (Feb 2001)
19. Falcão, A.X., Costa, L.F., da Cunha, B.S.: Multiscale skeletons by image foresting transform and its applications to neuromorphometry. Pattern Recogn. **35**(7), 1571–1582 (2002)
20. Falcão, A.X., Bergo, F.P.G., Miranda, P.A.V.: Image segmentation by tree pruning. In: XVII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI), pp. 65–71. IEEE Press, New York (2004)
21. Falcão, A.X., Stolfi, J., Lotufo, R.A.: The image foresting transform: Theory, algorithms, and applications. IEEE Trans. Pattern Anal. Mach. Intell. **26**(1), 19–29 (2004)
22. Falcão, A.X., Miranda, P.A.V., Rocha, A.: A linear-time approach for image segmentation using graph-cut measures. In: 8th Intl. Conf. on Advanced Concepts for Intelligent Vision Systems (ACIVS), LNCS, vol. 4179, pp. 138–149. Antwerp, Belgium, Springer, Berlin (2006)
23. Ford, L., Fulkerson, D.: Flows in Networks. Princeton University Press, Princeton (1962)
24. Fowlkes, C., Belongie, S., Malik, J.: Efficient spatiotemporal grouping using the Nyström method. In: Intl. Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 231–238 (2001)
25. Grau, V., Mewes, A.U.J., Alcaniz, M., Kikinis, R., Warfield, S.K.: Improved watershed transform for medical image segmentation using prior information. IEEE Trans. Med. Imag. **23**(4), 447–458 (2004)
26. Herman, G.T., Carvalho, B.M.: Multiseeded segmentation using fuzzy connectedness. IEEE Trans. Pattern Anal. Mach. Intell. **23**(5), 460–474 (2001)
27. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **26**(2), 147–159 (2004)
28. Lei, T., Udupa, J.K., Saha, P.K., Odhner, D.: Artery-vein separation via MRA—An image processing approach. IEEE Trans. Med. Imag. **20**(8) (2001)
29. Liang, L., Rehm, K., Woods, R.P., Rottenberg, D.A.: Automatic segmentation of left and right cerebral hemispheres from MRI brain volumes using the graph cuts algorithm. NeuroImage **34**(3), 1160–1170 (2007)
30. Lotufo, R.A., Falcão, A.X.: The ordered queue and the optimality of the watershed approaches. In: Mathematical Morphology and its Applications to Image and Signal Processing. vol. 18, pp. 341–350. Kluwer Academic, Dordrecht (2000)
31. Miranda, P.A.V., Falcão, A.X., Rocha, A., Bergo, F.P.G.: Object delineation by κ-connected components. EURASIP J. Adv. Signal Process. **2008**, 467928 (2008). doi:10.1155/2008/467928
32. Miranda, P.A.V., Falcão, A.X., Udupa, J.K.: *CLOUDS*: A model for synergistic image segmentation. In: Proc. of the 5th IEEE Intl. Symp. on Biomedical Imaging: From Nano to Macro (ISBI), pp. 209–212. Paris, France, May 14–17th, 2008

33. Miranda, P.A.V., Falcão, A.X., Udupa, J.K.: Synergistic arc-weight estimation for interactive image segmentation using graphs. Technical Report IC-08-21, Institute of Computing, University of Campinas (Sep 2008)
34. Moonis, G., Liu, J., Udupa, J.K., Hackney, D.B.: Estimation of tumor volume with fuzzy-connectedness segmentation of MR images. Am. J. Neuroradiol. **23**, 356–363 (2002)
35. Nguyen, H.T., Worring, M., van den Boomgaard, R.: Watersnakes: energy-driven watershed segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **25**(3), 330–342 (2003)
36. Protiere, A., Sapiro, G.: Interactive image segmentation via adaptive weighted distances. IEEE Trans. Image Process. **16**(4), 1046–1057 (2007)
37. Roerdink, J.B.T.M., Meijster, A.: The watershed transform: Definitions, algorithms and parallelization strategies. Fundam. Inform. **41**, 187–228 (2000)
38. Sá, A.M., Vieira, M.B., Montenegro, A.A., Carvalho, P.C.P., Velho, L.: Actively illuminated objects using graph-cuts. In: XIX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI), pp. 45–52. IEEE Press, New York (2006)
39. Saha, P.K., Udupa, J.K.: Relative fuzzy connectedness among multiple objects: theory, algorithms, and applications in image segmentation. Comput. Vis. Image Underst. **82**, 42–56 (2001)
40. Sarkar, S., Boyer, K.L.: Quantitative measures of change based on feature organization: eigenvalues and eigenvectors. In: Intl. Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 478–483 (1996)
41. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 888–905 (2000)
42. Tsai, Y.P., Lai, C.C., Hung, Y.P., Shih, Z.C.: A Bayesian approach to video object segmentation via merging 3-D watershed volumes. IEEE Trans. Circuits Syst. Video Technol. **15**(1), 175–180 (2005)
43. Udupa, J.K., Samarasekera, S.: Fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. Graph. Models Image Process. **58**, 246–261 (1996)
44. Udupa, J.K., Saha, P.K.: Fuzzy connectedness and image segmentation. Proc. IEEE **91**(10), 1649–1669 (2003)
45. Udupa, J.K., Saha, P.K., Lotufo, R.A.: Relative fuzzy connectedness and object definition: Theory, algorithms, and applications in image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **24**, 1485–1500 (2002)
46. Vicente, S., Kolmogorov, V., Rother, C.: Graph cut based image segmentation with connectivity priors. In: Computer Vision and Pattern Recognition, pp. 1–8, Jun 2008
47. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: International Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 511–518 (2001)
48. Wang, S., Siskind, J.M.: Image segmentation with minimum mean cut. In: International Conference on Computer Vision (ICCV), vol. 1, pp. 517–525, Jul 2001
49. Wang, S., Sinkind, J.M.: Image segmentation with ratio cut. IEEE Trans. Pattern Anal. Mach. Intell. **25**(6), 675–690 (2003)
50. Wu, Z., Leahy, R.: An optimal graph theoretic approach to data clustering: theory and its applications to image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **15**(11), 1101–1113 (1993)
51. Zheng, D., Zhao, Y., Wang, J.: An efficient method of license plate location. Pattern Recogn. Lett. **26**(15), 2431–2438 (2005)

Paulo A.V. Miranda received the Computer Engineering degree (2003) and the M.Sc. degree in Computer Science (2006) from the University of Campinas (UNICAMP), SP, Brazil. Since 2006, he is pursuing a Ph.D. in Computer Science at the University of Campinas (UNICAMP), SP, Brazil. During 2008–2009, he worked at the University of Pennsylvania, PA, USA, on image segmentation for his doctorate. His research involves image segmentation and analysis, medical imaging applications, pattern recognition and content-based image retrieval.

Alexandre X. Falcão received a B.Sc. in Electrical Engineering (1988) from the Federal University of Pernambuco (UFPE), PE, Brazil. He has worked in image processing and analysis since 1991. In 1993, he received a M.Sc. in Electrical Engineering from the University of Campinas (UNICAMP), SP, Brazil. During 1994–1996, he worked at the University of Pennsylvania, PA, USA, on interactive image segmentation for his doctorate. He got his doctorate in Electrical Engineering from the University of Campinas (UNICAMP) in 1996. In 1997, he developed video quality evaluation methods for TV Globo, RJ, Brazil. He has been Professor at the Institute of Computing, University of Campinas, since 1998, and his research interests include image/video segmentation and analysis, volume visualization, content-based image retrieval, mathematical morphology, medical imaging applications and pattern recognition.