Contents

Fuzzy Connectedness image segmentation in Graph Cut formulation: A linear-time algorithm and a comparative analysis

Krzysztof Chris Ciesielski,[a,b,*] Jayaram K. Udupa,[b] A.X. Falcão,[c] and P.A.V. Miranda[d]

[a]Department of Mathematics, West Virginia University, Morgantown, WV 26506-6310

[b]Department of Radiology, MIPG, University of Pennsylvania, Blockley Hall – 4th Floor, 423 Guardian Drive, Philadelphia, PA 19104-6021

[c]Institute of Computing, University of Campinas, Campinas, SP, Brazil

[d]Department of Computer Science, IME, University of São Paulo (USP), São Paulo, SP, Brazil

February 10, 2012

Abstract

A deep theoretical analysis of the graph cut image segmentation framework presented in this paper simultaneously translates into important contributions in several directions.

The most important practical contribution of this work is a full theoretical description, and implementation, of a novel powerful segmentation algorithm, $\mathrm{GC}_{\max}$. The output of $\mathrm{GC}_{\max}$ coincides with a version of a segmentation algorithm known as Iterative Relative Fuzzy Connectedness, IRFC. However, $\mathrm{GC}_{\max}$ is considerably faster than the classic IRFC algorithm, which we prove theoretically and show experimentally. Specifically, we prove that, in the worst case scenario, the $\mathrm{GC}_{\max}$ algorithm runs in linear time with respect to the variable

*Corresponding author;
E-mail: KCies@math.wvu.edu; web page: `http://www.math.wvu.edu/~kcies`;

$M = |C| + |Z|$, where $|C|$ is the image scene size and $|Z|$ is the size of the allowable range, Z, of the associated weight/affinity function. For most implementations, Z is identical to the set of allowable image intensity values, and its size can be treated as small with respect to $|C|$, meaning that $O(M) = O(|C|)$. In such a situation, $\mathrm{GC}_{\max}$ runs in linear time with respect to the image size $|C|$.

We show that the output of $\mathrm{GC}_{\max}$ constitutes a solution of a graph cut energy minimization problem, in which the energy is defined as the ℓ_∞ norm $||F_P||_\infty$ of the map F_P that associates, with every element e from the boundary of an object P, its weight $w(e)$. This formulation brings IRFC algorithms to the realm of the graph cut energy minimizers, with energy functions $||F_P||_q$ for $q \in [1, \infty]$. Of these, the best known minimization problem is for the energy $||F_P||_1$, which is solved by the classic min-cut/max-flow algorithm, referred to often as the Graph Cut algorithm.

We notice that a minimization problem for $||F_P||_q$, $q \in [1, \infty)$, is identical to that for $||F_P||_1$, when the original weight function w is replaced by w^q. Thus, any algorithm $\mathrm{GC}_{\mathrm{sum}}$ solving the $||F_P||_1$ minimization problem, solves also one for $||F_P||_q$ with $q \in [1, \infty)$, so just two algorithms, $\mathrm{GC}_{\mathrm{sum}}$ and $\mathrm{GC}_{\max}$, are enough to solve all $||F_P||_q$-minimization problems. We also show that, for any fixed weight assignment, the solutions of the $||F_P||_q$-minimization problems converge to a solution of the $||F_P||_\infty$-minimization problem ($||F_P||_\infty = \lim_{q\to\infty} ||F_P||_q$ is not enough to deduce that).

An experimental comparison of the performance of $\mathrm{GC}_{\max}$ and $\mathrm{GC}_{\mathrm{sum}}$ algorithms is included. This concentrates on comparing the actual (as opposed to provable worst scenario) algorithms' running time, as well as the influence of the choice of the seeds on the output.

1 Introduction

The image segmentation field has a rich literature dating back to the 60's. For the consideration of this paper, it is useful to categorize the segmentation algorithms into three groups: purely image-based (pI), appearance model-based (AM), and hybrid. pI methods focus on delineating objects based entirely on the information about the object that can be harnessed from the given image. AM approaches bring in information about the object family in terms of its appearance variation in the form of statistical/fuzzy texture and/or shape models to bear on the segmentation problem. Hy-

brid approaches are recent; they combine synergistically the pI and AM approaches in an attempt to overcome the weaknesses of the individual approaches. The major frameworks existing under the pI approaches include level sets (LS), active boundaries, fuzzy connectedness (FC), graph cut (GC), watershed (WS), clustering, and Markov Random Field. Since pI approaches rely mostly on information present in the given image, they offer the opportunity of studying them under a unified mathematical theory, irrespective of the different mathematical frameworks utilized in expressing the individual methods, such as variational calculus for LS, graph theory for GC, etc. In this paper, we will continue such a study previously dedicated to a comparison of FC and LS [18], but now focusing on GC and its variations and its comparison to FC.

We will put a special emphasis on the delineation algorithms, that is, the segmentation procedures returning only one object region of interest at a time rather than multiple objects simultaneously. This makes the presentation clearer, even when a method can be easily extended to a multi-object version. In addition, the comparisons of different segmentation methods, both theoretical and experimental, is easier in this single-object setting. The general, multi-object segmentation algorithms will be also discussed here, but in a format of generalizations of the appropriate delineation methods and only at a theoretical level.

We will focus on the *Graph Cut* image segmentation methods, in which the image information is represented in the form of a weighted graph and the delineated objects P minimize the energy functions $||F_P||_q$ for different $q \in [1, \infty]$, where F_P is a map that assigns to every element e from the boundary of object P its weight $w(e)$. In this formulation, our approach is similar to that from papers [43, 21]. We will show that all minimization problems associated with the energies $||F_P||_q$ can be solved by only two types of algorithms: $\mathrm{GC}_{\mathrm{sum}}$ and $\mathrm{GC}_{\mathrm{max}}$, that solve, respectively, the minimization problems for the energies $||F_P||_1$ and $||F_P||_\infty$.

The *graph cut* $\mathrm{GC}_{\mathrm{sum}}$ algorithms, minimizing the energy $\varepsilon^{\mathrm{sum}}(P) = ||F_P||_1$, have a rich literature [12, 7, 8, 9, 6, 10, 11]. (See also [42, 29, 27].) The energy $\varepsilon^{\mathrm{max}}(P) = ||F_P||_\infty$ used as an optimizer is relatively a new phenomenon — it seems to appear so far only in the papers [43, 21] and in a slightly different setting from the one we use in this paper. (But see also [2, 3, 23].) However, as we will show here, the energy $\varepsilon^{\mathrm{max}}$ is actually minimized by most of the algorithms from the *fuzzy connectedness, FC*, framework, which was extensively studied since 1996 [44, 37, 45, 38, 19, 46]. (See also [13, 26, 14],

where a slightly different approach to this methodology is used. For other extensions of FC, compare e.g., [33, 25], and the references listed in [15, 16].) Recall also that the *watershed, WS,* framework [5, 41, 32, 22, 23, 4, 30] can be encompassed in the FC framework [2, 23, 29], so it too minimizes the energy $\varepsilon^{\max}$.

The paper is organized as follows. In Section 2, we formally introduce the graph cut energy minimization framework for delineation of objects in digital images. All terms are precisely defined in such a way that the proofs can be presented with mathematical rigor and precision rather than as an outline at the intuitive level. In Section 3, we will describe the results from [21] and discuss their relation to the presented work. In particular, while both in [21] and in this paper, the optimizing energies, $||F_P||_q$, are identical, they are actually used to optimize different mathematical objects: we optimize the actual delineated (hard) objects; in [21], on the other hand, the optimization returns the labeling (fuzzy subset) of the scene, which only at the final step is transformed to (not necessarily $||F_P||_q$ optimal) hard object. Nevertheless, the output of the main algorithm from [21], Power Watershed, is closely related to the one we introduce here: $\mathrm{GC}_{\max}$, a fast implementation of the *Iterative Relative Fuzzy Connectedness, IRFC*, algorithm. In Section 4, we prove that the standard FC optimization algorithms, *Relative Fuzzy Connectedness, RFC* and *Iterative Relative Fuzzy Connectedness, IRFC*, minimize the energy $\varepsilon^{\max}(P) = ||F_P||_\infty$. We will also present the novel $\mathrm{GC}_{\max}$ algorithm and prove that it returns, in linear time with respect to the variable $M = |C| + |Z|$ discussed in the abstract (so, practically, with respect to the image size), a delineated IRFC object, so an $\varepsilon^{\max}$-optimizer. The $\mathrm{GC}_{\max}$ algorithm can be used to return multi-object segmentations, in both IRFC and RFC settings. The final output of the $\mathrm{GC}_{\max}$ algorithm, the object, is an $\varepsilon^{\max}$-optimizer for essentially an arbitrary choice of seeds. In Section 5, we compare optimization problems related to $\varepsilon^{\mathrm{sum}}$ and $\varepsilon^{\max}$ and the associated algorithms $\mathrm{GC}_{\mathrm{sum}}$ and $\mathrm{GC}_{\max}$. In particular, we show that, for any fixed weight assignment, the solutions of the $||F_P||_q$-minimization problems converge to a solution of the $||F_P||_\infty$-minimization problem. We point out the differences between algorithms $\mathrm{GC}_{\mathrm{sum}}$ and $\mathrm{GC}_{\max}$ that can be deduced theoretically, emphasizing computational speed and the dependence on the choice of seeds. Finally, we compare empirically different versions of algorithms $\mathrm{GC}_{\mathrm{sum}}$ and $\mathrm{GC}_{\max}$, concentrating on comparing the actual (as opposed to provable worst scenario) algorithms' running time, as well as

influence of the choice of the seeds on the algorithms' output.

2 The Graph Cut segmentation framework

2.1 Image segmentations as energy minimizers

We will identify a *digital image* $I = \langle C, f\rangle$ with its *intensity function* $f\colon C \to \mathbb{R}^\ell$, that is, a map from its *domain* — a finite set C, whose elements will be referred to as *spels*, short for space elements — into $\mathbb{R}^\ell$. The value $f(c)$ of f at c represents image intensity, an ℓ-dimensional vector (each component of which indicates a measure of some aspect of the signal, like color), at this spel.

A *segmentation* of an image $I = \langle C, f\rangle$ is any sequence $\vec{P} = \langle P_1, \ldots, P_m\rangle$ of pairwise disjoint subsets of C. We will be especially interested in the segmentations consisting of a single object (i.e., with $m = 1$), to which we refer as *delineations*. In this special case, the objects will be indicated by two disjoint sets S and T of seeds, S indicating the object, while T indicating the background. For such a pair $\langle S, T\rangle$, a "desired delineated object" will be chosen from the family $\mathcal{P}(S, T)$ of all objects $P \subset C$ containing S and which are disjoint with T.

The "desired delineated objects" are chosen with the help of an *energy* (or *cost*) *function* ε (sometimes denoted as ε_f, to stress that it is derived from the image intensity map f), which to any object P assigns a number $\varepsilon(P)$. Specifically, the "desired object" will be a $P_{\min} \in \mathcal{P}(S, T)$ which minimizes ε in $\mathcal{P}(S, T)$, that is, such that $\varepsilon(P_{\min}) \leq \varepsilon(P)$ for every $P \in \mathcal{P}(S, T)$.[1] More precisely, if, for an energy threshold θ, $\mathcal{P}_\theta(S, T)$ stands for the family of all $P \in \mathcal{P}(S, T)$ such that $\varepsilon(P) \leq \theta$, then $P_{\min}$ belongs to the family $\mathcal{P}_{\theta_{\min}}(S, T)$, where $\theta_{\min}$ is the smallest number θ for which the family $\mathcal{P}_\theta(S, T)$ is non-empty.

We will use the term *minimization problem*, $\mathrm{MP}(\varepsilon_f)$, when we refer to the process described above, that is, finding the map $\langle f, S, T\rangle \mapsto \mathcal{P}_{\theta_{\min}}(S, T)$. We will use the term *delineation (or segmentation) algorithm* (for $\mathrm{MP}(\varepsilon_f)$) when we refer to a specific numerical recipe that, given f and $\langle S, T\rangle$, returns a $P_{\min}$ from $\mathcal{P}_{\theta_{\min}}(S, T)$.

[1] A minimizing argument, in our case $P_{\min}$, for a function, in our case ε, is often denoted as $P_{\min} = \arg\min_P \varepsilon(P)$. (See e.g. [21].) However, such a notation incorrectly suggests that a minimizing object is unique.

The framework indicated above is very general and applies to a large class of image segmentation methods in the pI group. It leaves open two fundamental questions, which must be answered in any specific case of image segmentation algorithm: (Q1) How to define an energy function ε, so that the family $\mathcal{P}_{\theta_{\min}}(S,T)$ contains the "desired segmentation?" (Q2) How to find an algorithm that returns $P_{\min}$ from $\mathcal{P}_{\theta_{\min}}(S,T)$ constituting the "desired segmentation?" Below we discuss several possible answers to these questions within the graph cut segmentation framework.

2.2 Specifics of graph cut segmentation

Graph cut delineations (and segmentations) of the images are found through a two-stage process: (S1) formation of a weighted graph associated with an image, and (S2) object delineation in the graph. All results presented in this paper concern the second stage (S2). In particular, since the quality of the segmentations considered in (S2) depends solely on the structure of the graph formed in stage (S1), all theoretical results of this paper remain true, independently of how the edges of the graph and their weights are assigned.

2.2.1 Formation of a graph associated with an image

A weighted directed graph $G = \langle V, E, w\rangle$ associated with a digital image $I = \langle C, f\rangle$ has the following properties.

- V is the set of vertices of the graph and is equal to the image domain C. (In the min-cut/max-flow algorithm $\mathrm{GC}_{\mathrm{sum}}$, the set is V often expanded by two additional vertices, a source and a sink. However, in the set-up we consider, this expansion is non-essential, as explained in Section 5.)
- $E \subset V \times V$ is a binary relation representing the set of all directed edges of G, that is, $\langle c, d\rangle$ is an edge if, and only if, $\langle c, d\rangle \in E$. We will assume that E is symmetric, that is, $\langle d, c\rangle$ is an edge provided so is $\langle c, d\rangle$. In particular, all considered graphs can be treated as undirected graphs.
- $w\colon E \to [0, \infty)$ is a weight function associating with any edge $e \in E$ its weight $w(e)$. Once again, we will assume that w is symmetric, that is, that $w(c, d) = w(d, c)$ for every edge $\langle c, d\rangle$.

These properties are the only requirements we impose on G in our theoretical investigations. However, it seems appropriate to provide here the standard

interpretations of these notions used in digital imaging, which we also use in our experiments.

The scene's domain C is of the rectangular form $C_1 \times \cdots \times C_{\mathfrak{n}}$ (the symbol $\mathfrak{n} = 2, 3, 4, \ldots$ standing for the dimension of the scene, which, in our experiments is either 3 or 2), where each C_i is a set $\{1, \ldots, m_i\}$ of integers. The set E of edges is identified with the *adjacency relation* on C, often denoted as α. It is routine to declare $c \in C$ to be adjacent to $d \in C$ when the Euclidean distance $|| \cdot ||$ (or any other form of distance) does not exceed some fixed number. In most applications, we use adjacencies like 4-adjacency (for $\mathfrak{n} = 2$) or 6-adjacency (in the three-dimensional case), defined as $||c-d|| \leq 1$. Similarly, the 8-adjacency (for $\mathfrak{n} = 2$) and 26-adjacency (for $\mathfrak{n} = 3$) relations can be defined as $||c - d|| \leq \sqrt{3}$.

One of the most standard weight assignments, measuring the level of inhomogeneity between a pair of spels, is given by the following formula, where $\sigma > 0$ is a fixed constant:

$$w(c, d) = e^{-||f(c)-f(d)||^2/\sigma^2}, \quad \text{where} \quad \langle c, d\rangle \in E. \tag{1}$$

This w is related to the notion of directional derivative in direction given by $\vec{cd}$ and it has a value close to 1 (meaning that c and d are well connected) when the spels have very similar intensity values. In the fuzzy connectedness literature, w given by (1) is usually denoted as ψ and is called the *homogeneity based affinity*. (See e.g. [15, 16].) In general, a weight function is also often referred to as a *cost function* or, in FC literature, as an affinity function (and often denoted as κ).

The algorithms discussed in this paper rely on edge weights (costs) that are preferably much lower across the object's border than inside and outside it. Edge-weight assignment is an application-dependent task, which can exploit local image properties, such as pixel intensity, and object properties, such as its intensity distribution, shape, and texture, which can be obtained from user-drawn markers [48] and/or by matching with an object model [49]. The algorithms discussed here can take advantage of all these approaches for edge-weight assignment.

2.2.2 Graph cut energies and minimizing segmentations

The *boundary* $\mathrm{bd}(P)$ of an object $P \subset C$ (in a graph $G = \langle C, E, w\rangle$) is the set of all edges $\langle c, d\rangle, \langle d, c\rangle \in E$ for which $c \in P$ and d is not in P. We often refer to the set $\mathrm{bd}(P)$ as a *graph cut*, since removing these edges from G

disconnects P from its complement $C \setminus P$ in C (i.e., in this modified graph, there is no path from P to $C \setminus P$).

For $B \subset E$ let $W_B \colon E \to [0, \infty)$ be defined for every edge $e \in E$ as $W_B(e) = w(e)\chi_B(e)$, where $\chi_B \colon E \to \{0, 1\}$ is the *characteristic function* of B, that is, defined as $\chi_B(e) = 1$ for $e \in B$ and $\chi_B(e) = 0$ for $e \notin B$. For $q \in [1, \infty)$ we define the energy functions: ε^q as

$$\varepsilon^q(P) = ||W_{\mathrm{bd}(P)}||_q = \sqrt[q]{\textstyle\sum_{e \in E}[W_{\mathrm{bd}(P)}(e)]^q} = \sqrt[q]{\textstyle\sum_{\langle c,d\rangle \in \mathrm{bd}(P)}[w(c,d)]^q}$$

and ε^∞ as

$$\varepsilon^\infty(P) = \lim_{q \to \infty} \varepsilon^q(P) = \max_{\langle c,d\rangle \in \mathrm{bd}(P)} w(c,d). \tag{2}$$

(Notice that $||W_{\mathrm{bd}(P)}||_q$ equals $||F_P||_q$ for the mapping F_P defined earlier.)

In what follows, we will concentrate on the functions ε^q for $q = 1$ and $q = \infty$. We will use the notation $\varepsilon^{\mathrm{sum}}$ for ε^1 and $\varepsilon^{\max}$ for ε^∞, so that

$$\varepsilon^{\mathrm{sum}}(P) = \textstyle\sum_{\langle c,d\rangle \in \mathrm{bd}(P)} w(c,d) \quad \text{and} \quad \varepsilon^{\max}(P) = \max_{\langle c,d\rangle \in \mathrm{bd}(P)} w(c,d).$$

Of course, the minimization problem $\mathrm{MP}(\varepsilon^{\mathrm{sum}})$ is solved by the popular graph cut segmentation algorithm: the min-cut/max-flow algorithm. In what follows, we will show that $\mathrm{MP}(\varepsilon^{\max})$ is solved by the fuzzy connectedness (so, also standard watershed) segmentation algorithms. All delineation algorithms we consider in Sections 4 and 5 return, as $P_{\min}$, the smallest set[2] in $\mathcal{P}_{\theta_{\min}}(S,T)$ or in some well defined subfamily $\mathcal{P}^*(S,T)$ of $\mathcal{P}_{\theta_{\min}}(S,T)$. This additional property defines uniquely the solution of every optimization problem we consider.

3 Hard versus fuzzy graph cut minimization problems

This section can be treated as a discussion of the results presented in the papers [43] and [21]. In these papers the authors discuss image delineation algorithms that use a very similar approach to that described above: the same

[2]Smallest in the set inclusion sense, that is, such that $P_{\min} \subset P$ for all $P \in \mathcal{P}_{\theta_{\min}}(S,T)$. Notice that the existence of the smallest element of $\mathcal{P}_{\theta_{\min}}(S,T)$ is not obvious. Actually, its existence depends on the definition of the energy function ε.

weighted graphs are associated with the images and the same energy functions ε_p are used to find their minimizers which, in turn, are transformed to final image delineations. However, for most cases, the actual outputs of these algorithms need to minimize the energy functionals ε_p which they employ, see Theorem 3.2. As such, they actually do not fit the general framework described in the previous section. Nevertheless, there are interesting relationships between the two approaches, as we describe in more detail below.

Recall that a *fuzzy subset* (or, according to the terminology from [43, 21], a *labeling*) of a set C is any function $x\colon C \to [0,1]$, with the value $x(c)$ indicating a degree of membership with which c belongs to the set. Many delineation algorithms considered in the literature, as those surveyed in [21], deal with the fuzzy minimization problems, the notion obtained from that of minimization problem $\mathrm{MP}(\varepsilon_f)$ upon replacing in its definition the "hard" subsets of C by the "fuzzy" subsets of C. More precisely, for disjoint sets $S, T \subset C$, we define $\mathcal{P}^F(S,T)$ as the family of all fuzzy sets $x\colon C \to [0,1]$ with $x(c) = 1$ for all $c \in S$ and $x(c) = 0$ for all $c \in T$. For a threshold θ and an energy map $\hat{\varepsilon}$ from $\mathcal{P}^F \stackrel{\text{def}}{=} \mathcal{P}^F(\emptyset,\emptyset)$ into $[0,\infty)$, we define $\mathcal{P}^F_\theta(S,T)$ as the family of all $x \in \mathcal{P}^F(S,T)$ such that $\hat{\varepsilon}(x) \leq \theta$. Then, a *fuzzy minimization problem*, $\mathrm{MP}^F(\hat{\varepsilon}_f)$, is a map $\langle f, S, T\rangle \mapsto \mathcal{P}^F_{\hat{\theta}_{\min}}(S,T)$, where $\hat{\theta}_{\min}$ is the smallest number θ for which the family $\mathcal{P}^F_\theta(S,T)$ is non-empty. Finally, a delineation algorithm for $\mathrm{MP}^F(\hat{\varepsilon}_f)$ is any specific numerical recipe that, given f and $\langle S,T\rangle$, returns an $x_{\min}$ from $\mathcal{P}^F_{\hat{\theta}_{\min}}(S,T)$.

Any *hard set* $P \subset C$ can be treated as a fuzzy set, by identifying it with its *characteristic function* $\chi_P\colon C \to \{0,1\}$, defined as $\chi_P(c) = 1$ for $c \in P$ and $\chi_P(c) = 0$ for $c \notin P$. This allows us to identify the family $\mathcal{P}(S,T)$ with $\mathcal{P}^H(S,T) = \{\chi_P\colon P \in \mathcal{P}(S,T)\}$ and recognize $\chi_{P_{\min}}$ as a minimizer of the energy ε^H defined as $\varepsilon^H(\chi_P) = \varepsilon(P)$. In particular, if ε^H is equal to the restriction $\hat{\varepsilon} \restriction \mathcal{P}^H$ of $\hat{\varepsilon}$ to $\mathcal{P}^H \stackrel{\text{def}}{=} \mathcal{P}^H(\emptyset,\emptyset)$, then the three minimization problems $\mathrm{MP}(\varepsilon)$, $\mathrm{MP}^F(\varepsilon^H)$, and $\mathrm{MP}^F(\hat{\varepsilon} \restriction \mathcal{P}^H)$ coincide. In what follows, we will often write ε to denote ε^H.

If we are happy to accept a fuzzy minimizer that returns an x from $\mathcal{P}^F(S,T)$, not necessarily from $\mathcal{P}^H$, as a "desired object," this is a viable approach. (This can be useful, for example, in filtering [20].) However, most of the time, we are after the hard delineated objects. In particular, although the delineation algorithms presented in [43, 21] minimize the "fuzzy" energy functions $\hat{\varepsilon}\colon \mathcal{P}^F \to [0,\infty)$, they actually return a characteristic function $\bar{x}\colon C \to \{0,1\}$ of a hard object, rather than a fuzzy minimizing object

(labeling) $x_{\min} \in \mathcal{P}^F_{\hat{\theta}_{\min}}(S,T)$ indicated by the fuzzy minimization problem, where

$$\bar{x}(c) = 1 \text{ when } x_{\min}(c) \geq 0.5 \text{ and } \bar{x}(c) = 0 \text{ for } x_{\min}(c) < 0.5. \tag{3}$$

In [21], the authors consider the energy functions on $\mathcal{P}^F$ defined for every $p \in [0,\infty)$ and $q \in [1,\infty)$ via formula[3]

$$E_{p,q}(x) = \sum_{\langle c,d\rangle \in E} [w(c,d)]^p |x(c) - x(d)|^q. \tag{4}$$

Actually, for $p = 0$, the formula (4) is undefined whenever $w(c,d) = 0$. We interpret $E_{0,q}$ as $\lim_{p\to 0^+} E_{p,q}$, which leads to treating $[w(c,d)]^0$ as $\operatorname{sgn}[w(c,d)]$, where the value of the sign function $\operatorname{sgn}(a)$ is defined as 0 for $a = 0$ and 1 for $a > 0$. In [43, 21], the authors also allow $p = \infty$ and $q = \infty$, through different limiting processes, which we will discuss below.

Notice that, for every q, $E_{p,q} \restriction \mathcal{P}^H = E_{p,1} \restriction \mathcal{P}^H$ (i.e., $E_{p,q}$ agrees with $E_{p,1}$ for the hard delineations), rendering the parameter q in $E_{p,q}$ redundant for the hard optimization problem set-up:

Remark 3.1 If, for any energy functions $\bar{\varepsilon}$ and $\hat{\varepsilon}$ defined on $\mathcal{P}^F$, their restrictions $\bar{\varepsilon} \restriction \mathcal{P}^H$ and $\hat{\varepsilon} \restriction \mathcal{P}^H$ are equal, then the hard minimization problems $\mathrm{MP}(\bar{\varepsilon} \restriction \mathcal{P}^H)$ and $\mathrm{MP}(\hat{\varepsilon} \restriction \mathcal{P}^H)$ associated with them coincide.

3.1 Cases $p, q \in \mathbb{R}$ and $p = q \to \infty$

Paper [43] discusses the following variants of $E_{p,q}$:

$$\varepsilon_q(x) \stackrel{\text{def}}{=} (E_{q,q}(x))^{1/q} = \sqrt[q]{\sum_{\langle c,d\rangle \in E} \big(w(c,d)|x(c) - x(d)|\big)^q} = ||F_x||_q, \tag{5}$$

where $F_x \colon E \to \mathbb{R}$, $F_x(c,d) = w(c,d)|x(c) - x(d)|$ for $\langle c,d\rangle \in E$, and $||\cdot||_q$ is the standard ℓ_q-norm. In particular, for $q = \infty$, the formula (5) is interpreted as $\varepsilon_\infty(x) \stackrel{\text{def}}{=} \lim_{q\to\infty} (E_{q,q}(x))^{1/q} = \lim_{q\to\infty} ||F_x||_q$, leading to

$$\varepsilon_\infty(x) = ||F_x||_\infty = \max_{\langle c,d\rangle \in E} w(c,d)|x(c) - x(d)|. \tag{6}$$

[3]Actually, the most general energy formula defined in [21] is of the form $\hat{E}_{p,q}(x) = E_{p,q}(x) + \sum_{c\in V} (w_c)^p |x(c) - y(c)|^q$ for a $y \in \mathcal{P}^F$. However, in all theoretical investigations there, the unary constants w_c are taken as 0, in which case $\hat{E}_{p,q} = E_{p,q}$. Our analysis here applies only to this simplified case.

This energy function is the only form of the energy $E_{p,q}$, with $p, q \to \infty$, considered in [21] and in this paper.

The following theorem summarizes the relationships between these minimization problems.

Theorem 3.2 *Let $1 \leq q < \infty$ and $0 \leq p < \infty$.*

(a) The hard delineation optimization problem associated with ε_∞ coincides with $\mathrm{MP}(\varepsilon^{\max})$.

(b) The hard delineation optimization problems associated with $E_{p,q}$ and with $(E_{p,q})^{1/q}$ (so, also with $\varepsilon_q = (E_{q,q})^{1/q}$) coincide with $\mathrm{MP}(\varepsilon_p^{\mathrm{sum}})$, *where $\varepsilon_p^{\mathrm{sum}}$ is the energy $\varepsilon^{\mathrm{sum}}$ associated with the graph $G = \langle C, E, w^p\rangle$.*

(c) Moreover, if $q \neq 1$, then the hard object $\bar{x}$ associated, as in (3), with a fuzzy minimizer $x_{\min}$ for the fuzzy energy function $E_{p,q}$ need not minimize the associated hard delineation energy function; that is, $\bar{x}$ need not belong to the appropriate family $\mathcal{P}^H_{\theta_{\min}}(S,T)$.

Proof. (a) Clearly $\varepsilon_\infty \restriction \mathcal{P}^H = \varepsilon^{\max}$, so also $\mathrm{MP}(\varepsilon_\infty \restriction \mathcal{P}^H) = \mathrm{MP}(\varepsilon^{\max})$.

(b) The map $y \mapsto y^{1/q}$ is strictly increasing for every $q \in [1, \infty)$, so the optimization problem (fuzzy or hard) associated with $E_{p,q}$ is clearly equivalent to that for $(E_{p,q})^{1/q}$ (since the associated families $\mathcal{P}_{\theta_{\min}}(S,T)$ are identical). Since $E_{p,q} \restriction \mathcal{P}^H = \varepsilon_p^{\mathrm{sum}}$, (b) follows.

(c) This part is justified by Example A.1 in the Appendix. ∎

It was noticed in [43] that, for the energy $\varepsilon_1 = E_{1,1}$ (i.e., for $q = 1$), we have $\mathcal{P}^F_{\theta_{\min}}(S,T) = \mathcal{P}^H_{\theta_{\min}}(S,T)$, so, in this case, the fuzzy $\mathrm{MP}(E_{1,1})$ and the hard $\mathrm{MP}(E_{1,1} \restriction \mathcal{P}^H)$ minimization problems coincide with the "classic" min-cut/max-flow problem $\mathrm{MP}(\varepsilon_p^{\mathrm{sum}})$. For all other energy functions considered in Theorem 3.2 (including the cases of random walk $\varepsilon_2 = (E_{2,2})^{1/2}$ and of the ℓ_∞ energy ε_∞ studied in [43, 21]), the algorithmic output $\bar{x}$ (derived from $x_{\min}$) does not constitute (an exact) solution to the related hard optimization problem. This, for example, explains why the experimental results from [43] for the ℓ_∞ algorithm are not robust (i.e., the delineations lack stability with changing seeds), in spite of Theorem 5.2, according to which the related hard optimization problem is provably robust. We will come back to this point in Section 5.

3.2 The case of $q \to \infty$ and $p \in \mathbb{R}$

The most natural understanding of this case would be to define the energy as $E_{p,\infty}(x) \stackrel{\text{def}}{=} \lim_{q\to\infty} E_{p,q}(x) = \sum_{\langle c,d\rangle\in E}[w(c,d)]^p \lim_{q\to\infty} |x(c)-x(d)|^q = \sum_{\langle c,d\rangle\in E}[w(c,d)]^p \lfloor x(c)-x(d)\rfloor$, where the value of the floor function $\lfloor a\rfloor$ is defined as the largest integer less than or equal to a. (This is the case since, for $a \in [0,1]$, we have $\lim_{q\to\infty} a^q = \lfloor a\rfloor$.) However, for such function, $\mathcal{P}^F_{\hat{\theta}_{\min}}(S,T)$ contains all $x \in \mathcal{P}^F(S,T)$ with $0 < x(c) < 1$ for all $c \in C$ not in $S \cup T$. In particular, any element of $\mathcal{P}^H(S,T)$ could end up as $\bar{x}$, rendering such $E_{p,\infty}$ useless. Instead, in [21, sec. 3.3] the authors use the function $\varepsilon_{p,\infty}(x) \stackrel{\text{def}}{=} \lim_{q\to\infty}(E_{p,q}(x))^{1/q} = \lim_{q\to\infty} \sqrt[q]{\sum_{\langle c,d\rangle\in E}\Big([w(c,d)]^{p/q}|x(c)-x(d)|\Big)^q}$, that is,

$$\varepsilon_{p,\infty}(x) = \max_{\langle c,d\rangle\in E} \operatorname{sgn}(w(c,d))|x(c)-x(d)|$$

and relate it to the Voronoi diagram delineation. It is clear, that $\varepsilon_{p,\infty}$ is equal to ε_∞ associated with the graph $G = \langle C, E, \operatorname{sgn}(w)\rangle$, so Theorem 3.2(a) is applicable in this case.

3.3 The case of $p \to \infty$ and $q \in \mathbb{R}$

Let $G = \langle C, E, w\rangle$ be a weighted graph associated with an image and let $S, T \subset C$ be fixed disjoint sets of seeds. Let $W = S \cup T$. A *forest* for G is any subgraph $\mathbb{F} = \langle C, E'\rangle$ free of cycles; a forest $\mathbb{F}$ is *spanning with respect to* W provided any connected component of $\mathbb{F}$ contains precisely one element of W. With any forest $\mathbb{F}$ for G we associate a set $P(S,\mathbb{F}) \in \mathcal{P}(S,T)$ of all vertices connected to S by a path in $\mathbb{F}$. A *maximal spanning forest, MSF*, for G and W is any forest $\mathbb{F} = \langle C, E'\rangle$ for G spanning with respect to W for which the number $\sum_{e\in E'} w(e)$ is maximal.

The delineation algorithm in [21] associated with $q \in [1,\infty)$ and $p \to \infty$ is referred to as *Power Watershed, PW, algorithm*. Although, as in the previous cases, its hard set output $\bar{x}$ is obtained from a fuzzy object (labeling) x, it is proved in [21, property 2] that such an $\bar{x}$ belongs to the family

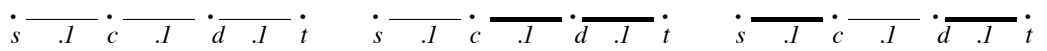
$$\mathcal{P}_M(S,W) = \{P(S,\mathbb{F})\colon \mathbb{F} \text{ is an MSF for } G \text{ and } S\cup T\},$$

that is, $\bar{x}$ is generated by a maximal spanning forest. At the same time, every object from $\mathcal{P}_M(S,W)$ maximizes the energy $\varepsilon^{\max}$ on $\mathcal{P}(S,T)$, as proved in

Theorem 4.6. (This last result is closely related to the subject of papers [3, 23].)

The above shows that PW returns an optimizer for the energy $\varepsilon^{\max}$. Since the same is true about IRFC returned objects (see Theorem 4.4), it can be argued that PW is nothing more than a version of Fuzzy Connectedness algorithm. This impression is even deepened by the fact (Theorem 4.6) that the output of the $\mathrm{GC}_{\max}$ algorithm also belongs to the family $\mathcal{P}_M(S, W)$ of objects indicated by MSF. In particular, if $\mathcal{P}_M(S, W)$ has only one element (no tie-zones), then the outputs of PW and $\mathrm{GC}_{\max}$ are identical.

Nevertheless, the algorithms PW and $\mathrm{GC}_{\max}$ use different paradigms to choose their outputs from $\mathcal{P}_M(S, W)$: $\mathrm{GC}_{\max}$ always chooses its smallest element, while, within each plateau of the graph, PW chooses the object that minimizes the energy $E_{p,q}$ for a current value of q (which, for $q > 1$, is unique). In particular, Figure 1 provides an example of a graph, in which outputs of $\mathrm{GC}_{\max}$ and PW are different.



(a) Weighted graph G with $S = \{s\}$ and $T = \{t\}$

(b) MSF $\mathbb{F}$, indicated by thicker edges, returned by $\mathrm{GC}_{\max}$; $P(s, \mathbb{F}) = \{s\}$

(c) MSF $\hat{\mathbb{F}}$, indicated by thicker edges, returned by PW; $P(s, \hat{\mathbb{F}}) = \{s, c\}$

Figure 1: Example of different outputs of $\mathrm{GC}_{\max}$ and PW used with $q > 1$; the intermediate labeling x for PW is given by $x(c) = 1/3$ and $x(d) = 2/3$ (these numbers can be verified by multivariable calculus technique)

The association of PW with the limiting process $p \to \infty$ comes from the following (fuzzy) limiting property, that holds for $q > 1$ and essentially no restriction on the seed choice (see [21, theorem 3]):

(LP^{F}) the fuzzy labeling x returned by PW is equal to a limit, as $p \to \infty$, of the fuzzy sets (labelings) x_p minimizing the energy $E_{p,q}$.

Since, in this paper, we are predominantly interested in the hard set objects, of considerably more interest to us is the following (hard) limiting property, which is analogous to the property of the $\mathrm{GC}_{\max}$ algorithm proved in Theorem 5.3:

(LP^{H}) the output $\bar{x}$ of PW is equal to a limit, as $p \to \infty$, of $\bar{x}_p$, where each is a fuzzy set (labeling) x_p minimizing the energy $E_{p,q}$.

However, unlike ($\mathrm{LP^F}$) or Theorem 5.3, the property ($\mathrm{LP^H}$) is proved in [21, theorem 1] only under an additional strong assumption[4] on seeds S and T. Moreover, as noted in [21, figure 2], ($\mathrm{LP^H}$) may be false without the assumption on seeds.

In summary,

- The algorithms $\mathrm{GC_{max}}$ and PW return outputs with very similar properties: they both minimize the same energy $\varepsilon^{\max}$ and, in both cases, can be generated by an MSF. Nevertheless, their outputs can be different (Figure 1).

- The algorithm $\mathrm{GC_{max}}$ is very fast: it provably runs in linear time with respect to the variable $M = |C| + |Z|$ discussed in the abstract (so, practically, with respect to the image size), see Theorem 4.5. There is no similar theoretical result for PW. (The experimental results presented in [21] suggest that PW runs in linear time, at least for a simple case of $q = 2$. It is also true that components of the PW algorithm, Kruskal's algorithm and plateau optimizations for $E_{p,2}$, run, provably, in a linear time with respect to the image size. However, their complicated amalgamation, formation of merged graphs, puts under question, whether a provable linear time implementation of PW can be found.)

Finally, notice that, at first glance, the most natural candidate for $E_{\infty,q}(x)$ is the limit $L(x) \stackrel{\text{def}}{=} \lim_{p\to\infty} E_{p,q}(x) = \lim_{p\to\infty} \sum_{\langle c,d\rangle\in E} [w(c,d)]^p |x(c) - x(d)|^q$, rather than $\varepsilon_{\infty,q}$. However, $L(x)$ does not exist, unless $w(c,d) \le 1$ for all $\langle c,d\rangle \in E$. Moreover, even if the limit exists (i.e., when $w(c,d) \le 1$ for all $\langle c,d\rangle \in E$), the energy $E_{\infty,q}(x) \stackrel{\text{def}}{=} L(x) = \sum_{\langle c,d\rangle\in E} \lfloor w(c,d)\rfloor |x(c) - x(d)|^q$ does not lead to a new optimization problem, as such $E_{\infty,q}$ is equal to $E_{1,q}$ for the graph $G = \langle C, E, \lfloor w\rfloor\rangle$.

4 Fuzzy connectedness as $\varepsilon^{\max}$-energy graph cut minimizer

In this section, we will prove that the standard *Fuzzy Connectedness, FC*, segmentation algorithms are graph cut minimizers for the energy function

[4]The assumption is that for every threshold t, the set $S \cup T$ intersects every connected component of the graph $\langle C, E_t\rangle$, where $E_t = \{e \in E\colon w(e) \ge t\}$.

$\varepsilon = \varepsilon^{\max}$, where a weighted graph $G = \langle C, E, w\rangle$ is associated with an image I as described in Section 2. As it is standard in the FC literature, we will denote here the graph weight function w as κ and refer to it as an *affinity function*. (The affinity functions are discussed in detail in [39, 15, 16, 48].) We will assume here that affinity is normalized, that is, that its values are in the interval $[0, 1]$. (This assumption is not essential [15, 16]. However, it is standard in the FC literature and it facilitates expression of several results in the fuzzy setting.)

We will also describe, in detail, a new version of an algorithm, $\mathrm{GC}_{\max}$ (based on the *Image Forest Transform, IFT*, [24, 29]), which returns IRFC and RFC image segmentations. Assuming that the range of the weight/affinity function $w = \kappa$ is restricted to a fixed finite set Z, we prove that $\mathrm{GC}_{\max}$ runs in linear time with respect to the variable $M = |C| + |Z|$. (For a finite set S, the symbol $|S|$ denotes the number of elements in S.) For most weight functions, $O(|Z|)$ is of the same order as the size of the set of allowable image intensity values. Moreover, $|Z|$ usually can be treated as small with respect to $|C|$, meaning that $O(M) = O(|C|)$. In such situations, $\mathrm{GC}_{\max}$ runs in linear time with respect to the image size $|C|$. (Even without a prior information on the range Z of $w = \kappa$, $\mathrm{GC}_{\max}$ runs in $O(|C| \ln |C|)$ time, the sorting time of an arbitrary set of size $|C|$.)

4.1 FC basics

A *path* p in a subset A of C is any finite sequence $\langle c_1, \ldots, c_k\rangle$ of spels in A such that any consecutive spels c_i, c_{i+1} in p are adjacent (i.e., $\langle c_i, c_{i+1}\rangle \in E$). The family of all paths in A is denoted by $\mathbb{P}^A$. Spels c and s are connected in A provided that there exists a path $p = \langle c_1, \ldots, c_k\rangle$ in A from c to s such that $c_1 = c$ and $c_k = s$. The family of all paths in A from c to d is denoted by $\mathbb{P}^A_{cd}$.

The *strength* of a path $p = \langle c_1, \ldots, c_k\rangle$, $k > 1$, is defined as $\mu(p) \stackrel{\text{def}}{=} \min\{\kappa(c_{i-1}, c_i) \colon 1 < i \leq k\}$, that is, the strength of the κ-weakest link of p. For $k = 1$ (i.e., when p has length 1) we associate with p the strongest possible value: $\mu(p) \stackrel{\text{def}}{=} 1$.[5] For $c, d \in A \subseteq C$, the (global) *κ-connectedness*

[5]For $k = 1$, the set $\{\kappa(c_{i-1}, c_i) \colon 1 < i \leq k\}$ is empty, so the first part of the definition leads to equation $\mu(\langle c_1\rangle) = \min \emptyset$. This agrees with our definition of $\mu(\langle c_1\rangle) = 1$ if we define $\min \emptyset$ as equal to 1, the highest possible value for κ. Thus, in the rest of this article we will assume that $\min \emptyset = 1$.

strength in A between c and d is defined as the strength of a strongest path in A between c and d; that is,

$$\mu^A(c,d) \stackrel{\text{def}}{=} \max\left\{\mu(p)\colon p \in \mathbb{P}^A_{cd}\right\}. \tag{7}$$

Notice that $\mu^A(c,c) = \mu(\langle c\rangle) = 1$. We will refer to the function μ^A as a *connectivity measure* (on A) induced by κ.[6] For $c \in A \subseteq C$ and a non-empty $D \subset A$, we also define $\mu^A(c,D) \stackrel{\text{def}}{=} \max_{d\in D} \mu^A(c,d)$. The basic FC object determined by S and a threshold θ is called the *absolute fuzzy connectedness, AFC, object* and can be defined as

$$P_{S\theta} \stackrel{\text{def}}{=} \left\{c \in C\colon \theta < \mu^C(c,S)\right\}.$$

Recall that $\mathcal{P}_\theta(S,T) = \{P \in \mathcal{P}(S,T)\colon \varepsilon(P) \le \theta\}$, where, in this section, $\varepsilon = \varepsilon^{\max}$. We will write also $\mathcal{P}_\theta(S)$ for $\mathcal{P}_\theta(S,\emptyset)$. The relation between AFC objects and energy $\varepsilon = \varepsilon^{\max}$ is described by the following two facts.

Lemma 4.1 *If $S \subset P \subset C$ and $P_{S\theta} \not\subset P$, then $\varepsilon(P) > \theta$.*

Proof. Fix a $c \in P_{S\theta} \setminus P$ and let $p = \langle c_1, \ldots, c_k\rangle$ be a path from c to an $s \in S$ with $\mu(p) = \mu^C(c,S) > \theta$. As $c_1 = c \notin P$ and $c_k = s \in S \subset P$, there is a $j \in \{2,\ldots,k\}$ such that $c_{j-1} \notin P$ while $c_j \in P$. This means that $\langle c_{j-1}, c_j\rangle \in \mathrm{bd}(P)$. Hence, $\varepsilon(P) = \max_{\langle c,d\rangle\in\mathrm{bd}(P)} \kappa(c,d) \ge \kappa(c_{j-1},c_j) \ge \min\{\kappa(c_{i-1},c_i)\colon 1 < i \le k\} = \mu(p) > \theta$. ∎

Theorem 4.2 *If $\emptyset \ne S \subset C$ and $\theta < 1$, then $P_{S\theta}$ is the smallest element of the family $\mathcal{P}_\theta(S)$.*

Proof. We will show first that $P_{S\theta}$ belongs to the family $\mathcal{P}_\theta(S)$ and then that it is its smallest member.

To see that $P_{S\theta} \in \mathcal{P}_\theta(S)$ we need to show that $S \subset P_{S\theta}$ and that $\varepsilon(P_{S\theta}) \le \theta$. The first condition holds, since for every $s \in S$ we have $\mu^C(s,S) \ge \mu(\langle s\rangle) = 1 > \theta$, that is, $s \in P_{S\theta}$.

To prove that $\varepsilon(P_{S\theta}) = \max_{\langle c,d\rangle\in\mathrm{bd}(P_{S\theta})} \kappa(c,d) \le \theta$, take $\langle c,d\rangle \in \mathrm{bd}(P_{S\theta})$ with $c \in P_{S\theta}$. Then $d \notin P_{S\theta}$. We need to show that $\kappa(c,d) \le \theta$. Since

[6] The min-max concept for capturing the strength of connectivity was first suggested by Rosenfeld [34, 35, 36], although by different notion of path strength without the use of affinity.

$c \in P_{S\theta}$, there exists a path $p = \langle c_1, \ldots, c_k\rangle$ in C from c to an $s \in S$ with $\mu(p) = \mu^C(c, s) > \theta$. Then, the inequality $\kappa(c, d) > \theta$ implies that $\mu^C(d, S) \geq \mu(\langle d, c_1, \ldots, c_k\rangle) = \min\{\kappa(d, c), \mu(p)\} > \theta$, implying $d \in P_{S\theta}$, which contradicts $d \notin P_{S\theta}$. Therefore, indeed $\kappa(c, d) \leq \theta$.

To see that $P_{S\theta}$ is the smallest set in $\mathcal{P}_\theta(S)$, we need to show that $P_{S\theta} \subset P$ for every $P \in \mathcal{P}_\theta(S)$. But this follows immediately from Lemma 4.1: if $P \in \mathcal{P}_\theta(S)$, then $P_{S\theta} \not\subset P$ is impossible, since otherwise, by the lemma, we would have $\varepsilon(P) > \theta$, what contradicts $P \in \mathcal{P}_\theta(S)$. ∎

If a set of seeds S contains only one seed s, then we will write $P_{s\theta}$ for the object $P_{S\theta} = P_{\{s\}\theta}$. It is easy to see that $P_{S\theta}$ is a union of all objects $P_{s\theta}$ for $s \in S$, that is, $P_{S\theta} = \bigcup_{s\in S} P_{s\theta}$.

Notice that $P_{s\theta}$ is connected, since for every $c \in P_{s\theta}$ there is a path $p = \langle c_1, \ldots, c_k\rangle$ from s to c with $\mu(p) = \mu^C(c, s) > \theta$, and such a path is contained in $P_{s\theta}$. Moreover, if $G_\theta = \langle C, E_\theta\rangle$ is a graph with E_θ consisting of the graph edges $\langle c, d\rangle$ with weight $\kappa(c, d)$ greater than θ, then $P_{s\theta}$ is a connected component of G_θ containing s, and $P_{S\theta}$ is a union of all components of G_θ intersecting S.

4.2 Simple optimization: Relative FC

An FC object that minimizes energy $\varepsilon = \varepsilon^{\max}$ is indicated by two non-empty disjoint sets $S, T \subset C$. It is referred to as the *relative fuzzy connectedness, RFC*, object and is classically defined via competition of seed sets S and T for attracting a given spel c to their realms (see [37]):

$$P_{S,T} \stackrel{\text{def}}{=} \{c \in C\colon \mu^C(c, S) > \mu^C(c, T)\}.$$

Clearly, we would like for $P_{S,T}$ to belong to $\mathcal{P}(S, T)$. It is easy to see that for this to be true, it is necessary that the number $\mu^C(S, T) \stackrel{\text{def}}{=} \max_{s\in S} \mu^C(s, T)$ is strictly less than 1. Therefore, we will always assume that the seed sets are chosen properly, that is, such that $\mu^C(S, T) < 1$.

Notice also that

$$P_{S,T} = \bigcup_{s\in S} P_{\{s\},T},$$

since $P_{S,T} = \{c \in C\colon (\exists s \in S)\mu^C(c, s) > \mu^C(c, T)\} = \bigcup_{s\in S} P_{\{s\},T}$, as $\mu^C(c, S) = \max_{s\in S} \mu^C(c, s)$.

The fact that $P_{S,T}$ minimizes the energy ε in $\mathcal{P}(S, T)$ follows, in particular, from the following theorem. Notice also that its part (iii) indicates that

$P_{S,T} = \bigcup_{s\in S} P_{\{s\},T}$ not only minimizes ε globally, but that each of its components $P_{\{s\},T}$ minimizes ε on $\mathcal{P}(\{s\},T)$ with its own version of the minimum, $\theta_s = \mu^C(s,T)$, which may be (and often is) smaller than the global minimizer $\theta_S = \mu^C(S,T)$. In other words, the object $P_{S,T}$ can be viewed as a result of minimization procedure used separately for each seed $s \in S$, which gives a sharper result than a simple minimization of global energy for the entire object $P_{S,T}$.

Theorem 4.3 *Assume that $\mu^C(S,T) < 1$. Then $P_{S,T}$ minimizes the energy $\varepsilon = \varepsilon^{\max}$ on $\mathcal{P}(S,T)$. Moreover,*

(i) *The number $\theta_S = \mu^C(S,T)$ is the minimum of ε on $\mathcal{P}(S,T)$, that is, $\theta_S = \min\{\varepsilon(P)\colon P \in \mathcal{P}(S,T)\}$.*

(ii) *If S is a singleton, then $P_{S,T}$ is the smallest set in $\mathcal{P}_{\theta_S}(S,T)$.*

(iii) *For general S, let $\mathcal{P}^*_{\theta_S}(S,T)$ be the family of all sets of the form $\bigcup_{s\in S} P^s$, where each P^s belongs to $\mathcal{P}_{\theta_{\{s\}}}(\{s\},T)$. Then $\mathcal{P}^*_{\theta_S}(S,T) \subset \mathcal{P}_{\theta_S}(S,T)$ and $P_{S,T}$ is the smallest set in $\mathcal{P}^*_{\theta_S}(S,T)$.*

Proof. First we will show that, if $\mu^C(S,T) < 1$, then

$$P_{S,T} \in \mathcal{P}(S,T). \tag{8}$$

Indeed, $S \subset P_{S,T}$, as for every $s \in S$, $\mu^C(s,S) = 1 > \mu^C(S,T) \geq \mu^C(s,T)$. Similarly, T is disjoint with $P_{S,T}$ since for every $t \in T$ we have $\mu^C(t,S) \leq \mu^C(S,T) < 1 = \mu^C(t,T)$.

Next, notice that

$$\text{if } P \in \mathcal{P}(S,T), \text{ then } \varepsilon(P) \geq \mu^C(S,T). \tag{9}$$

Indeed, choose a path $p = \langle c_1, \ldots, c_k\rangle$ from $s \in S$ to a $t \in T$ such that $\mu(p) = \mu^C(S,T)$. Since $c_1 = s \in P$ and $c_k = t \notin P$, there exists a $j \in \{2,\ldots,k\}$ with $c_{j-1} \in P$ while $c_j \notin P$. This means that $\langle c_{j-1}, c_j\rangle \in \mathrm{bd}(P)$. Hence, $\varepsilon(P) = \max_{\langle c,d\rangle\in\mathrm{bd}(P)} \kappa(c,d) \geq \kappa(c_{j-1},c_j) \geq \min\{\kappa(c_{i-1},c_i)\colon 1 < i \leq k\} = \mu(p) = \mu^C(S,T)$.

Now, to finish the proof, it is enough to prove (iii). Indeed, (iii) implies that $P_{S,T} \in \mathcal{P}_{\theta_S}(S,T)$. So, $\varepsilon(P_{S,T}) \leq \theta_S = \mu^C(S,T)$. On the other hand, (8) and (9) imply that $\mu^C(S,T) \leq \varepsilon(P_{S,T})$. Thus, $\varepsilon(P_{S,T}) = \theta_S = \mu^C(S,T)$. So, using (8) and (9) again, we conclude that $\min\{\varepsilon(P)\colon P \in \mathcal{P}(S,T)\}$ is equal

to $\varepsilon(P_{S,T}) = \theta_S = \mu^C(S,T)$. This concludes the argument for (i) and the fact that $P_{S,T}$ minimizes ε on $\mathcal{P}(S,T)$. Also, (ii) is a particular case of (iii), since for a singleton S we have $\mathcal{P}^*_{\theta_S}(S,T) = \mathcal{P}_{\theta_S}(S,T)$. Therefore, it remains to prove (iii).

For this, notice first that for every $s \in S$

$$P_{\{s\},T} = P_{\{s\}\theta_{\{s\}}} \in \mathcal{P}_{\theta_{\{s\}}}(\{s\}). \tag{10}$$

Indeed, for any $c, s, t \in C$ we have the equivalence: $\mu^C(c,s) > \mu^C(c,t)$ if and only if $\mu^C(c,s) > \mu^C(s,t)$. An easy proof of this fact can be found in [19, Proposition 2.1]. Therefore, for every $T \subset C$ we also have:

$$\mu^C(c,s) > \mu^C(c,T) \text{ if and only if } \mu^C(c,s) > \mu^C(s,T). \tag{11}$$

This implies the second equation in the following

$$\begin{aligned} P_{\{s\},T} &= \{c \in C\colon \mu^C(c,s) > \mu^C(c,T)\} \\ &= \{c \in C\colon \mu^C(c,s) > \mu^C(s,T)\} \\ &= \{c \in C\colon \mu^C(c,s) > \theta_{\{s\}}\} = P_{\{s\}\theta_{\{s\}}}. \end{aligned}$$

The relation $P_{\{s\}\theta_{\{s\}}} \in \mathcal{P}_{\theta_{\{s\}}}(\{s\})$ follows from Theorem 4.2.

Now, (8) and (10) imply $P_{S,T} = \bigcup_{s\in S} P_{\{s\},T} = \bigcup_{s\in S} P_{\{s\}\theta_{\{s\}}} \in \mathcal{P}^*_{\theta_S}(S,T)$. Next, we will show that $\varepsilon(P) \le \theta_S$ for every $P \in \mathcal{P}^*_{\theta_S}(S,T)$. This will imply that $\mathcal{P}^*_{\theta_S}(S,T) \subset \mathcal{P}_{\theta_S}(S,T)$.

To see $\varepsilon(P) \le \theta_S$, assume that $P = \bigcup_{s\in S} P^s$, where P^s belongs to $\mathcal{P}_{\theta_{\{s\}}}(\{s\},T)$ for every $s \in S$. Then, since $\mathrm{bd}(P) \subset \bigcup_{s\in S} \mathrm{bd}(P^s)$, we conclude $\varepsilon(P) = \max_{\langle c,d\rangle\in \mathrm{bd}(P)} \kappa(c,d) \le \max_{s\in S} \max_{\langle c,d\rangle\in \mathrm{bd}(P^s)} \kappa(c,d) = \max_{s\in S} \varepsilon(P^s) = \max_{s\in S} \theta_s = \max_{s\in S} \mu^C(s,T) = \theta_S$.

To finish the proof it is enough to show that $P_{S,T}$ is the smallest set in $\mathcal{P}^*_{\theta_S}(S,T)$. But, by (10), $P_{\{s\},T} = P_{\{s\}\theta_{\{s\}}} \in \mathcal{P}_{\theta_{\{s\}}}(\{s\})$, so, by Theorem 4.2, $P_{\{s\},T} \subset P^s$ for every $P^s \in \mathcal{P}_{\theta_{\{s\}}}(\{s\},T)$. Therefore, every $P = \bigcup_{s\in S} P^s$ from $\mathcal{P}^*_{\theta_S}(S,T)$ contains $P_{S,T} = \bigcup_{s\in S} P_{\{s\},T}$. ∎

The above described RFC delineation procedure easily and naturally generalizes to the segmentation algorithm of $m > 1$ separate objects. More precisely, assume that for an image $I = \langle C, f\rangle$ we have a sequence $\mathcal{S} = \langle S_1, \dots, S_m\rangle$ of pairwise disjoint non-empty sets of seeds, each S_i indicating an associated object P_i. If for each i we put $T_i = \left(\bigcup_{j=1}^m S_j\right) \setminus S_i$, then the RFC segmentation is defined as a family $\mathcal{P} = \{P_{S_i,T_i}\colon i = 1, \dots, m\}$. It is

easy to see that the different objects in $\mathcal{P}$ are disjoint. Moreover, each object P_{S_i,T_i} contains S_i provided the seeds are chosen properly, that is, when $\mu^C(S_i, S_j) < 1$ for every $j \neq i$.

It is worth to mention that while each P_{S_i,T_i} minimizes the energy $\varepsilon = \varepsilon^{\max}$ in $\mathcal{P}(S_i, T_i)$ with the energy value $\theta_i = \mu^C(S_i, T_i)$, the numbers θ_i's need not be equal when the number m of objects is greater than 2.

To find the RFC segmentation $\mathcal{P} = \{P_{S_i,T_i} \colon i = 1, \ldots, m\}$ for a given sequence $\mathcal{S} = \langle S_1, \ldots, S_m \rangle$ of seeds, it is enough to use m-times an algorithm that, for disjoint non-empty sets $S, T \subset C$, with $\mu^C(S, T) < 1$, returns the object $P_{S,T}$. In the experimental section we examine two versions of such an algorithm: RFC-standard and RFC-IFT. Each version follows the same simple procedure, as displayed. They differ only in a routine that, given a non-empty set $S \subset C$, returns $\mu^C(\cdot, S)$. So, their outputs are identical.

Algorithm RFC (-standard or -IFT)
Input: Affinity function defined on a graph $G = \langle C, E \rangle$ and non-empty disjoint sets $S, T \subset C$.
Output: The RFC object $P_{S,T} = \{c \in C \colon \mu^C(c, S) > \mu^C(c, T)\}$.
begin
1. calculate $\mu^C(\cdot, S)$ and $\mu^C(\cdot, T)$
(running appropriate subroutine twice, once for S and once for T);
2. return $P_{S,T} = \{c \in C \colon \mu^C(c, S) > \mu^C(c, T)\}$;
end

The RFC-standard algorithm calculates function $\mu^C(\cdot, S)$ using the routine $\kappa FOEMS$ [44] (not presented in this paper) that runs in time of order $O(|C|^2)$, (or, more precisely, $O(\Delta^2|C|^2)$), where $|C|$ is the size of the image domain C and Δ is the degree of the graph (i.e., the largest number of vertices that can be adjacent to a single vertex; e.g., $\Delta = 6$ for the 6-adjacency).

Thus, since line 2 of RFC runs in time $O(|C|)$, the RFC-standard algorithm stops in time of order $O(|C|^2)$.

The RFC-IFT algorithm calculates function $\mu^C(\cdot, W)$ using the $\mathrm{GC}_{\max}$ routine described in the next section, which takes as an input a non-empty set $W \subset C$ and returns, in $O(M)$ time (which is smaller than $O(|C|^2)$, see Theorem 4.6), the function $\mu^C(\cdot, W)$. Clearly, RFC-IFT runs in $O(M)$ time, since so does $\mathrm{GC}_{\max}$.

4.3 Iterative RFC and the algorithm

The RFC segmentation $\mathcal{P} = \{P_{S_i,T_i} : i = 1, \ldots, m\}$ of a graph G, associated with a sequence $\mathcal{S} = \langle S_1, \ldots, S_m \rangle$ of seeds, can still leave quite a sizable "leftover" background set $B = B_{\mathcal{P}}$ of all spels c outside any of the objects wherein the strengths of connectedness are equal with respect to the seeds. The goal of the *iterative relative fuzzy connectedness segmentation, IRFC,* is to find a way to naturally redistribute some of the spels from $B_{\mathcal{P}}$ among the object regions in a new generation (iteration) of segmentation. There are two FC delineation approaches that lead to the IRFC objects: the standard, bottom-up approach, in which the RFC object P_{S_i,T_i} is expanded to the "maximal" IRFC object $P^{\infty}_{S_i,T_i}$; and the IFT top-down approach, in which the IRFC object $P^{IFT}_{S_i,T_i}$ is chosen as the minimal among all S_i-indicated objects that result from different IFT $\mathcal{S}$-indicated segmentations. In this section we describe briefly both of these approaches and prove that indeed the objects $P^{\infty}_{S_i,T_i}$ and $P^{IFT}_{S_i,T_i}$ are identical. In addition, we show that this common object can be viewed as a result of the energy $\varepsilon = \varepsilon^{\max}$ minimization, that is, it satisfies an analog of Theorem 4.3.

Historically, the first IRFC approach was bottom-up [38, 19], so we start with it. The idea is to treat the RFC delineated objects P_{S_i,T_i} as the first iteration $P^1_{S_i,T_i}$ approximation of the final segmentation, while the next step iteration is designed to redistribute some of the background spels $c \in B_{\mathcal{P}}$, for which $\mu^C(c, S_i) = \mu^C(c, T_i)$ for some i. Such a tie can be resolved if the strongest paths justifying $\mu^C(c, S_i)$ and $\mu^C(c, T_i)$ cannot pass through the spels already assigned to another object. In other words, we like to add spels from the set $P^* = \{c \in B : \mu^{B \cup P_{S_i,T_i}}(c, S_i) > \mu^{B \cup P_{S_j,T_i}}(c, S_j) \text{ for every } j \neq i\}$, to a new generation $P^2_{S_i,T_i}$ of $P^1_{S_i,T_i}$, that is, define $P^2_{S_i,T_i}$ as $P^1_{S_i,T_i} \cup P^*$. This formula can be taken as a definition. However, from the algorithmic point of view, it is more convenient to define $P^2_{S_i,T_i}$ as

$$P^2_{S_i,T_i} = P^1_{S_i,T_i} \cup \left\{ c \in C \setminus P^1_{S_i,T_i} : \mu^C(c, S_i) > \mu^{C \setminus P^1_{S_i,T_i}}(c, T_i) \right\},$$

while the equation $P^2_{S_i,T_i} = P^1_{S_i,T_i} \cup P^*$ always holds, as proved in [19, thm. 3.7]. Thus, the IRFC object is defined as $P^{\infty}_{S_i,T_i} = \bigcup_{k=1}^{\infty} P^k_{S_i,T_i}$, where sets $P^k_{S_i,T_i}$ are defined recursively by the formulas $P^1_{S_i,T_i} = P_{S_i,T_i}$ and

$$P^{k+1}_{S_i,T_i} = P^k_{S_i,T_i} \cup \left\{ c \in C \setminus P^k_{S_i,T_i} : \mu^C(c, S_i) > \mu^{C \setminus P^k_{S_i,T_i}}(c, T_i) \right\}. \tag{12}$$

Notice that formula (12) holds also for $k = 0$, where we define $P^0_{S_i,T_i}$ as the empty set $\emptyset$.

The IRFC segmentation associated with the sequence $\mathcal{S}$ of seeds is defined as the collection $\mathcal{P}^I_{\mathcal{S}} = \{P^\infty_{S_i,T_i} : i = 1, \ldots, m\}$. Its members are still disjoint, as proved in [19]. More importantly, each IRFC object $P^\infty_{S_i,T_i}$ has the same energy value as its RFC counterpart P_{S_i,T_i}:

Theorem 4.4 *Assume that $\theta = \mu^C(S,T) < 1$. Then $P^\infty_{S,T}$ minimizes energy $\varepsilon = \varepsilon^{\max}$ on $\mathcal{P}(S,T)$, i.e., $P^\infty_{S,T} \in \mathcal{P}_\theta(S,T) \stackrel{\text{def}}{=} \{P \in \mathcal{P}(S,T) : \varepsilon(P) \leq \theta\}$.*

Proof. Clearly $P^\infty_{S,T} \in \mathcal{P}(S,T)$. In Theorem 4.3 we proved that the number $\theta = \mu^C(S,T)$ constitutes the minimal energy ε in $\mathcal{P}(S,T)$. Thus, it is enough to show that $\varepsilon(P^\infty_{S,T}) \geq \theta$. For this, take an adjacent pair $\langle c, d\rangle$ of spels such that $c \in P^\infty_{S,T}$ while $d \notin P^\infty_{S,T}$. We need to show that $\kappa(c,d) \leq \theta$.

By way of contradiction, assume that $\kappa(c,d) > \theta$ and let i be the such that $c \in P^i_{S,T} \setminus P^{i-1}_{S,T}$. (Recall that $P^0_{S,T} = \emptyset$.) We will show that this implies $d \in P^{i+1}_{S,T}$, contradicting $d \notin P^\infty_{S,T}$.

Indeed, $c \in P^i_{S,T} \setminus P^{i-1}_{S,T}$ implies that $\mu^C(c,S) > \mu^{C\setminus P^i_{S,T}}(c,T)$. We will show that $\mu^C(d,S) > \mu^{C\setminus P^{i+1}_{S,T}}(d,T)$.

First notice that $\kappa(c,d) > \mu^{C\setminus P^i_{S,T}}(c,T)$, since otherwise we would have $\mu^C(c,S) > \mu^{C\setminus P^i_{S,T}}(c,T) \geq \kappa(c,d) > \mu(S,T)$, leading to a path from S to T (via spel c) of strength $\kappa(c,d) > \mu(S,T)$, contradicting the definition of $\mu(S,T)$. Then $\mu^C(d,S) \geq \min\{\mu^C(c,S), \kappa(c,d)\} > \mu^{C\setminus P^i_{S,T}}(c,T)$. Finally, notice that $\mu^{C\setminus P^i_{S,T}}(c,T) \geq \mu^{C\setminus P^i_{S,T}}(d,T)$, since otherwise a strongest path in $C \setminus P^i_{S,T}$ from T to d extended to the spel c would have the strength $\min\{\mu^{C\setminus P^i_{S,T}}(d,T), \kappa(c,d)\} > \mu^{C\setminus P^i_{S,T}}(c,T)$, contradicting the definition of number $\mu^{C\setminus P^i_{S,T}}(d,T)$. Thus, $\mu^C(d,S) > \mu^{C\setminus P^i_{S,T}}(c,T) \geq \mu^{C\setminus P^i_{S,T}}(d,T)$, leading to the promised contradiction $d \in P^{i+1}_{S,T}$. ∎

Next, we will describe the IFT, top-down approach, which was originally developed in [29]. In this subsection, we will use a definition of a forest for a graph $G = \langle C, E\rangle$ which is equivalent to that from Subsection 3.3, but is more convenient for constructing IFT. Thus, for a graph $G = \langle C, E\rangle$ and a non-empty set $W \subset C$, a *forest rooted at W* will be understood here as any family $\mathbb{F}$ of paths initiating from W such that: (1) for every spel $c \in C$ there is at most one path p_c in $\mathbb{F}$ which terminates at c; (2) for every path $p = \langle c_1, \ldots, c_k\rangle$ in $\mathbb{F}$, every initial segment of p (i.e., a path $\langle c_1, \ldots, c_j\rangle$ with $j = 1, \ldots, k$) also belongs to $\mathbb{F}$. A forest $\mathbb{F}$ rooted at W is *spanning* provided every $c \in C$ belongs to a path in $\mathbb{F}$. A spanning forest $\mathbb{F}$ (rooted at W) is

often (e.g. [24, 29]) identified with its *predecessor map* $Pr_{\mathbb{F}}\colon C \to C \cup \{nil\}$ defined as follows: if $p_c = \langle c_1, \ldots, c_k\rangle \in \mathbb{F}$ is the unique path with $c_k = c$, then $Pr_{\mathbb{F}}(c) = nil$ provided $k = 1$ and $Pr_{\mathbb{F}}(c) = c_{k-1}$ whenever $k > 1$. The forest $\mathbb{F}$ can be easily recovered from the predecessor function $Pr_{\mathbb{F}}$, so the objects $\mathbb{F}$ and $Pr_{\mathbb{F}}$ are often identified with each other.

Any spanning forest $\mathbb{F}$ for G rooted at W induces also its *root function* $R_{\mathbb{F}}$ from C onto $S_{\mathbb{F}} \stackrel{\text{def}}{=} \{c \in C\colon Pr_{\mathbb{F}}(c) = nil\} = W$ defined for any $c \in C$ as $R_{\mathbb{F}}(c) = c_1$, where $p_c = \langle c_1, \ldots, c_k\rangle$ is the unique path in $\mathbb{F}$ which terminates at c (i.e., with $c_k = c$). For an $S \subset C$ we also define $P(S, \mathbb{F})$ as the set of all $c \in C$ with $R_{\mathbb{F}}(c) \in S$. In particular, if $\mathcal{S} = \langle S_i\colon i = 1, \ldots, m\rangle$ is a sequence of pairwise disjoint non-empty sets of seeds in G, and $\mathbb{F}$ is a spanning forest on G for which $S_{\mathbb{F}} = \bigcup_{i=1}^{m} S_i$, then the family $\mathcal{P}_{\mathbb{F},\mathcal{S}} = \{P(S_i, \mathbb{F})\colon i = 1, \ldots, m\}$ is a partition of C to which we refer as the *segmentation indicated by* $\mathbb{F}$ *and* $\mathcal{S}$. Notice that $S_i \subset P(S_i, \mathbb{F})$ for every $P(S_i, \mathbb{F}) \in \mathcal{P}_{\mathbb{F},\mathcal{S}}$.

For a fixed non-empty $S \subset C$, we say that a *path* $p = \langle c_1, \ldots, c_k\rangle$ *is optimal* (with respect to S and a path cost function μ) provided that $c_1 \in S$ and $\mu(p) = \mu^C(c_k, S)$. A spanning forest $\mathbb{F}$ (rooted at $S_{\mathbb{F}}$) is an *optimum path forest, OPF*, (with respect to a path cost function μ) provided every path in $\mathbb{F}$ is optimal with respect to $S_{\mathbb{F}}$. Following [24] (compare also [29]), we say that any partition $\mathcal{P}_{\mathbb{F},\mathcal{S}}$ of C for which $\mathbb{F}$ is optimal is an *IFT segmentation by Seed Competition, IFT-SC*. Such partitions $\mathcal{P}_{\mathbb{F},\mathcal{S}}$ are closely related to the IRFC partition $\mathcal{P}^I_{\mathcal{S}} = \{P^{\infty}_{S_i,T_i}\colon i = 1, \ldots, m\}$, as recognized in [24]. However, the segmentations $\mathcal{P}_{\mathbb{F},\mathcal{S}}$ are, in general, not unique and, usually, not equal to $\mathcal{P}^I_{\mathcal{S}}$, since $\mathcal{P}^I_{\mathcal{S}}$ is usually not a partition of C (i.e, there may be spels in C that belong to no $P^{\infty}_{S_i,T_i}$). In addition, not all segmentations $\mathcal{P}_{\mathbb{F},\mathcal{S}}$ must minimize the energy function ε. Therefore, we need to modify slightly the IFT-SC approach to make the objects $P^{IFT}_{S_i,T_i}$ it generates equal to the IRFC objects $P^{\infty}_{S_i,T_i}$.

For a sequence $\mathcal{S} = \langle S_i\colon i = 1, \ldots, m\rangle$ of seeds in G, $i \in \{1, \ldots, m\}$, and $T_i = \bigcup_{j\neq i} S_j$, we define $P^{IFT}_{S_i,T_i}$ as the smallest set in the family

$$\mathcal{P}^{IRFC}(S_i, T_i) = \{P(S_i, \mathbb{F})\colon \mathbb{F} \text{ is an OPF with respect to } S_{\mathbb{F}} = S_i \cup T_i\}.$$

Of course, for this definition to be correct, it needs to be argued that the family $\mathcal{P}^{IRFC}(S_i, T_i)$ indeed has the smallest element. This, and the fact that $P^{IFT}_{S_i,T_i} = P^{\infty}_{S_i,T_i}$, is proved in Theorem 4.5.

The proof of Theorem 4.5 and the effective construction of objects $P^{IFT}_{S_i,T_i}$ will be based on the following $\mathrm{GC}_{\max}$ algorithm, which is a version of Dijkstra's procedure for computing minimum-cost path from a single source in

a graph. $\mathrm{GC}_{\max}$ also constitutes a modification of the algorithm from [29] to the format that best suits our goals here. Actually, $\mathrm{GC}_{\max}$ is a relatively simple modification of the Dijkstra's procedure. In the classical Dijkstra's procedure, a spel d assignments (the value of an optimal path, in our case $h(d)$, and the root and predecessor pointers, $R(d)$ and $Pr(d)$) are updated only when the new value $h(d)$ of the optimal path is strictly better than the old one. In $\mathrm{GC}_{\max}$, a spel d assignment is also updated, when the optimal path value $h(d)$ remains unchanged but the new assigned root $R(d)$ has strictly higher rank (i.e., higher $\lambda(R(d))$-value) than the old root assignment. This additional updating strategy is insured by the preorder relation $\langle h(c_1), \lambda(R(c_1))\rangle \preceq \langle h(c_2), \lambda(R(c_2))\rangle$ used in $\mathrm{GC}_{\max}$. The rank between all spels, including the set W of all seeds (which, in application, equals either $S_i \cup T_i$ or, in the RFC algorithm, $S \cup T$, where the roles of S and T can be swapped), is indicated by function λ. The smallest rank is assigned to $S = S_\lambda$, which ensures that the resulting output object $P^{IFT}_{S,T}$ is the smallest member of the family $\mathcal{P}^{IRFC}(S,T)$.

In the algorithm we will use a *dictionary linear order* relation defined on a set $\mathbb{R}^2$ as

$\langle r_1, r_2\rangle \preceq \langle s_1, s_2\rangle$ if, and only if, either $r_1 < s_1$ or both $r_1 = s_1$ and $r_2 \leq s_2$.

We will write $\langle r_1, r_2\rangle \prec \langle s_1, s_2\rangle$ when $\langle r_1, r_2\rangle \preceq \langle s_1, s_2\rangle$ but $\langle r_1, r_2\rangle \neq \langle s_1, s_2\rangle$.

In $\mathrm{GC}_{\max}$ we will use the following data structure Q, ordered by $\preceq$ as indicated in the description of $\mathrm{GC}_{\max}$. Q will hold at most $|C|$ spels at any given time and it can be defined as a simple priority queue, like binary heap, that allows insertion and deletion of any element in $O(\ln |C|)$ time [31]. However, in digital imaging practice, the set of possible values of an affinity function κ is usually restricted to a fixed set Z of a modest size, most frequently of the form $\{i/D\colon i = 0, 1, \ldots, D\}$ for D of the order $2^{12} = 4096$. With such, up-front given, information on the range Z of κ, we can use a more efficient data structure, as described below, to obtain a better estimate of the running time of $\mathrm{GC}_{\max}$.

Notice, that the restriction on the form of the range Z of κ can be insured either directly, by imposing that the values for $\kappa(c,d)$ are rounded to numbers from Z, or by assuming that the values of the image intensity function f belong to a fixed set Z_0 (different from Z) and the value of $\kappa(c,d)$ is given as $\Phi(f(c), f(d))$ for some function Φ (see e.g. (1)), that is, when $\kappa(c,d)$ depends only on the intensity values at c and d. In this second case, the values of

κ are in the set $Z = \{\Phi(z_1, z_2)\colon z_1, z_2 \in Z_0\}$, which has a fixed structure, independent of the format of the image, and contains at most $|Z_0|^2$ elements.

Algorithm $\mathrm{GC}_{\max}$

Input: Affinity function κ with values in a set $Z \subset [0,1]$ and defined on a connected graph $G = \langle C, E\rangle$. A non-empty set $W \subset C$ of seeds. A priority labeling map $\lambda\colon C \to \{0,1\}$ such that $\emptyset \neq S_\lambda \subseteq W$, where $S_\lambda \stackrel{\text{def}}{=} \{c \in C\colon \lambda(c) = 0\}$.

Output: Function $\mu^C(\cdot, W)$; an optimum path forest $\mathbb{F}$, with $S_{\mathbb{F}} = W$, indicated by its predecessor map $Pr_{\mathbb{F}}$; and the object $P^{IFT}_{S,T} = P(S, \mathbb{F})$, where $S = S_\lambda$ and $T = W \setminus S$.

Auxiliary Data Structures: Functions: $h\colon C \to \{-1\} \cup Z$ approximating $\mu^C(\cdot, W)$, $Pr\colon C \to C \cup \{nil\}$ eventually becoming the predecessor map $Pr_{\mathbb{F}}$, and $R\colon C \to C$ eventually becoming the root function $R_{\mathbb{F}}$. A priority queue Q (of size n) for spels which are ordered such that: spel c_1 can precede spel c_2 in Q (denoted $c_1 \preceq c_2$) if, and only if, $\langle h(c_1), \lambda(R(c_1))\rangle \preceq \langle h(c_2), \lambda(R(c_2))\rangle$ (see text for a description).

begin

1. initialize: $h(c) = 1$, $R(c) = c$, and $Pr(c) = nil$ for all $c \in W$, $h(c) = -1$, $R(c) = c$, and $Pr(c) = c$ for all $c \in C$ not in W;
2. insert every $c \in C$ into Q according to the priority $\preceq$;
3. *while* Q is not empty *do*
4. remove from Q a $\preceq$-maximal spel c;
5. *for* every spel d adjacent to c *do*
6. *if* $\langle h(d), \lambda(R(d))\rangle \prec \langle \min\{h(c), \kappa(c,d)\}, \lambda(R(c))\rangle$ *then*
7. set $h(d) = \min\{h(c), \kappa(c,d)\}$;
8. set $R(d) = R(c)$ and $Pr(d) = c$;
9. remove temporarily d from Q;
10. push d to Q with the current values of h and R;
11. *endif*;
12. *endfor*;
13. *endwhile*;
14. return h as $\mu^C(\cdot, W)$, forest $\mathbb{F}$ indicated by $Pr = Pr_{\mathbb{F}}$, and the object $P(S, \mathbb{F})$ which, for $\mu^C(S,T) < 1$, equals $P^{IFT}_{S,T}$;

end

In the (most important) case when the range of κ is restricted to a set

Z, we define Q as an array of buckets indexed by $Y = (\{-1\} \cup Z) \times \{0, 1\}$, which is ordered according to $\preceq$. Each bucket, say with an index $\langle z, \ell \rangle \in Y$, is represented by a pointer (possibly empty) to the first element of a doubly linked list, DLL, of all spels with the current label value of $\langle z, \ell \rangle$. Each DLL (of spels in a bucket) is represented by associating with every spel two pointers, 'prev' and 'next,' indicating, respectively, the previous and the next spel in this bucket. These pointers can have null value, if no previous (next) spel in the bucket exists. Notice that, these pointers never point to spels in different buckets. In particular, the execution of lines 9 and 10 in $\mathrm{GC}_{\max}$ can be done in $O(1)$ time. (The insertion of a spel into Q, line 10, in $O(1)$ time might be impossible, if the pointers between consecutive non-empty buckets are required as a structure of Q.) The above described Q is described in more detail in [50, Figures 4 and 5].

Theorem 4.5 *The output of $GC_{\max}$ is as indicated in the algorithm. $GC_{\max}$ runs, the worst case scenario, in time of order $O(M)$, where $M = |C| + |Z|$. Alternatively, without the explicit structure of Z, $GC_{\max}$'s running time can be estimated as $O(|C| \ln |C|)$. Moreover, if $T \neq \emptyset$ and $\mu^C(S, T) < 1$, then*

(i) $P(S, \mathbb{F}) = P^{\infty}_{S,T}$;

(ii) $P(S, \mathbb{F})$ *is the smallest element of the family* $\mathcal{P}^{IRFC}(S, T) \subset \mathcal{P}(S, T)$;

(iii) $P(S, \mathbb{F})$ *minimizes the energy* ε *on* $\mathcal{P}(S, T)$.

In particular, $P(S, \mathbb{F}) = P^{IFT}_{S,T} = P^{\infty}_{S,T}$;

Notice that, in essentially all practical applications, the set Z is known and $O(|Z|+|C|)$ coincides with $O(|C|)$, in which case, $\mathrm{GC}_{\max}$ runs in a linear time with respect to the image size.

The proof of Theorem 4.5 is presented in the next subsection.

According to Theorem 4.5(ii), the output $P^{IFT}_{S,T}$ of $\mathrm{GC}_{\max}$ is given by an OPF. The next theorem shows, in particular, that it is also given by a maximal spanning forest, MSF. It also relates the family $\mathcal{P}_\theta(S, T)$ of all $\varepsilon^{\max}$ optimizing objects with the families $\mathcal{F}_M(S, W)$ and $\mathcal{F}^{IRFC}(S, W)$ of all objects $P(S, \mathbb{F})$ associated with MSF and OPF, respectively.

Theorem 4.6 *Let $G = \langle V, E, w \rangle$ be a weighted graph, $S, T \subset V$ be non-empty disjoint sets of seeds and $W = S \cup T$. If $\mu(S, T) < 1$, then*

$$P^{IFT}_{S,T} \in \mathcal{F}_M(S, W) \subset \mathcal{F}^{IRFC}(S, W) \cap \mathcal{P}_\theta(S, T), \tag{13}$$

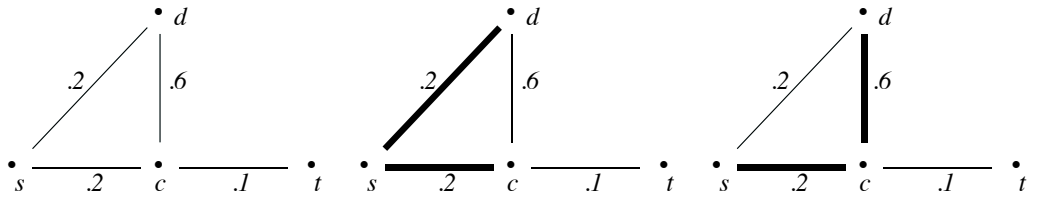
where $P^{IFT}_{S,T} = P(S, \mathbb{F})$ is the object returned by $GC_{\max}$.

In particular, the families $\mathcal{F}_M(S, W)$ and $\mathcal{F}^{IRFC}(S, W)$ share the same minimal element, $P^{IFT}_{S,T}$.

Notice, that the OPF $\mathbb{F}$ returned by $\mathrm{GC}_{\max}$ need not be MSF. (See Figure 1.) However, by Theorem 4.6, there is always an MSF $\hat{\mathbb{F}}$ for which $P^{IFT}_{S,T} = P(S, \mathbb{F}) = P(S, \hat{\mathbb{F}})$. Moreover, if one is after MSF $\hat{\mathbb{F}}$ for which $P(S, \hat{\mathbb{F}}) = P^{IFT}_{S,T}$, such an $\hat{\mathbb{F}}$ can still be found (in time $O(M)$) as follows: (1) Run $\mathrm{GC}_{\max}$ (which returns $P^{IFT}_{S,T}$ as $P(S, \mathbb{F})$ for some OPF $\mathbb{F}$, which need not be an MSF). (2) Find an MSF $\hat{\mathbb{F}}$ with $P(S, \hat{\mathbb{F}}) = P(S, \mathbb{F})$ using Kruskal's algorithm, as indicated in the proof of the theorem.

It was proved in [3, proposition 8] that every MSF is also an OPF. (The same result, proved independently, is also included in [23, theorem 21]. In both papers optimum path spanning forests are referred to as shortest path forests.) This justifies inclusion $\mathcal{F}_M(S, W) \subset \mathcal{F}^{IRFC}(S, W)$. The proof or the other parts of Theorem 4.6 is postponed to Appendix.

Next, we will provide several examples, in form of the graphs shown on figures, indicating that little can be improved in the statement of Theorem 4.6. In all figures forest edges are indicated by thicker lines. Figure 2(b) shows that the OPF $\mathbb{F}$ returned by $\mathrm{GC}_{\max}$ (i.e., with $P^{IFT}_{S,T} = P(S, \mathbb{F})$) need to be MSF. Thus, the additional work for finding MSF $\hat{\mathbb{F}}$ (indicated on Figure 2(c)) with $P(S, \hat{\mathbb{F}}) = P(S, \mathbb{F}) = P^{IFT}_{S,T}$ is essential.



(a) Weighted graph G, with s indicating the object and t the background

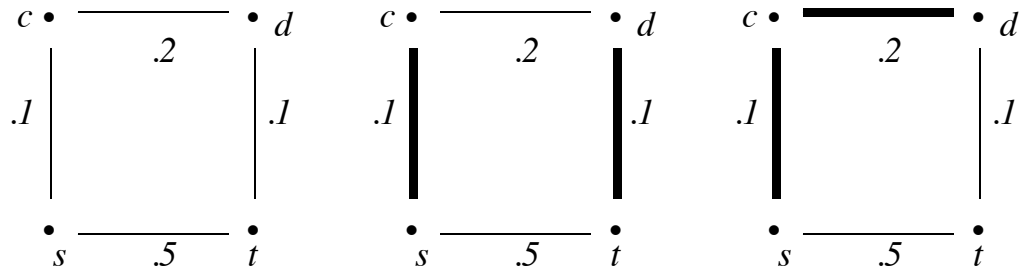
(b) OPF returned by $\mathrm{GC}_{\max}$, which is not MSF

(c) MSF for G, which indicates the same object $\{s, c, d\}$ as OPF from (b)

Figure 2: Weighted graph G, with $S = \{s, t\}$

An example of an object in $\mathcal{F}^{IRFC}(S, W) \cap \mathcal{P}_\theta(S, T)$ but not in $\mathcal{F}_M(S, W)$ is given in Figure 3. So, inclusion in Theorem 4.6 cannot be reversed.

Also, there is no inclusion between $\mathcal{F}_O(S, W)$ and $\mathcal{P}_\theta(S, T)$. An object in $\mathcal{P}_\theta(S, T) \setminus \mathcal{F}_O(S, W)$ can be chosen as $\{s, d\}$ for the graph from Figure 1.



(a) Weighted graph G, with $S = \{s\}$ and $T = \{t\}$

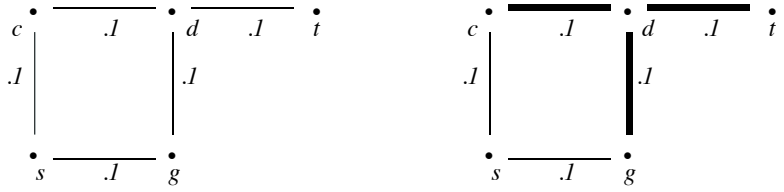
(b) OPF $\mathbb{F}$ indicating object $P(S, \mathbb{F}) = \{s, c\}$

(c) The OPF $\hat{\mathbb{F}}$ with $w(\hat{\mathbb{F}}) = .3 > .2 = w(\mathbb{F})$

Figure 3: The OPF $\mathbb{F}$ in (b) indicates object $P(S, \mathbb{F}) \in \mathcal{P}_\theta(S, T) \setminus \mathcal{F}_M(S, W)$

The object indicated in Figure 1(b) belongs to $\mathcal{F}_O(S, W) \setminus \mathcal{P}_\theta(S, T)$, if the weight of the middle edge is changed to .5.

We will finish this section, by relating the above results to the minimizers of the energy $\varepsilon^{\text{sum}}(P) = \sum_{e \in \text{bd}(P)} w_e$, which are usually calculated via graph cut algorithm GC_{sum}. It is well known that the graph cut algorithms have the so called shrinking problem: if the object is indicated only by a small set of seeds, it is likely that the object minimized by ε^{sum} will have a short boundary composed of edges with high weights, even if there is another object with a long boundary of edges with very small weight. In such a case, the families of minimizers of ε^{sum} and $\varepsilon^{\max}$ are disjoint, indicating no relation between such minimizers.



(a) Weighted graph G, with $S = \{s\}$ and $T = \{t\}$

(b) An example of OPF $\mathbb{F}$ returned by $\text{GC}_{\max}$

Figure 4: The object $P^{IFT}_{S,T} = P(S, \mathbb{F})$ has ε^{sum}-energy .2, while minimum ε^{sum}-energy on $\mathcal{P}(S, T)$ is .1, for the object $\{s, c, d, g\}$

Still an interesting question is: what happens if we know that the objects minimizing ε^{sum} also minimize $\varepsilon^{\max}$? Is it true, that an object $P^{IFT}_{S,T}$ returned

by $\mathrm{GC}_{\max}$ (so, minimizing $\varepsilon^{\max}$) minimizes also $\varepsilon^{\mathrm{sum}}$? A negative answer to this question is provided in Figure 4.

Actually, the results presented in Figure 4 remain the same, if all weights in the graph are raised to some finite power p. This shows that the limit, as $p \to \infty$, of the $\varepsilon^{\mathrm{sum}}$-minimizers need not coincide with the output of $\mathrm{GC}_{\max}$, although, as we proved in Theorem 5.3, the limit minimizes $\varepsilon^{\max}$.

4.4 Proof of correctness of $\mathrm{GC}_{\max}$ algorithm

Proof of Theorem 4.5. Since initially Q contains $n \stackrel{\mathrm{def}}{=} |C|$ spels and a spel can be removed from Q only during an execution of line 4, the loop from lines 3-13 must be executed at least n times. We will start our proof with showing, that it will be executed exactly n times.

In the proof, we will use the following notation. For every $i = 0, 1, \ldots, n$ and $c \in C$ let $h_i(c)$, $R_i(c)$, $Pr_i(c)$, and Q_i represent, respectively, the values of $h(c)$, $R(c)$, $Pr(c)$, and Q immediately after the ith execution of the loop 3-13 is completed. (For $i = 0$, we mean the moment right after the execution of line 2.) Also, to shorten the notation, we put $\gamma_i(d) \stackrel{\mathrm{def}}{=} \langle h_i(d), \lambda(R_i(d))\rangle$. In addition, for $i = 0, \ldots, n-1$, let c_i be the spel removed from $Q = Q_i$ during the $(i+1)$st execution of line 4.

Notice that, for every $i = 0, \ldots, n-1$ and $d \in C$, we have

(A) either $\langle h_{i+1}(d), R_{i+1}(d)\rangle = \langle h_i(d), R_i(d)\rangle$ or $\gamma_i(d) \prec \gamma_{i+1}(d) \preceq \gamma_i(c_i)$.

Indeed, $\langle h_{i+1}(d), R_{i+1}(d)\rangle \neq \langle h_i(d), R_i(d)\rangle$ means that we execute lines 7 and 8 for this d. So, $\gamma_i(c) = \langle h_i(d), \lambda(R_i(d))\rangle \prec \langle \min\{h_i(c_i), \kappa(c_i, d)\}, \lambda(R_i(c_i))\rangle$ holds, and, in lines 7-8, we put $\gamma_{i+1}(c) = \langle \min\{h_i(c_i), \kappa(c_i, d)\}, \lambda(R_i(c_i))\rangle$. Thus, $\gamma_i(c) \prec \gamma_{i+1}(c) = \langle \min\{h_i(c_i), \kappa(c_i, d)\}, \lambda(R_i(c_i))\rangle \preceq \gamma_i(c_i)$.

Property (A) implies, in particular, that

(A^*) $\gamma_j(d) \preceq \gamma_i(d)$ for every $d \in C$ and $0 \leq j \leq i \leq n$.

Next, we prove the following property by induction on $i = 0, \ldots, n$.

(B_i) $Q_i = C \setminus \{c_j \colon j < i\}$ and $\gamma_i(c_i) \preceq \gamma_i(c_j)$ for every $j \leq i$.

For $i = 0$, clearly (B_i) is satisfied, since $Q_0 = C$. So, assume that (B_i) is satisfied for some $0 \leq i < n$. We will show that this implies (B_{i+1}).

To prove the first part, consider the $(i+1)$st execution of the loop 3-13. Since $i < n$, $Q = Q_i$ is still not empty at line 3, and, after the execution

of line 4, we have $Q = C \setminus \{c_j \colon j < i+1\}$. To prove that Q_{i+1} is indeed equal to this Q, we need to notice only that any d for which lines 7-10 are executed belongs already to Q. But if lines 7-10 are executed for d, then $\gamma_{i+1}(d) \neq \gamma_i(d)$. Therefore, by (A), $\gamma_i(d) \prec \gamma_i(c_i)$. But, by the second part of the inductive assumption (B_i), this inequality is false for any $d \in \{c_j \colon j \leq i\}$. Thus, indeed $d \in C \setminus \{c_j \colon j < i+1\}$.

To prove the second part of (B_{i+1}), take a $j \leq i+1$. We need to show that $\gamma_{i+1}(c_{i+1}) \preceq \gamma_{i+1}(c_j)$. Clearly, this is true for $j = i+1$. So, assume that $j \leq i$. We already proved that $Q_{i+1} = C \setminus \{c_j \colon j < i+1\}$, which, by (B_i), is a subset of Q_i. So, $c_{i+1} \in Q_{i+1} \subset Q_i$. Since c_i is a $\preceq$-maximal element of Q_i, we have $\gamma_i(c_{i+1}) \preceq \gamma_i(c_i)$. So, by (A) used with $d = c_{i+1}$, we have

$$\gamma_{i+1}(c_{i+1}) \preceq \gamma_i(c_i). \tag{14}$$

In addition, by (B_i), we have $\gamma_i(c_i) \preceq \gamma_i(c_j)$. In particular, $\gamma_i(c_j) \prec \gamma_i(c_i)$ is false, and, by (A) used with $d = c_j$, we have $\gamma_i(c_j) = \gamma_{i+1}(c_j)$. Combining this with (14), we conclude $\gamma_{i+1}(c_{i+1}) \preceq \gamma_{i+1}(c_j)$, as needed.

Notice also that

(C) $\langle h_n(c_i), R_n(c_i)\rangle = \langle h_i(c_i), R_i(c_i)\rangle$ for every $i < n$.

Indeed, to prove (C) it is enough to show that, for every $i \leq j < n$, we have $\langle h_j(c_i), R_j(c_i)\rangle = \langle h_{j+1}(c_i), R_{j+1}(c_i)\rangle$. But this is true, since property (B_j) implies that $\gamma_j(c_j) \preceq \gamma_j(c_i)$, so $\gamma_j(c_i) \prec \gamma_j(c_j)$ is false. Therefore, $\langle h_{j+1}(c_i), R_{j+1}(c_i)\rangle = \langle h_j(c_i), R_j(c_i)\rangle$ follows from (A) used with j in place of i and $d = c_i$.

Now, we are in a position to prove our running time estimate for $\mathrm{GC}_{\max}$. By (B_i), for every $i < n$ we have $c_i \in Q_i = C \setminus \{c_j \colon j < i\}$. In particular, $c_i \neq c_j$ for all $j < i < n$. Therefore, Q_n is empty, that is, the loop 3-13 is executed precisely n times.

The time estimate $O(|C| + |Z|)$ is obtained by noticing that lines 1-2 are executed in this time, while the execution of lines 5-10 is done in $O(1)$ time. Thus, putting aside the execution of line 4, the loop is executed in $n \cdot O(1) = O(|C|)$ time. Therefore, the final estimate is obtained by noticing that during the entire algorithm run, the execution of line 4 requires at most $|Y| + n$ (i.e., $O(|C| + |Z|)$) operations. Indeed, although the structure of Q does not keep track of its top element (we do this only for individual buckets, but not globally), we assume that a pointer to the last non-empty bucket we

considered is maintained. If this bucket is non-empty, we simply remove its top element; this kind of operation is executed precisely n-times. If, on the other hand, the pointed bucket is empty, we consecutively move the pointer to the next bucket, until we find a non-empty one; since the pointer moves only in one direction along the list of length $|Y|$ (i.e., $O(|Z|)$), this operation can be executed at most $|Y|$ many times.

The time estimate $O(|C| \ln |C|)$ (with a different structure of Q) follows by noting that lines 1-2 are executed in $O(|C| \ln |C|)$ time and the loop is executed n-times, each of its execution requiring $O(\ln n)$ operations. (Note, that in this case, we keep a global pointer to the top element of Q, so the execution of line 4 always requires just one operation.)

To argue that $\mathrm{GC}_{\max}$ indeed returns an optimum path forest, let IRFC* represent the modification of $\mathrm{GC}_{\max}$ in which we replace an input function λ with a constant 0 mapping — this effectively reduces the dictionary order $\preceq$ on C to the standard order given by the function h, that is, $\langle h(c_1), \lambda(R(c_1))\rangle \preceq \langle h(c_2), \lambda(R(c_2))\rangle$ becomes equivalent to $h(c_1) \leq h(c_2)$. The resulting algorithm IRFC* is precisely[7] the algorithm from [24]. So, as proved in [24], IRFC* returns an optimum path forest. However, if $\mathcal{A} = \langle\langle h_i(c), R_i(c), Pr_i(c)\rangle \colon c \in C \ \&\ i \leq n\rangle$ constitutes a description of an execution of our algorithm $\mathrm{GC}_{\max}$, then it is also one possible execution of the algorithm IRFC*. It particular, $\mathbb{F}$ associated with $\mathcal{A}$ is indeed an OPF.

Next, we will prove (i)-(iii). Clearly $\mathcal{P}^{IRFC}(S,T) \subset \mathcal{P}(S,T)$, as for every spanning forest $\hat{\mathbb{F}}$ with $S_{\hat{\mathbb{F}}} = W$, the associated set $P(S,\hat{\mathbb{F}})$ contains S and is disjoint with T. Since $P^{IFT}_{S,T} \in \mathcal{P}^{IRFC}(S,T)$, to prove (i) and the rest of (ii), it is enough to show the following two facts.

(a) $P^{\infty}_{S,T} \subset P(S,\hat{\mathbb{F}})$ for every OPF $\hat{\mathbb{F}}$ with $S_{\hat{\mathbb{F}}} = W$.

(b) $P(S,\mathbb{F}) \subset P^{\infty}_{S,T}$.

To show (a), fix an OPF $\hat{\mathbb{F}}$ with $S_{\hat{\mathbb{F}}} = W$ and, by way of contradiction, assume that $P^{\infty}_{S,T} \not\subset P(S,\hat{\mathbb{F}})$.

Let $k \geq 0$ be the smallest number such that $P^{k+1}_{S,T} \not\subset P(S,\hat{\mathbb{F}})$ and let $c \in P^{k+1}_{S,T} \setminus P^{k}_{S,T}$. We will show that $c \in P(S,\hat{\mathbb{F}})$, contradicting the choice of k. Since $c \in P^{k+1}_{S,T} \setminus P^{k}_{S,T}$, we have $\mu^{C}(c,S) > \mu^{C\setminus P^{k}_{S,T}}(c,T)$. Take a

[7]Modulo changes of notation and of order of optimization.

$p_c = \langle c_1, \ldots, c_m\rangle \in \hat{\mathbb{F}}$ with $c_m = c$ and notice that $c_1 \in S$. Indeed, otherwise $c_1 \in T$ and the path p_c is disjoint with $P(S, \hat{\mathbb{F}})$. Since, by minimality of k, we have $P^k_{S,T} \subset P(S, \hat{\mathbb{F}})$, the path p_c is also disjoint with $P^k_{S,T}$. Hence, $\mu(p_c) \le \mu^{C\setminus P^k_{S,T}}(c, T) < \mu^C(c, S) \le \mu^C(c, W)$, contradicting optimality of p_c. So, $c_1 \in S$ and indeed $c \in P(S, \hat{\mathbb{F}})$, completing the proof for (a).

To prove (b), we need to show $P(S, \mathbb{F}) \subset P^\infty_{S,T}$, where $\mathbb{F}$ is an OPF returned by $\mathrm{GC}_{\max}$. Pick $i \ge 1$ with $P(S, \mathbb{F}) \cap P^i_{S,T} = P(S, \mathbb{F}) \cap P^\infty_{S,T}$ and, by way of contradiction, assume that the set $P(S, \mathbb{F}) \setminus P^\infty_{S,T} = P(S, \mathbb{F}) \setminus P^i_{S,T}$ is non-empty. Notice that

$$\mu^{C\setminus P^i_{S,T}}(d, T) \ge \mu^C(d, S) = \mu^C(d, W) \text{ for any } d \in P(S, \mathbb{F}) \setminus P^i_{S,T}. \tag{15}$$

Indeed, $\mu^C(d, S) = \mu^C(d, W) = \theta$ is implied by $d \in P(S, \mathbb{F})$. The inequality follows from $d \notin P^\infty_{S,T}$, since any d with $\mu^{C\setminus P^i_{S,T}}(d, T) < \mu^C(d, S)$ belongs to $P^{i+1}_{S,T} \subset P^\infty_{S,T}$.

Let θ be a maximal number for which there exists a $d \in P(S, \mathbb{F}) \setminus P^i_{S,T}$ with $\mu^C(d, S) = \mu^C(d, W)$ equal to θ. For this θ, there exists a k for which the following condition holds.

(P_k) There is a path $p = \langle d_1, \ldots, d_k\rangle$ in $C \setminus P^i_{S,T}$ from T to $d \in P(S, \mathbb{F}) \setminus P^i_{S,T}$ with $\mu(p) \ge \mu^{C\setminus P^i_{S,T}}(d, T) \ge \mu^C(d, S) \ge \theta$.

Let k be a minimal number for which (P_k) holds and fix $p = \langle d_1, \ldots, d_k\rangle$ and $d = d_k$ satisfying (P_k).

Notice, that $k > 1$, since otherwise $d = d_k = d_1 \in T$, contradicting $d \in P(S, \mathbb{F})$. Let $c = d_{k-1}$ and notice that

$$c \in P(T, \mathbb{F}). \tag{16}$$

To see this, by way of contradiction, assume that c belongs to $P(S, \mathbb{F})$. Then we have $c \in P(S, \mathbb{F}) \setminus P^i_{S,T}$, as $c = d_{k-1} \notin P^i_{S,T}$. Therefore, by (15), $\mu^{C\setminus P^i_{S,T}}(c, T) \ge \mu^C(c, S) = \mu^C(c, W)$. Also, maximality of θ implies that $\mu^C(c, W) \le \theta$. So,

$$\mu^C(c, W) \ge \mu^{C\setminus P^i_{S,T}}(c, T) \ge \mu(\langle d_1, \ldots, d_{k-1}\rangle) \ge \theta \ge \mu^C(c, W) = \mu^C(c, S).$$

Thus, $\hat{p} = \langle d_1, \ldots, d_{k-1}\rangle$ is a path in $C \setminus P^i_{S,T}$ from T to $c \in P(S, \mathbb{F}) \setminus P^i_{S,T}$ with $\mu(\hat{p}) = \mu^{C\setminus P^i_{S,T}}(c, T) = \mu^C(c, S) = \theta$, contradicting minimality of k. So, (16) has been proved.

Next, notice that $\theta \leq \mu(p) \leq \mu(\langle d_1, \ldots, d_{k-1}\rangle) \leq \mu^C(c, W) = h_n(c)$ and $\theta \leq \mu(p) \leq \kappa(c_{k-1}, c_k) = \kappa(c, d)$. Since, by (16), $\lambda(R_n(d)) > 0$, this implies that $\langle \theta, 1\rangle \preceq \langle \min\{h_n(c), \kappa(c, d)\}, \lambda(R_n(c))\rangle$.

Let $c = c_i$. Then, by (C), $\langle h_n(c), R_n(c)\rangle = \langle h_i(c), R_i(c)\rangle$, so

$$\langle \theta, 1\rangle \preceq \langle \min\{h_i(c), \kappa(c, d)\}, \lambda(R_i(c))\rangle.$$

Look at the $(i+1)$st execution of the loop 3-13. Then, $c = c_i$ and d will satisfy the condition from line 5, so spel d will enter the execution of lines 6-11. This leads to

$$\langle \theta, 1\rangle \preceq \langle h_{i+1}(d), \lambda(R_{i+1}(d))\rangle. \tag{17}$$

Indeed, if $\langle \theta, 1\rangle \preceq \langle h_i(d), \lambda(R_i(d))\rangle$, then $\langle h_{i+1}(d), R_{i+1}(d)\rangle = \langle h_i(d), R_i(d)\rangle$ and (17) holds. Otherwise, lines 5-10 are executed, resulting with

$$\langle \theta, 1\rangle \preceq \langle \min\{h_i(c), \kappa(c, d)\}, \lambda(R_i(c))\rangle = \langle h_{i+1}(d), \lambda(R_{i+1}(d))\rangle,$$

finishing the argument for (17).

But (17) and (A^*) imply that

$$\langle \theta, 1\rangle \preceq \langle h_{i+1}(d), \lambda(R_{i+1}(d))\rangle \preceq \langle h_n(d), \lambda(R_n(d))\rangle = \langle \theta, \lambda(R_n(d))\rangle.$$

This is a desired contradiction, since this implies that $\lambda(R(d)) > 0$, so that $R(d) \notin S$, contradicting $d \in P(S, \mathbb{F})$. This completes the argument for (b).

The remaining condition (iii) follows from (i) and Theorem 4.4. Alternatively, it can be concluded from Theorem 4.6. ∎

5 FC versus $\mathrm{GC}_{\mathrm{sum}}$ algorithms

The term *GC algorithm* will be used here in the standard way, that is, for any $\mathrm{GC}_{\mathrm{sum}}$ algorithm minimizing energy functions $\varepsilon^{\mathrm{sum}}$. Nevertheless, perhaps in a more general sense, the term may be used to refer to either $\varepsilon^{\mathrm{sum}}$-*GC* or $\varepsilon^{\max}$-*GC* energy optimizing algorithm.

For the $\mathrm{GC}_{\mathrm{sum}}$ algorithms, a graph $G^I = \langle V, E, w\rangle$ associated with the image $I = \langle C, f\rangle$ is usually a slight modification of the graph $G = \langle C, \alpha, \kappa\rangle$ (with $V = C$, $\alpha = E$ the adjacency relation, and $\kappa = w$ the affinity/weight function) defined in Section 2. Specifically, the set of vertices V is usually defined as $C \cup \{s, t\}$; that is, the standard set C of image spels considered as

vertices is expanded by two new additional vertices s and t called *terminals*. Individually, s is referred to as *source* and t as *sink*. The set of edges is defined as

$$E = \alpha \cup \{\langle b, d\rangle\colon \text{ one of } b, d \text{ is in } C, \text{ the other in } \{s, t\}\}.$$

In other words, the edges between vertices in C remains as in G, while we connect each terminal vertex to each $c \in C$.

The simplest way to think about the terminals is that they serve as the seed indicators: s for seeds $S \subset C$ indicating the object; t for seeds $T \subset C$ indicating the background. The indication works as follows. For each edge connecting a terminal $r \in \{s, t\}$ with a $c \in C$, associate the weight: ∞ if either $r = s$ and $c \in S$, or $r = t$ and $c \in T$; and 0 otherwise. This means, that the source s has infinitely strong connection to any seed c in S, and the weakest possible to any other spel $c \in C$. Similarly, for the sink t and seeds c from T.

Now, assume that for every edge $\langle c, d\rangle \in \alpha$ we give a weight $\kappa(c, d)$ associated with the image $I = \langle C, f\rangle$. Since the algorithm for delineating the FC object uses only the information on the associated graph (which includes the weights given by the affinity κ), we can delineate RFC object $P^*_{\{s\},\{t\}} \subset V$ associated with this new graph G^I. It is easy to see that the RFC object $P_{S,T} \subset C$ associated with I is equal to $P^*_{\{s\},\{t\}} \cap C$. Similarly, for $\theta < 1$, if $P^*_{s\theta} \subset V$ is an AFC object associated with the graph G^I, then the AFC object $P_{S\theta} \subset C$ associated with I is equal to $P^*_{S\theta} \cap C$. All of this proves that, from the FC framework point of view, the matter of replacing the graph $G = \langle C, \alpha, \kappa\rangle$ with G^I is only technical in nature and results in no delineation differences.

Historically, the rationale for using in GC framework's graphs G^I, with distinctive terminals, is algorithmic in nature. More precisely, for a weighted graph $G = \langle V, E, w\rangle$ with positive weights and two distinct vertices s and t indicated in it, there is an algorithm returning the smallest set P_G in $\mathcal{P}_{\min} = \{P \in \mathcal{P}(s, t)\colon \varepsilon^{\text{sum}}(P) = \varepsilon_0\}$, where $\mathcal{P}(s, t) = \{P \subset V \setminus \{t\}\colon s \in P\}$, $\varepsilon_0 = \min\{\varepsilon^{\text{sum}}(P)\colon P \in \mathcal{P}(s, t)\}$, $\varepsilon^{\text{sum}}(P) = \sum_{e \in \text{bd}(P)} w(e)$, and $w(e)$ is the weight of the edge e in the graph.

Now, let $G^I = \langle C \cup \{s, t\}, E, w\rangle$ be the graph associated with an image I as described above; that is, the weights of edges between spels from C are obtained from the image I (in a manner similar to the affinity numbers) and weights between the other edges by seed sets S and T indicating foreground

and background. In this setting we can restate the above comments in a format similar to that of Theorems 4.3 and 4.4:

Theorem 5.1 *The GC object $P^{\Sigma}_{S,T} = C \cap P_{G^I}$ minimizes the energy ε^{sum} on $\mathcal{P}(S,T)$, and $P^{\Sigma}_{S,T}$ is the smallest set in $\mathcal{P}(S,T)$ with this energy.*

5.1 GC vs FC algorithms: theoretical comparison

In spite of similarities between the GC and RFC methodologies as indicated above, there are also considerable differences between them. One of the most important differences is sensitivity to the choice of seeds. From this point of view, the FC algorithms behave very nicely: the FC delineation results do not change if the seeds S indicating an object are replaced by another set U of seeds within the same segmentation.

Theorem 5.2 (Robustness) *Let $I = \langle C, f\rangle$ be a digital image.*

AFC: For every $s \in C$ and $\theta < 1$, if $P_{s\theta}$ is an associated AFC object, then $P_{U\theta} = P_{s\theta}$ for every $U \subset P_{s\theta}$. More generally, if $S \subset C$ and $U \subset P_{S\theta}$ is such that $U \cap P_{s\theta} \neq \emptyset$ for every $s \in S$, then $P_{U\theta} = P_{S\theta}$.

RFC: Let $\mathcal{P} = \{P_{S_i,T_i} \colon i = 1,\dots,m\}$ be an RFC segmentation associated with a sequence $\mathcal{S} = \langle S_1,\dots,S_m\rangle$ of seeds. For every i and $s \in S_i$ let $g(s)$ be in $P_{\{s\},T_i}$ and let $S'_i = \{g(s) \colon s \in S_i\}$. Then, for every i, if $T'_i = \left(\bigcup_{j=1}^{m} S'_j\right) \setminus S'_i$, then $P_{S_i,T_i} = P_{S'_i,T'_i}$.

In other words, if each seed s present in $\mathcal{S}$ is shifted to a new position $g(s) \in P_{\{s\},T_i}$, then the RFC segmentation $\{P_{S'_i,T'_i} \colon i = 1,\dots,m\}$ for the modified sequence of seeds is identical to the original one $\mathcal{P}$.

The IRFC segmentations (returned by either $\text{GC}_{\max}$ or old fashioned IRFC algorithm) are robust similarly as RFC outputs: if in the above RFC formulation, the seed choice is restricted to the first iteration approximations $P^1_{S_i,T_i}$ of $P^{\infty}_{S_i,T_i}$, then seed modification does not affect IRFC segmentation results.

The proof of Theorem 5.2 follows easily from the graph interpretation of FC objects. The proof based only on the topological description of the FC segmentations can be found in [37, 19].

Below we list several theoretical advantages of the FC framework over the GC ε^{sum}-minimization algorithms:

Speed: The FC algorithms run faster than those for GC. The theoretical estimation of the worst case run time of the two main FC algorithms, RFC-IFT and $\mathrm{GC}_{\max}$, is $O(|C|+|Z|)$ (or $O(|C|\ln|C|)$) with respect to the scene size $|C|$ (see Section 4), while the best theoretical run time for delineating $P^{\Sigma}_{S,T}$ is of order $O(|C|^3)$ (for the best known algorithms) or $O(|C|^{2.5})$ (for the fastest currently known), see [9]. The experimental comparisons of the running time also confirm that FC algorithms run faster as theoretically predicted. (See Section 5.2, Figure 5.)

Robustness: The outcome of FC algorithms is unaffected by reasonable (within the objects) changes of the position of seeds, Theorem 5.2. On the other hand, the results of GC delineation may become sensitive for even small perturbations of the seeds. See Section 5.2, Figures 5 and 6.

Multiple objects: The FC framework handles easily the segmentation of multiple objects, retaining its running time and robustness property from the single object case. (See Section 4.) In the multiple object setting, GC leads to an NP-hard problem (see [12]); so all existing algorithms for performing the required precise delineation run in exponential time, rendering them impractical. (However, there are algorithms that render approximate solutions for such GC problems in a practical time [12].)

GC shrinking problem: The GC algorithms have a tendency of choosing the objects with a very small size of the boundary, even if the weights of the boundary edges are very high. (See e.g. [6, 42]. Compare also Section 5.2, Figures 5 and 6.) This may easily lead to the segmented object being very close to either the foreground seed set S, or to the background seed set T. Therefore, unless sets S and T are already good approximations for the desired delineation, the object returned by GC may be far from desirable. This problem has been addressed by many authors, via modification of the GC method. The best known among these modifications is the method of normalized cuts (see [42]), in which the energy $\varepsilon^{\mathrm{sum}}$ is replaced by another "normalized" measure of energy cost. However, finding the resulting delineation minimizing this new energy measure is NP-hard as well (see [42]), and so only approximate solutions can be found in practical time. Notice that neither RFC nor the IRFC (so $\mathrm{GC}_{\max}$) method has any shrinking problem.

Iterative approach: The FC framework allows an iterative refinement of its connectivity measure μ^A (leading to the iterative relative FC), which in turn makes it possible to redefine ε as we go along. From the viewpoint of algorithm design, this is a powerful strategy. No such methods exist for GC at present.

All this said, it should be noted that GC also has some nice properties that FC does not possess. First notice that the shrinking problem is the result of favoring shorter boundaries over the longer; that is, GC has a smoothing effect on the boundaries. This, in several (but not all) cases of medically important image delineations, is perhaps a desirable feature. There is no boundary smoothing factor built into the FC basic framework, and, if desirable, boundary smoothing must be done at the FC post processing stage.

Another nice feature of GC graph representation G^I of an image I is that it allows effortless amalgamation of image homogeneity information (via weights of regular edges) with expected object intensity information (via weights of edges to terminal vertices). Similar amalgamation is difficult to achieve in the FC framework, since the $\varepsilon^{\max}$ energy chooses just one of these two types of information.

Next, notice that an FC delineation, output of $\mathrm{GC}_{\max}$, can be considered as a limiting case of the GC delineations, outputs of $\mathrm{GC}_{\mathrm{sum}}$. For this, first notice that the optimization problem for the energy function $\varepsilon^q(P) = \sqrt[q]{\sum_{\langle c,d\rangle\in \mathrm{bd}(P)}[w(c,d)]^q}$ coincides with the optimization problem for the energy $[\varepsilon^q(P)]^q = \sum_{\langle c,d\rangle\in \mathrm{bd}(P)}[w(c,d)]^q$, which constitutes the energy $\varepsilon^{\mathrm{sum}} = \varepsilon^1$ for the weight function w^q. In particular, the FC optimizing energy $\varepsilon^{\max}$ is a limit of the GC optimizing energies $\varepsilon^{\mathrm{sum}}$ for the modified weight function w^q. The next theorem says more: that for q large enough, the GC optimizing delineations for $\varepsilon^{\mathrm{sum}}$ used with the weigh function w^q become FC optimizing delineation for the weight function w.

In the theorem, we will use the following notation, where we assume that the graph $G = \langle C, \alpha, \kappa\rangle$ and disjoint non-empty sets $S, T \subset C$ are fixed: $\mathcal{P}_{\max}(S,T) = \{P \in \mathcal{P}(S,T)\colon \varepsilon^{\max}(P) = \theta_{\min}\}$ is the set of all delineations minimizing the energy $\varepsilon^{\max}$; $\hat{\mathcal{P}}_{\max}(S,T)$ is the set of all $\hat{P} \in \mathcal{P}(S,T)$ such that $|\hat{P}_\eta| \le |P_\eta|$ for any $P \in \mathcal{P}(S,T)$ and $\eta \in \mathbb{R}$, where $P_\eta = \{e \in \mathrm{bd}(P)\colon w(e) \ge \eta\}$;[8] for every $q \in [1,\infty)$, $\mathcal{P}^q(S,T)$ is the family of

[8] $\hat{\mathcal{P}}_{\max}(S,T)$ is the family of all $P \in \mathcal{P}_{\max}(S,T)$ minimizing the energy ε_{lex}, where $\varepsilon_{lex}(P)$ is a function from $\mathbb{R}$ to $\{0,1,\ldots\}$, with $\varepsilon_{lex}(\eta) = |P_\eta|$, and the range of ε_{lex} is

all $P \in \mathcal{P}_{\max}(S,T)$ minimizers for the energy $[\varepsilon^q]^q(P) = \sum_{\langle c,d\rangle \in \mathrm{bd}(P)} [w(c,d)]^q$; and $P^{IFT}_{S,T}(\kappa)$ and $P^{\Sigma}_{S,T}(\kappa)$ are the $\mathrm{GC}_{\max}$ and $\mathrm{GC}_{\mathrm{sum}}$ objects, respectively, determined by the set of seeds S and T, while using the same affinity/cost function $w = \kappa$.

In [29] it was proved that, under some assumptions, the objects $P^{IFT}_{S,T}(\kappa^q)$ and $P^{\Sigma}_{S,T}(\kappa^q)$ converge to the same set, as q goes to ∞. Notice also that, by [15, theorems 3 and 5], $P^{IFT}_{S,T}(\kappa^q) = P^{IFT}_{S,T}(\kappa)$ for every $q > 0$, since function x^q is increasing.

Theorem 5.3 *Let $G = \langle C, \alpha, \kappa\rangle$ and $S, T \subset C$ be such that $\theta = \mu^C(S,T) < 1$ and let*

$$q_0 = \log_\delta N, \text{ where } \delta = \min\{q > 1 \colon q = \eta/\hat{\theta} \text{ for positive } \eta, \hat{\theta} \in Z\}, \quad (18)$$

N is the size of α, and Z is the range of κ, i.e., $Z = \{\kappa(c,d) \colon \langle c,d\rangle \in \alpha\}$. Then

$$\mathcal{P}^q(S,T) = \hat{\mathcal{P}}_{\max}(S,T) \subset \mathcal{P}_{\max}(S,T)$$

for every $q > q_0$. In particular, $P^{\Sigma}_{S,T}(\kappa^q)$ belongs to $\mathcal{P}_{\max}(S,T)$ for every $q > q_0$ and, if $\mathcal{P}_{\max}(S,T)$ has only one element, then $P^{\Sigma}_{S,T}(\kappa^q) = P^{IFT}_{S,T}(\kappa)$.

Proof. Theorem 4.3 implies that $\theta = \theta_{\min} = \min\{\varepsilon^{\max}(P) \colon P \in \mathcal{P}(S,T)\}$.

To see that $\hat{\mathcal{P}}_{\max}(S,T) \subset \mathcal{P}_{\max}(S,T)$ take objects $\hat{P} \in \hat{\mathcal{P}}_{\max}(S,T)$ and $P \in \mathcal{P}_{\max}(S,T)$. Then, for $\eta > \theta$, $|\hat{P}_\eta| \le |P_\eta| = 0$ (as $\varepsilon^{\max}(P) = \theta$). Therefore, $\varepsilon^{\max}(\hat{P}) \le \theta$ and so, $\hat{P} \in \mathcal{P}_{\max}(S,T)$.

To see that $\mathcal{P}^q(S,T) \subset \hat{\mathcal{P}}(S,T)$ take a $\hat{P} \in \mathcal{P}^q(S,T)$ and an arbitrary $P \in \mathcal{P}(S,T)$. We need to show that, for arbitrary η, $|\hat{P}_\eta| \le |P_\eta|$. By way of contradiction, assume that this is not the case and let η be the largest number with $|\hat{P}_\eta| > |P_\eta|$. Then

$$[\varepsilon^q]^q(\hat{P}) \ge \sum_{e \in \hat{P}_\eta} [w(e)]^q \ge \sum_{e \in P_\eta} [w(e)]^q + \eta^q > \sum_{e \in \mathrm{bd}(P)} [w(e)]^q = [\varepsilon^q]^q(P),$$

since $\eta^q > N\hat{\theta}^q \ge \sum_{e \in \mathrm{bd}(P) \setminus P_\eta} [w(e)]^q$, where $\hat{\theta}$ is the largest $\theta \in Z$ with $\theta < \eta$. (The first inequality follows from the choice of q_0, since $N = \delta^{q_0} < (\eta/\hat{\theta})^q$.) However, $[\varepsilon^q]^q(\hat{P}) > [\varepsilon^q]^q(P)$ contradicts $\hat{P} \in \mathcal{P}^q(S,T)$.

---

ordered by the lexicographical order $\le_{lex}$, that is, $f <_{lex} g$ provided $f(m) < g(m)$, where $m = \max\{\eta \colon f(\eta) \ne g(\eta)\}$.

To see that $\hat{\mathcal{P}}(S,T) \subset \mathcal{P}^q(S,T)$, take a $\hat{P} \in \mathcal{P}^q(S,T)$ and an arbitrary $P \in \mathcal{P}(S,T)$. We need to show that $[\varepsilon^q]^q(\hat{P}) \leq [\varepsilon^q]^q(P)$. Using the fact that $|\hat{P}_\eta| \leq |P_\eta|$ for every η, an easy recursion in decreasing order over $\eta \in Z$ shows that $\sum_{e\in\hat{P}_\eta}[w(e)]^q \leq \sum_{e\in P_\eta}[w(e)]^q$ for every η. In particular, $[\varepsilon^q]^q(\hat{P}) = \sum_{e\in\hat{P}_0}[w(e)]^q \leq \sum_{e\in\hat{P}_0}[w(e)]^q = [\varepsilon^q]^q(P)$.

The last comment follows from the fact that $P^{IFT}_{S,T}(\kappa) \in \mathcal{P}_{\max}(S,T)$, proved in Theorem 4.5. ∎

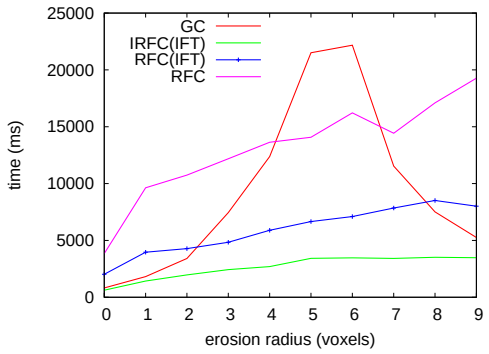
Notice that, in general, the families $\hat{\mathcal{P}}_{\max}(S,T)$ and $\mathcal{P}_{\max}(S,T)$ need not be equal. If $\hat{\mathcal{P}}_{\max}(S,T) \subsetneq \mathcal{P}_{\max}(S,T)$, then the optimization problem MP(ε^q) for $q > q_0$ in place of MP($\varepsilon^{\max}$) may reduce the number of ties, that is, the minimizing family $\mathcal{P}_{\max}(S,T)$ to $\hat{\mathcal{P}}_{\max}(S,T)$. The natural algorithm for solving MP(ε^q) is that of the classical graph cut, which is computationally expensive (as indicated above).

5.2 GC vs FC algorithms: experimental comparison

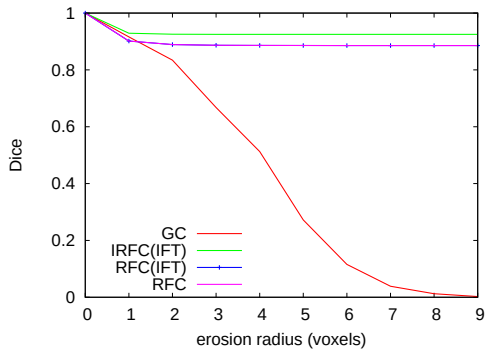
In this section, we describe the experiments that were designed to verify and demonstrate the main differences between FC and GC delineation algorithms discussed in Section 5.1: speed and robustness of FC versus GC and the GC shrinking problem.

We compared four algorithms: GC_{sum} using the min-cut/max-flow algorithm [11]; RFC algorithm, standard "old-fashioned" implementation [44]; RFC-IFT, implemented by using the IFT approach, as indicated in Section 4.2; GC_{max} — IRFC-IFT algorithm— which iteratively refines the choice of output among all energy minimizers from $\mathcal{P}(S,T)$, [19, 29]. The last two algorithms are described in detail in Section 4.

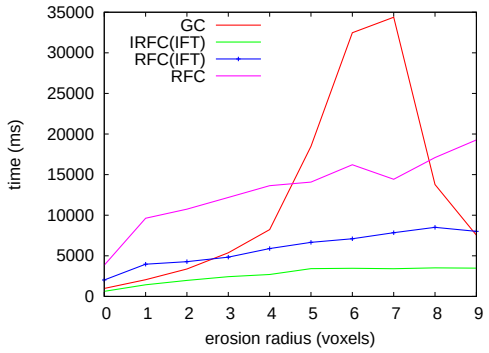
Simulated MR image phantom data from the BrainWeb repository [47] pertaining to 20 different normal patient anatomies were utilized for our evaluation. We used the T1 data sets, since separation of white matter (WM) and grey matter (GM) tissue regions is less challenging in these images than in images of other protocols such as T2 or PD. The parameters for the simulated T1 acquisition were as follows: spoiled FLASH sequence with TR=22ms and TE=9.2ms, flip angle = 30°, voxel size = $1 \times 1 \times 1\text{mm}^3$, noise = 3%, and background non-uniformity = 20%. In these simulated data sets, true segmentations are known, since the simulations were done starting with known anatomy. In the experiments we used a PC with an AMD Athlon 64 X2 Dual-Core Processor TK-57, 1.9 GHz, 2×256 KB L2 cache, and 2 GB



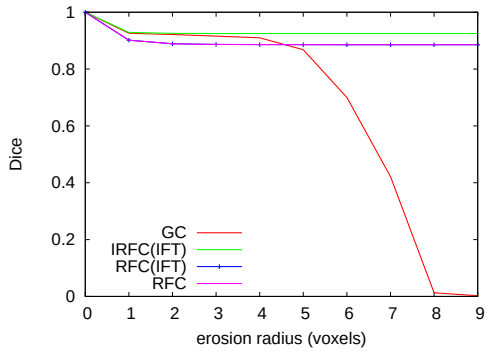
(a) Running time for $w(c,d) = (\kappa(c,d))^q$ with $q = 1$



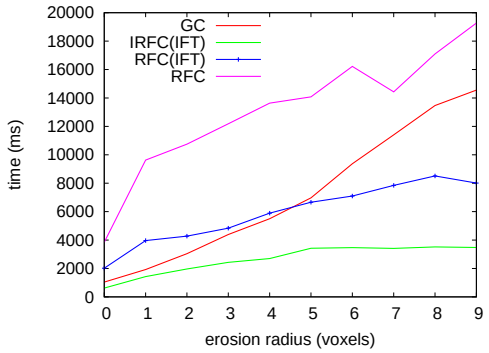
(b) Accuracy for $w(c,d) = (\kappa(c,d))^q$ with $q = 1$



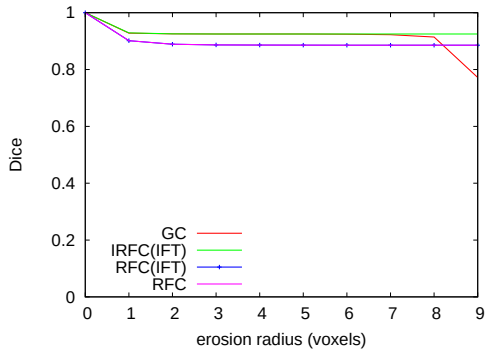
(c) Running time for $w(c,d) = (\kappa(c,d))^q$ with $q = 5$



(d) Accuracy for $w(c,d) = (\kappa(c,d))^q$ with $q = 5$



(e) Running time for $w(c,d) = (\kappa(c,d))^q$ with $q = 30$

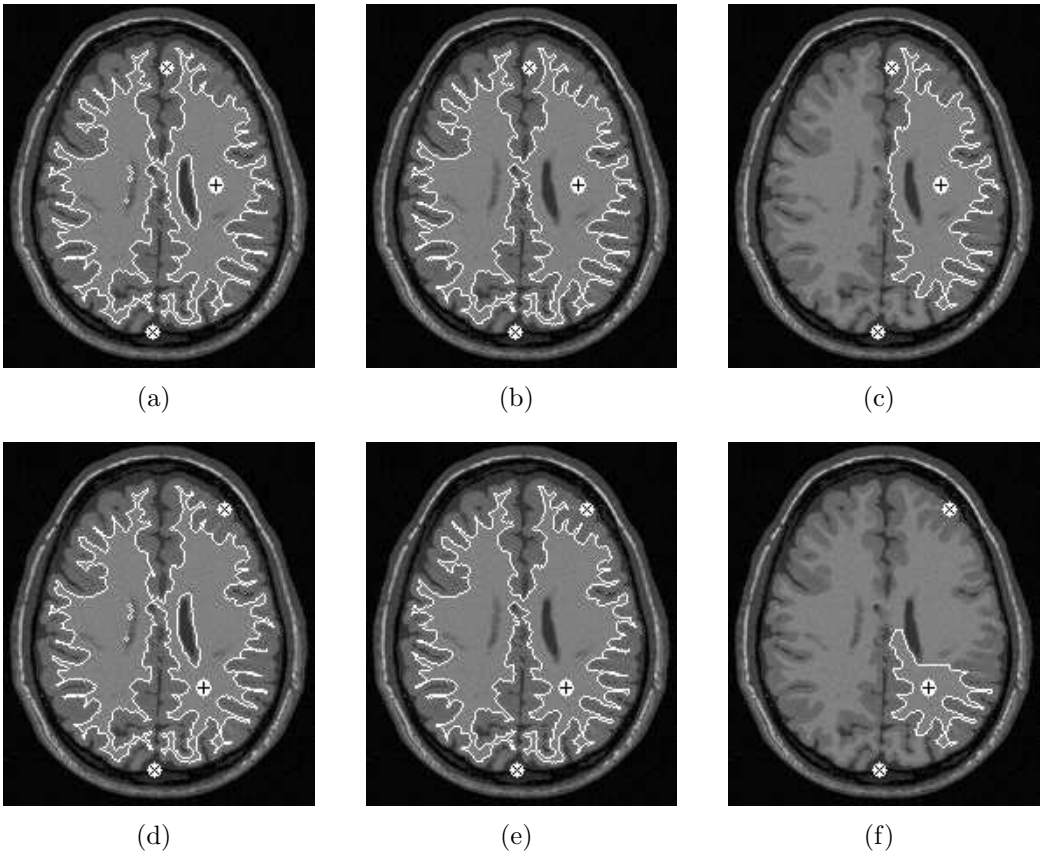


(f) Accuracy for $w(c,d) = (\kappa(c,d))^q$ with $q = 30$

Figure 5: Time and accuracy graphs for segmenting WM for GC algorithm and FC methods.

DDR2 of RAM.

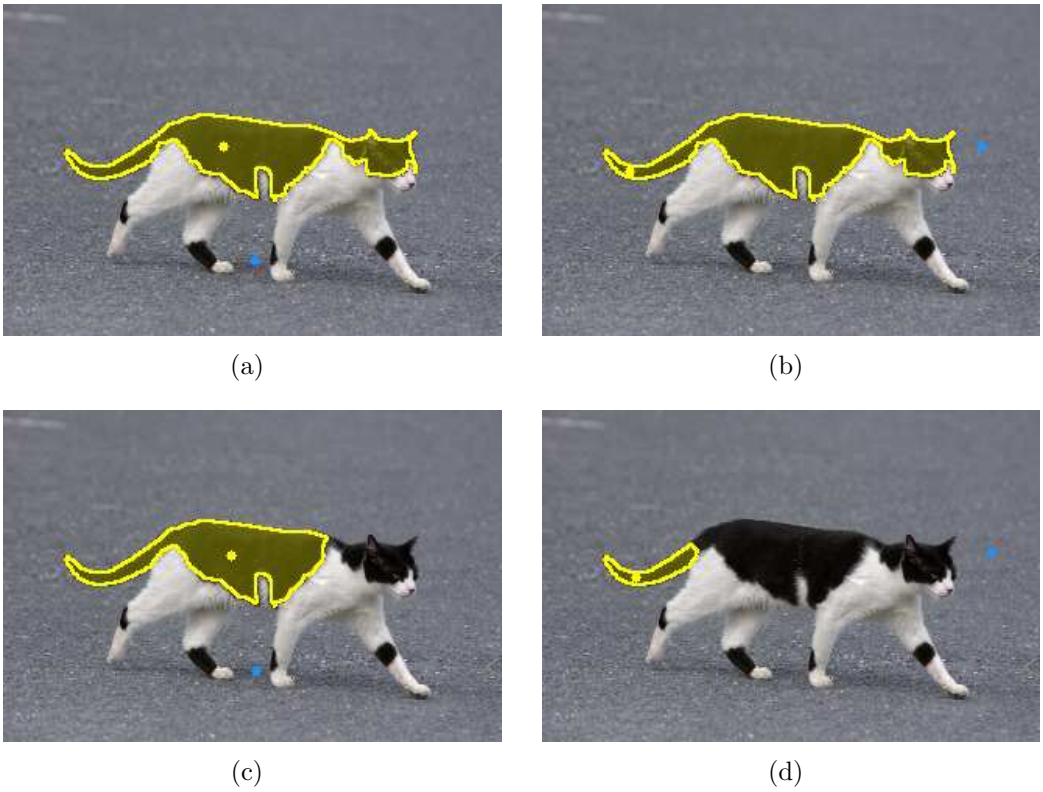
The affinity function $\kappa(c,d)$ was defined as follows. Each given image $I = \langle \mathcal{C}, f\rangle$ was filtered by a Gaussian function G_k with mean μ_k and standard deviation σ_k, where $k \in \{\text{WM,GM}\}$, separately for WM and GM, to produce two new images $I_k = \langle \mathcal{C}, f_k\rangle$, where $f_k(c) = G_k(f(c))$ for any $c \in C$. Parameters μ_k and σ_k were chosen appropriately separately for WM and GM. In each image I_k, the appropriate tissue k was segmented by using each algorithm, where the affinity was defined for I_k via formula (1). For the GC_{sum}



(a) (b) (c)

(d) (e) (f)

Figure 6: (a)&(d) The same segment is obtained independent of seed location with the RFC algorithm. (b)&(e) The results of IRFC were also not affected by these seed translations. (c)&(f) However, GC segmentation may be sensitive to the seed location, when the power q is not large enough.



(a) (b)

(c) (d)

Figure 7: (a)&(b) Results from IRFC (GC_{max}) segmentation for two different seed locations indicated by the dots. (c)&(d) Results from GC (GC_{sum}, with $q = 1$) segmentation for the same seed locations as in (a)&(b).

algorithm, the weight function $w(c, d) = (\kappa(c, d))^q$ was used with $q = 1, 5$, and 30. The rationale for this choice of w was that, according to Theorem 5.3 (compare also [29]), the output of GC_{sum} converges to the output of GC_{max} when q goes to infinity. Changing q does not influence the output of the FC algorithms [15, theorems 3 and 5]. Thus, for large q, the outputs of these two forms of algorithms should be similar. Different sets of seeds were generated, and fed automatically as input to the algorithms, by applying different degrees of erosion operations to the known true segmentation binary images. The degree of erosion is expressed by the erosion radius — the larger the radius, the greater is the degree and smaller is the seed set. As the radius of erosion increases, we may therefore expect lower delineation accuracy.

The graphs in Figure 5 summarize our experimental results. (Only the graphs for WM are shown. The graphs for GM are similar.) Figures 5(a), (c), and (e) display the run time of each algorithm, as a function of the erosion radius, averaged over the 20 images, for $q = 1, 5$, and 30, respectively. Similarly, Figures 5(b), (d), and (f) demonstrate the accuracy of the algorithms expressed in terms of the Dice coefficient as a function of the erosion radius, for $q = 1, 5$, and 30, respectively. In Figure 6, some slices of the given image I are displayed overlaid with the seed sets and the resulting segmentations for different algorithms. A non-medical example is shown in Figure 7 to illustrate the differences in the sensitivity of the algorithms to the selection of seeds. Below, we will examine the efficiency, precision, and accuracy of the different algorithms.

As to the *efficiency* of delineations, as described earlier, the theoretical worst case run times for GC, standard RFC, RFC-IFT, and $\mathrm{GC}_{\max}$ are $O(|C|^{2.5})$ (or $O(|C|^3)$), $O(|C|^2)$, $O(|C|)$ (or $O(|C| \ln |C|)$), and $O(|C|)$ (or $O(|C| \ln |C|)$), respectively. These are borne out and supported by the graphs in Figures 5(a)-(c). The time curve of GC should be properly interpreted in conjunction with its accuracy curve. It shows an unstable time behavior — the peak represents the high computational cost of GC and its drop off denotes the shrinkage problem or finding small cuts when larger erosions (and hence small seed sets) are used that lead to highly inaccurate segmentations. That is, in order to obtain fast segmentations via GC, in many situations in these data sets, we need to specify seed sets that are close to the true segmentation.

As to the *robustness* of delineations, GC is sensitive to the position of the seed sets, not just their size, as illustrated in Figures 6 and 7, as it has a bias toward small boundaries unlike the FC family. This implies that, in an interactive setup, its precision may suffer owing to intra- and inter-operator variability in seed specification. If seeds are selected automatically, the lack of robustness implies lower accuracy as we decrease the seed set to insure that it is properly included within an object region, for example, guided by a statistical model of the object. One way to circumvent this problem is to increase the value of q, which will bring the accuracy of $\mathrm{GC}_{\mathrm{sum}}$ close to that of $\mathrm{GC}_{\max}$. However, this will further add to the computational cost of $\mathrm{GC}_{\mathrm{sum}}$. In practice, in addition to the burden of the power of computation, large q may also cause integer overflow and otherwise storage issues. Note that an efficient implementation of $\mathrm{GC}_{\mathrm{sum}}$ requires integer weight values, but choices such as 255^5 already cause integer overflow in a 32-bit machine. Thus $(\kappa(c, d))^q$

must be subsequently normalized within a valid integer value range to avoid overflow. Thus it is possible to only approximate the theoretical results because an exact match is not possible due to loss of information during normalization. However, in our experiments, these effects were minimal, since GC and IRFC showed almost the same results for $q = 30$.

As to the *accuracy* of delineations, it is clear from Figures 5(b), (d), and (f) that, as the seed set becomes smaller, the delineations by GC become less accurate, mainly due to the shrinkage issue. Even when $q = 30$, this effect is noticeable. The accuracy of FC algorithms, however, is more or less completely independent of the seed set size. In the best case, FC algorithms require only two seeds — one inside and another outside the object, especially when $\kappa(c, d)$ is strictly lower across the object's boundary than elsewhere in the image. However, GC may face the shrinkage problem even in such ideal scenarios. In summary, the higher efficiency, the high degree of independence of the efficiency, precision, and accuracy of FC algorithms to the size and position of seed sets, and their results not losing precision when using integer arithmetic are the strongest features and advantages of FC algorithms over GC methods.

6 Concluding remarks

Focusing on FC and GC algorithms, we have presented a unified mathematical theory for describing these pI approaches as energy optimization methods. We have taken a graph and topological approach to present these combinatorial optimization analyses. In the process, we have arrived at a linear time algorithm $\mathrm{GC}_{\max}$ that retains the robustness and accuracy of the erstwhile IRFC algorithm but improves upon its efficiency. The unifying theory established the limiting relationship between GC and FC and has helped us in delineating the similarities and differences among these methods. We have also demonstrated the forecast theoretical behavior via experiments conducted on the 20 BrainWeb MR image data sets.

The results demonstrate that, while the theoretical underpinning for GC and FC are similar, the subtle differences that remain between them are vitally responsible for their different behavior. The major differences are the dependence of GC's results on the size and position of the seed set compared to a relative independence of FC's results of these parameters. Traceable exactly to those characteristics, GC suffers in computational efficiency (time

and storage), precision (repeatability), and accuracy compared to FC algorithms. Also due to these characteristics, there is a complex interplay among GC's efficiency, precision, and accuracy.

A couple of issues we have not addressed in this paper are perhaps worth investigating further. First, the simultaneous segmentation of multiple (more than 2) objects. This leads to an NP-hard problem in the case of GC, while posing no difficulties for FC in retaining its properties relating to its run time behavior and precision (or robustness). We did not consider the multiple object case in this paper because of the theoretical challenges these differences bring about. (However, a multi-object version of the $\mathrm{GC}_{\max}$ algorithm is indicated in the text.) Second, the graph formulation of GC seems more natural than FC in handling the unary and binary relations implied by the object feature (intensity/texture) and homogeneity properties, respectively. Whether an appropriate FC-type theory can be developed on such a graph remains to be seen. Finally, to extend these studies to a scale-based setting and to vectorial images may allow us to harness the full power of these different frameworks.

A Appendix

Example A.1 For the energies ε_q and $E_{q,q}$ with $q \in (1, \infty]$ it is possible that $\mathcal{P}^F_{\hat{\theta}_{\min}}(S,T)$ and $\mathcal{P}^H_{\theta_{\min}}(S,T)$ are disjoint and that $\bar{x} \in \mathcal{P}^H(S,T)$ associated with $x_{\min} \in \mathcal{P}^F_{\hat{\theta}_{\min}}(S,T)$ does not belong to $\mathcal{P}^H_{\theta_{\min}}(S,T)$.

Proof. Take $C = \{s, c, d, t\}$, where s is a foreground seed and t is a background seed, that is, $S = \{s\}$ and $T = \{t\}$. Consider a graph on C with just three symmetric edges, $\{s, c\}$, $\{c, d\}$, and $\{d, t\}$ (so, with six directed edges) with the respective weights 1, v, and v for $v > 1$ to be determined. Then, $\mathcal{P}^F(S,T)$ consists of all fuzzy sets $x_{y,z} \colon C \to [0,1]$ with $y, z \in [0,1]$, where $x_{y,z}(s) = 1$, $x_{y,z}(c) = y$, $x_{y,z}(d) = z$, and $x_{y,z}(t) = 0$.

First fix a $q \in (1, \infty)$. Then, $E_{q,q}(x_{y,z}) = 2[(1-y)^q + v^q|y-z|^q + v^q z^q]$ is a function of two variables, y and z. It has precisely one minimum[9] at

[9]This can be found by simple multivariable calculus. First notice, that both second partial derivatives, $\frac{\partial^2}{\partial z^2} E_{q,q}(x_{y,z}) = 2q(q-1)v^q[|y-z|^{q-2} + z^{q-2}]$ and $\frac{\partial^2}{\partial y^2} E_{q,q}(x_{y,z}) = 2q(q-1)[(1-y)^{q-2} + v^q|y-z|^{q-2}]$ are positive, so the function $E_{q,q}$ is convex and it can have only one global minimum. For $y \geq z$, $\frac{\partial}{\partial z} E_{q,q}(x_{y,z}) = 2[-qv^q(y-z)^{q-1} + qv^q z^{q-1}]$ equals 0 when $(y-z)^{q-1} = z^{q-1}$, that is, when $y = 2z$. Similarly, the other derivative

$z_q = (v^{q/(q-1)} + 2)^{-1}$ and $y_q = 2(v^{q/(q-1)} + 2)^{-1}$. Thus, $\mathcal{P}^F_{\hat{\theta}_{\min}}(S,T) = \{x_{y_q,z_q}\}$, leading to $x_{\min} = x_{y_q,z_q}$. Now, if $v \in (1, 2^{(q-1)/q})$, then $1 < v^{q/(q-1)} < 2$ and we have $0 < z_q < 0.5 < y_q < 1$, leading to $\bar{x}$ with $\bar{x}(s) = \bar{x}(c) = 1$ and $\bar{x}(d) = \bar{x}(t) = 0$. But this implies that $E_{q,q}(\bar{x}) = v^q > 1 = E_{q,q}(\chi_{\{s\}})$, so indeed $\bar{x} \notin \mathcal{P}^H_{\theta_{\min}}(S,T)$.

To see that the same example works for $q = \infty$, fix a $v \in (1, 2^{1/2})$. Then, for every $q > 2$ and $y, z \in \mathbb{R}$, we have $||F(x_{y,z})||_q \geq ||F(x_{y_q,z_q})||_q$. Taking the limit, as $q \to \infty$, gives $||F(x_{y,z})||_\infty \geq ||F(x_{y_\infty,z_\infty})||_\infty$, where $z_\infty = \lim_{q\to\infty} z_q = (v+2)^{-1}$ and $y_\infty = 2z_\infty$. Then, similarly as above, $\mathcal{P}^F_{\theta_{\min}}(S,T) = \{x_{y_\infty,z_\infty}\}$, leading to $x_{\min} = x_{y_q,z_q}$ and $\bar{x}$ with $\bar{x}(s) = \bar{x}(c) = 1$ and $\bar{x}(d) = \bar{x}(t) = 0$. But this implies that $\varepsilon_\infty(\bar{x}) = v > 1 = \varepsilon_\infty(\chi_{\{s\}})$, so once again $\bar{x} \notin \mathcal{P}^H_{\theta_{\min}}(S,T)$. ∎

Proof of Theorem 4.6. We start with proving that $P^{IFT}_{S,T} \in \mathcal{F}_M(S,W)$. Let $\mathbb{F}$ be the OPF returned by $\mathrm{GC}_{\max}$, so that we have $P^{IFT}_{S,T} = P(S,\mathbb{F})$. We will find an MSF $\hat{\mathbb{F}}$ relative to W which returns the same object, that is, such that $P(S,\hat{\mathbb{F}}) = P(S,\mathbb{F})$.

Recall, that the Kruskal's algorithm creates MSF $\hat{\mathbb{F}} = \langle C, \hat{E}\rangle$ as follows:

- it lists all edges of the graph in a queue Q such that their weights form a decreasing sequence;
- it removes consecutively the edges from Q, adding to $\hat{E}$ those, whose addition creates in the expanded $\hat{\mathbb{F}} = \langle C, \hat{E}\rangle$ neither a cycle nor a path between different spels from W; other edges are discarded.

This schema has a leeway in choosing the order of the edges in Q: those that have the same weight can be ordered arbitrarily.

Let B be the boundary of $P(S,\mathbb{F})$, $B = \mathrm{bd}(P(S,\mathbb{F}))$. Assume, that we create the list Q in such a way that, among the edges with the same weight, all those that do not belong to B precede all those that belong to B. We will show that Kruskal's algorithm with Q so chosen, indeed returns MSF $\hat{\mathbb{F}}$ with $P(S,\hat{\mathbb{F}}) = P(S,\mathbb{F})$.

$\frac{\partial}{\partial y}E_{q,q}(x_{y,z}) = 2[-q(1-y)^{q-1} + qv^q(y-z)^{q-1}]$ equals 0 when $(1-y)^{q-1} = v^q(y-z)^{q-1}$, which, with $y = 2z$, leads to $\left(\frac{1-2z}{2z-z}\right)^{q-1} = v^q$ and $\frac{1}{z} - 2 = v^{q/(q-1)}$. So, $z = (v^{q/(q-1)}+2)^{-1}$ and $y = 2(v^{q/(q-1)} + 2)^{-1}$ minimize $E_{q,q}$ on $[0,1]\times[0,1]$, since for $z > y$, the derivative $\frac{\partial}{\partial z}E_{q,q}(x_{y,z}) = 2[qv^q(z-y)^{q-1} + qv^q z^{q-1}]$ never equals 0.

Clearly, by the power of Kruskal's algorithm, the returned $\hat{\mathbb{F}} = \langle C, \hat{E}\rangle$ will be MSF relative to W. We will show that $\hat{E}$ is disjoint with B. This easily implies the equation $P(S, \hat{\mathbb{F}}) = P(S, \mathbb{F})$.

To prove that $\hat{E}$ is disjoint with B, choose an edge $e = \{c, d\} \in B$. Consider the step in Kruskal's algorithm when we remove e from Q. We will argue, that adding e to the already existing part of $\hat{E}$ would add a path from S to T, which implies that e would not be added to $\hat{E}$.

Let p_c and p_d be the paths in $\mathbb{F}$ from W to c and d, respectively. By symmetry, we can assume that $c \in V \setminus P(S, \mathbb{F}) = P(T, \mathbb{F})$ and $d \in P(S, \mathbb{F})$. We will first show that

$$\mu(p_c) \geq w_e \text{ and } \mu(p_d) \geq w_e. \tag{19}$$

Indeed, if $\mu(p_c) > \mu(p_d)$, then $w_e \leq \mu(p_d)$, since otherwise $\mu(d, S) = \mu(p_d) < \min\{\mu(p_c), w_e\} \leq \mu(d, T)$, implying that d belongs to the RFC object $P_{T,S} \subset P(T, \mathbb{F})$, which is disjoint with $P(S, \mathbb{F})$. Similarly, if $\mu(p_c) < \mu(p_d)$, then $w_e \leq \mu(p_c)$, since otherwise $\mu(c, T) = \mu(p_c) < \min\{\mu(p_d), w_e\} \leq \mu(c, S)$, implying that c belongs to the RFC object $P_{S,T} \subset P(S, \mathbb{F})$. Finally, assume that $\mu(p_c) = \mu(p_d)$. Then $w_e < \mu(p_c) = \mu(p_d)$, since otherwise $\mathrm{GC}_{\max}$ (during the execution of lines 6-8 for c and d) would reassign d to $P(T, \mathbb{F})$, which is disjoint with $P(S, \mathbb{F})$. So, (19) is proved.

Next, let $E' = \{e' \in E\colon w_{e'} \geq w_e\} \setminus B$. Then, every edge in E' is already considered by the Kruskal's algorithm by the time we remove e from Q. In particular, $\hat{E} \cap E'$ is already constructed. We claim, that there is a path $\hat{p}_d$ in $\hat{G} = \langle C, \hat{E} \cap E'\rangle$ from S to d.

Indeed, the component of d in $\hat{G}$ must intersect S, since otherwise there is an edge $\hat{e}$ in p_d (so, in E') only one vertex of which intersects this component. But this means that $\hat{e} \in E'$ would have been added to $\hat{E}$, which was not the case. So, indeed, there is a path $\hat{p}_d$ in $\hat{G}$ from S to d. Similarly, there is a path $\hat{p}_c$ in $\hat{G}$ from T to c. But this means that adding e to $\hat{E}$ would create a path from S to T, which is a forbidden situation. Therefore, indeed, Kruskal's algorithm discards e, what we had to prove. This completes the argument for $P^{IFT}_{S,T} \in \mathcal{F}_M(S, W)$.

The inclusion $\mathcal{F}_M(S, W) \subset \mathcal{F}^{IRFC}(S, W)$ is proved in [3, proposition 8]. Thus, to finish the proof, we need to show that $\mathcal{F}_M(S, W) \subset \mathcal{P}_\theta(S, T)$. So, fix a $P \in \mathcal{F}_M(S, W)$. Then, there is an MSF $\mathbb{F} = \langle C, E'\rangle$ with respect to W for which $P = P(S, \mathbb{F})$. Clearly, $P = P(S, \mathbb{F}) \in \mathcal{P}(S, T)$. So, to finish the proof, it is enough to show that $\varepsilon^{\max}(P) \leq \theta_{\min} = \mu(S, T)$.

By way of contradiction, assume that this is not the case. Then, there exists an edge $e = \{c, d\} \in E$ with $c \in P = P(S, \mathbb{F})$ and $d \in V \setminus P = P(T, \mathbb{F})$ for which $w_e > \theta_{\min} = \mu(S, T)$. Let p_c and p_d be the paths in $\mathbb{F}$ from W to c and d, respectively. Then either $\mu(p_c) < w_e$ or $\mu(p_d) < w_e$, since otherwise the path p starting with p_c, followed by e, and then by p_d is a path from S to T with $\mu(p) = w_e > \mu(S, T)$, a contradiction.

Assume that $\mu(p_c) < w_e$. Then $p_c = \langle c_1, \ldots, c_k \rangle$ with $k > 1$ and the edge $e' = \{c_{k-1}, c_k\}$ has weight $\leq \mu(p_c) < w_e$. But then $\hat{\mathbb{F}} = \langle C, \hat{E} \rangle$ with $\hat{E} = E' \cup \{e\} \setminus \{e'\}$ is a spanning forest rooted at W with $\sum_{e \in \hat{E}} w(e) = \sum_{e \in E'} w(e) + w_e - w_{e'} > \sum_{e \in E'} w(e)$, what contradicts maximality of $\mathbb{F}$. This completes the proof of the theorem.

References

[1] E.D. Angelini, Y. Jin, and A.F. Laine, "State-of-the-art of levelset methods in segmentation and registration of medical imaging modalities," in: Jasjit Suri, David L. Wilson and Swamy Laximinarayan, (editors), *Handbook of Medical Image Analysis: Advanced Segmentation and Registration Models*, Kluwer Academic Publishers: New York, NY, 2004.

[2] R. Audigier and R.A. Lotufo, "Duality between the watershed by image foresting transform and the fuzzy connectedness segmentation approaches," in Proceedings of the 19th Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI06), Manaus (AM), Brazil, 2006.

[3] R. Audigier and R.A. Lotufo, "Relationships between some watershed definitions and their tie-zone transforms," *Image Vision Comput.* **28** (2010), 1472–1482.

[4] G. Bertrand, "On topological watersheds," *J. Math. Imaging Vis.* **22**(2-3) 2005, 217–230.

[5] S. Beucher, "The watershed transformation applied to image segmentation," in: *10th Pfefferkorn Conf. Signal and Image Processing in Microscopy and Microanalysis* (1992), 299–314.

[6] Y. Boykov and G. Funka-Lea: "Graph Cuts and Efficient N-D Image Segmentation," International Journal of Computer Vision, 70: 109-131, 2006.

[7] Y. Boykov and M. Jolly: "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in Proceedings of ICCV, Part I, 105-112, 2001.

[8] Y. Boykov and V. Kolmogorov: "Computing geodesics and minimal surfaces via graph cuts," International Conference on Computer Vision, I: 26-33, 2003.

[9] Y. Boykov and V. Kolmogorov: "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," IEEE Transactions on Pattern Analysis and Machine Intelligence, 26:1124-1137, 2004.

[10] Y. Boykov, V. Kolmogorov, D. Cremers, and A. Delong: "An Integral Solution to Surface Evolution PDEs via Geo-Cuts," International Conference on Computer Vision, LNCS 3953, vol.III, 409-422, 2006.

[11] Y. Boykov and O. Veksler: "Graph cuts in vision and graphics: Theories and applications," in: N.Paragios, Y. Chen, and O. Faugeras, (Eds.), *Handbook of Mathematical Models in Computer Vision*, Springer-Verlag (2006), 79–96.

[12] Y. Boykov, O. Veksler, and R. Zabih, Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Machine Intell.* **23**(11) (2001), 1222–1239.

[13] B.M. Carvalho, C.J. Gau, G.T. Herman, and Y.T. Kong, Algorithms for fuzzy segmentation. *Pattern Analysis and Applications* **2** (1999), 73–81.

[14] B.M. Carvalho, G.T. Herman, and Y.T. Kong, Simultaneous fuzzy segmentation of multiple objects. *Discrete Applied Mathematics* **151** (2005), 65–77.

[15] K.C. Ciesielski and J.K. Udupa, Affinity functions in fuzzy connectedness based image segmentation I: Equivalence of affinities, *Computer Vision and Image Understanding* **114** (2010), 146–154.

[16] K.C. Ciesielski and J.K. Udupa, Affinity functions in fuzzy connectedness based image segmentation II: Defining and recognizing truly novel affinities, *Computer Vision and Image Understanding* **114** (2010), 155–166.

[17] K.C. Ciesielski and J.K. Udupa: "Region-based segmentation: fuzzy connectedness, graph cut, and other related algorithms," in: *Biomedical Image Processing* (T.M. Deserno, editor), Springer-Verlag, 2011, 251–278.

[18] K.C. Ciesielski and J.K. Udupa, A framework for comparing different image segmentation methods and its use in studying equivalences between level set and fuzzy connectedness frameworks, *Computer Vision and Image Understanding* **115** (2011), 721–734.

[19] K.C. Ciesielski, J.K. Udupa, P.K. Saha, and Y. Zhuge, Iterative Relative Fuzzy Connectedness for Multiple Objects, Allowing Multiple Seeds. *Computer Vision and Image Understanding* **107**(3) (2007), 160–182.

[20] C. Couprie, L. Grady, L. Najman, and H. Talbot, "Anisotropic Diffusion Using Power Watersheds," in International Conference on Image Processing (ICIP10), Sep. 2010, pp. 4153–4156.

[21] Camille Couprie, Leo Grady, Laurent Najman, and Hugues Talbot, "Power Watersheds: A Unifying Graph-Based Optimization Framework," *IEEE Trans. Pattern Anal. Machine Intell.* **33**(7) (2011), 1384–1399.

[22] Cousty J., Bertrand G., Najman L., and Couprie M., "Watershed-Cuts:MinimumSpanning Forests and the Drop of Water Principle," *IEEE Trans. Pattern Anal. Machine Intell.* **31**(8) (2009), 1362–1374.

[23] Cousty J., Bertrand G., Najman L., and Couprie M., "Watershed Cuts: Thinnings, Shortest Path Forests, and Topological Watersheds," *IEEE Trans. Pattern Anal. Machine Intell.* **32**(5) (2010), 925–939.

[24] A.X. Falcão, J. Stolfi, and R.A. Lotufo: The image foresting transform: Theory, algorithms, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.* **26(1)**, 19-29 (2004).

[25] X. Fan, J. Yang, and L. Cheng, A novel segmentation method for MR brain images based on fuzzy connectedness and FCM. *Lecture Notes in Computer Science* **3613** (2005), 505–513.

[26] G.T. Herman and B.M. Carvalho, Multiseeded segmentation using fuzzy connectedness. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23** (2001), 460–474.

[27] K. Li, X. Wu, D.Z. Chen, and M. Sonka, "Optimal surface segmentation in volumetric images – A graph-theoretic approach," *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(1), 119134 (2006).

[28] R. Malladi, J. Sethian, and B. Vemuri: "Shape modeling with front propagation: A level set approach," IEEE Transactions on Pattern Analysis and Machine Intelligence, 17:158-175, 1995.

[29] P.A.V. Miranda and A.X. Falcão: "Links Between Image Segmentation based on Optimum-Path Forest and Minimum Cut in Graph," Journal of Mathematical Imaging and Vision **35**, 128-142 (2009).

[30] L. Najman, "On the equivalence between hierarchical segmentations and ultrametric watersheds," *Journal of Mathematical Imaging and Vision* **40**(3) (2011), 231–247.

[31] L.G. Nyul, A.X-. Falcao, J.K. Udupa, *Fuzzy-connected 3D image segmentation at interactive speeds*, Graphical Models and Image Processing **64** (2003), 259–281.

[32] J. Park and J. Keller: "Snakes on the Watershed," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**:1201-1205, 2001.

[33] A. Pednekar, I.A. Kakadiaris, Image segmentation based on fuzzy connectedness using dynamic weights. *IEEE Trans Image Process.* **15**(6) (2006), 1555–1562.

[34] A. Rosenfeld, Fuzzy digital topology. *Information and Control* **40** (1979), 76–87.

[35] A. Rosenfeld, On connectivity properties of grayscale pictures. *Pattern Recognition* **16** (1983), 47–50.

[36] A. Rosenfeld, The fuzzy geometry of image subsets. *Pattern Recognition Letters* **2** (1984), 311–317.

[37] P.K. Saha and J.K. Udupa, Relative fuzzy connectedness among multiple objects: Theory, algorithms, and applications in image segmentation. *Computer Vision and Image Understanding* **82**(1) (2001), 42–56.

[38] P.K. Saha and J.K. Udupa, Iterative relative fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. In *Proceedings of IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, Hilton Head, South Carolina 2002, 28–35.

[39] P.K. Saha, J.K. Udupa, and D. Odhner, Scale-Based Fuzzy Connectedness Image Segmentation: Theory, Algorithms, and Validation. *Computer Vision and Image Understanding* **77** (2000), 145–174.

[40] J.A. Sethian, *Fast Marching Methods and Level Sets Methods. Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, Cambridge Univ. Press, 1999.

[41] L. Shafarenko, M. Petrou, and J. Kittler, "Automatic watershed segmentation of randomly textured color images," *IEEE Transactions on Image Processing* **6** (1997), 1530–1544.

[42] J. Shi and J. Malik: "Normalized cuts and image segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, 22:888-905, 2000.

[43] A.K. Sinop and L. Grady: "A seeded image segmentation frame – work unifying graph cuts and random walker which yields a new algorithm," *Proc. of ICCV07*, 2007.

[44] J.K. Udupa and S. Samarasekera, Fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. *Graphical Models and Image Processing* **58**(3) (1996), 246–261.

[45] J.K. Udupa, P.K. Saha, and R.A. Lotufo, Relative fuzzy connectedness and object definition: Theory, algorithms, and applications in image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24** (2002), 1485–1500.

[46] Y. Zhuge, J.K. Udupa, and P.K. Saha, Vectorial scale-based fuzzy connected image segmentation. *Computer Vision and Image Understanding* **101** (2006), 177–193.

[47] BrainWeb repository, http://mouldy.bic.mni.mcgill.ca/brainweb/anatomic_normal_20.html

[48] P.A.V. Miranda, A.X. Falcão, and J.K. Udupa: "Synergistic arc-weight estimation for interactive image segmentation using graphs," *Computer Vision and Image Understanding* **114**(1), 85–99 (2010).

[49] P.A.V. Miranda, A.X. Falcão, and J.K. Udupa: "Cloud Bank: A multiple clouds model and its use in MR brain image segmentation," *Proceedings of the Sixth IEEE International Symposium on Biomedical Imaging from Nano to Macro (ISBI)*, Boston 2009, 506–509.

[50] A.X. Falcão, J.K. Udupa, and F.K. Miyazawa, "An Ultra-Fast User-Steered Image Segementation Paradigm: Live-Wire-On-The-Fly," *IEEE Trans. Med. Imaging* **19**(1) (2000), 55–62.