



HAL
open science

Revisiting the Isoperimetric Graph Partitioning Problem

Sravan Danda, Aditya Challa, B S Daya Sagar, Laurent Najman

► **To cite this version:**

Sravan Danda, Aditya Challa, B S Daya Sagar, Laurent Najman. Revisiting the Isoperimetric Graph Partitioning Problem. IEEE Access, 2019, 7, pp.50636 - 50649. 10.1109/ACCESS.2019.2901094 . hal-01810249

HAL Id: hal-01810249

<https://hal.science/hal-01810249v1>

Submitted on 7 Jun 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Revisiting the Isoperimetric Graph Partitioning Problem

Sravan Danda*, Aditya Challa†, B.S.Daya Sagar‡, and Laurent Najman§

Abstract. Isoperimetric graph partitioning which is also known the Cheeger cut is NP-Hard in its original form. In literature, multiple modifications to this problem have been proposed to obtain approximation algorithms for clustering applications. In the context of image segmentation, a heuristic continuous relaxation to this problem has yielded good quality results. This algorithm is based on solving a linear system of equations involving the Laplacian of the image graph. Further, the same algorithm applied to a maximum spanning tree (MST) of the image graph was shown to produce similar results at a much lesser computational cost. However, the data reduction step (i.e. considering a MST, a much sparser graph compared to the original graph) leading to a faster yet useful algorithm has not been analysed. In this article, we revisit the isoperimetric graph partitioning problem and rectify a few discrepancies in the simplifications of the heuristic continuous relaxation, leading to a better interpretation of what is really done by this algorithm. We then use the Power Watershed (PW) framework to show that is enough to solve the relaxed isoperimetric graph partitioning problem on the graph induced by Union of Maximum Spanning Trees (UMST) with a seed constraint. The UMST has a lesser number of edges compared to the original graph, thus improving the speed of sparse matrix multiplication. Further, given the interest of PW framework in solving the relaxed seeded isoperimetric partitioning problem, we discuss the links between the PW limit of the discrete isoperimetric graph partitioning and watershed cuts. We then illustrate with experiments, a detailed comparison of solutions to the relaxed seeded isoperimetric partitioning problem on the original graph with the ones on the UMST and a MST. Our study opens many research directions which are discussed in the conclusions section.

Key words. Image Segmentation, Isoperimetric Partitioning, Cheeger cut, Spectral Clustering, Power Watersheds.

AMS subject classifications. 90C05, 90C27, 94A08, 94A12

1. Introduction. In this article, we consider the graph partitioning problem stated as - given an edge weighted graph $G = (V, E, w)$ with edge-weights reflecting similarity measure between adjacent nodes, find a ‘suitable’ partition of the finite set V into 2 subsets. There are, of course, several criteria to find a ‘suitable’ partition. One such criterion, which is the focus of this article is that of **isoperimetric partitioning**. This criterion arises from the classic isoperimetric problem - for a fixed area, find a region with minimum perimeter [10].

The isoperimetric graph partitioning problem also known the Cheeger cut problem is NP-Hard [25]. This problem is closely related to total variation (TV) minimization [24, 8] whose role is crucial to many inverse problems in computer vision. The Cheeger cut problem also has links with problems such as ratio cut [30] and normalized cut minimizations [30] which belong to the family of spectral clustering methods [30]. In the recent past, the Cheeger cut

*Systems Science and Informatics Unit, Indian Statistical Institute (sravan8809@gmail.com, <https://sites.google.com/site/sravandanda1988>).

†Systems Science and Informatics Unit, Indian Statistical Institute (aditya.challa.20@gmail.com)

‡Systems Science and Informatics Unit, Indian Statistical Institute (bsdsagar@yahoo.co.uk)

§Université Paris-Est, Laboratoire d'Informatique Gaspard-Monge, Équipe A3SI, ESIEE Paris, France (laurentnajman@esiee.fr, <http://www.laurentnajman.org/>).

38 problem has been of interest to many researchers and multiple approximation algorithms have
 39 been proposed [24, 5, 18, 9, 21] .

40 One such approximation, which is the focus of this paper is a continuous relaxation to this
 41 problem proposed in [21]. Its application to image segmentation problem is discussed in [20].
 42 In cases such as medical image segmentation, due to large data size, extremely fast algorithms
 43 are necessary. Thus, in [19], the authors propose a fast algorithm to obtain an approximate
 44 solution to the isoperimetric graph partitioning problem for medical image segmentation. A
 45 simple overview of the idea presented in [19] is - instead of solving the problem on the original
 46 graph, the authors in [19] construct a maximum spanning tree (MST), and solve the problem
 47 on this graph. This allows for orders of magnitude speed up in the algorithm.

48 However, a few questions remain - (i) What is the basis of the simplifications that led
 49 to the heuristic provided in [20, 21]. (ii) Why would solving the problem on a MST give
 50 similar results to the solution on the original graph. (iii) How close are these solutions? These
 51 questions were answered empirically in [19]. However, to our knowledge, in depth analysis
 52 into this was not done.

53 In this article, we aim to answer the questions highlighted in the previous paragraph by
 54 presenting a detailed analysis. In section 2, we review the isoperimetric graph partitioning
 55 problem and focus on the details of the algorithm provided in [20], rectifying a few discrep-
 56 ancies. This algorithm solves a seeded version of the isoperimetric partitioning problem, and
 57 we discuss the differences with the original formulation. In section 3, we calculate the limit
 58 of minimizers of the isoperimetric graph partitioning problem in the Power Watershed (PW)
 59 framework [27]. Specifically, to calculate the said limit of minimizers, we show that is enough
 60 to solve the isoperimetric graph partitioning problem on the graph induced by **Union of**
 61 **Maximum Spanning Trees (UMST)**. The UMST has a lesser number of edges compared
 62 to the original graph, thus improving the speed of sparse matrix multiplication¹.

63 Given the interest of the PW limit in solving the relaxed seeded isoperimetric graph parti-
 64 tioning problem, in section 4 we discuss the links between the PW limit of the discrete version
 65 (original formulation) and watershed cuts. In section 5, some experiments are performed to
 66 illustrate properties of the solutions to the relaxed seeded isoperimetric graph partitioning
 67 problem on each of the original, the UMST and MST graphs. In section 6, we provide some
 68 prospective research directions building on the ideas from this article.

69 **Remark:** For brevity and clarity, the proofs of all the results are moved to the appendix.

70 **2. Isoperimetric Graph Partitioning Problem.** In this paper, $G = (V, E, w)$ denotes an
 71 edge-weighted graph where V denotes the set of vertices, E denotes the set of edges and
 72 $w : E \rightarrow \mathbb{R}^+$ denotes the weights assigned to each edge reflecting similarity between adjacent
 73 vertices. To simplify the notation, shorter expression w_{ij} is used instead of $w(e_{ij})$ to denote
 74 the weight of the edge between vertices i and j . Let $S \subset V$, then \bar{S} denotes the complement
 75 of the set S .

76 Given a graph, its isoperimetric number is defined by

$$77 \quad (1) \quad h_G = \inf_S \frac{|\partial S|}{\min\{vol(S), vol(\bar{S})\}}$$

¹Given a $n \times n$ matrix with m non-zero entries, the complexity of matrix-vector multiplication is $\mathcal{O}(m+n)$

78 where $S \subset V$, the boundary edges of the set S is denoted by $\partial S = \{e_{ij} \mid i \in S, j \in \bar{S}\}$. The
 79 sum of the edge weights on the boundary is denoted by $|\partial S|$. The volume of the subset S
 80 denoted by $vol(S)$ is given by the cardinality of the set S .

81 The indicator of a set $S \subset V$ is defined as the vector $\mathbf{x}(S) = (x_1, x_2, \dots, x_{|V|})$ where

$$82 \quad (2) \quad x_i = \begin{cases} 1 & \text{if } i \in S \\ 0 & \text{otherwise} \end{cases}$$

83 Also, the Laplacian of an edge-weighted graph G is defined as

$$84 \quad (3) \quad L_{ij} = \begin{cases} \sum_k w_{ik} & \text{if } i = j \\ -w_{ij} & \text{if } e_{ij} \in E \\ 0 & \text{otherwise} \end{cases}$$

85 Using the definitions above, the isoperimetric graph partitioning problem can be stated
 86 as the following optimization problem.

$$87 \quad (4) \quad \begin{aligned} & \text{Find } \arg \min_{\mathbf{x}} \frac{\mathbf{x}^t L \mathbf{x}}{\min\{\mathbf{x}^t \mathbf{1}, (\mathbf{1} - \mathbf{x})^t \mathbf{1}\}} \\ & \text{subject to } x_i \in \{0, 1\} \text{ for all } i \end{aligned}$$

88 Here $\mathbf{1}$ indicates the column vector with all elements equal to 1.

89 Observe that $x_i = 0$ for all i is invalid since the denominator of the cost function is 0.
 90 Similarly, $x_i = 1$ for all i is also invalid. The optimization problem (4) is a NP-hard problem
 91 [25]. To address this, the authors in [20, 21] propose a continuous relaxation of the problem
 92 as in (5) and continue to solve the relaxed problem.

$$93 \quad (5) \quad \begin{aligned} & \text{Find } \arg \min_{\mathbf{x}} \frac{\mathbf{x}^t L \mathbf{x}}{\min\{\mathbf{x}^t \mathbf{1}, (\mathbf{1} - \mathbf{x})^t \mathbf{1}\}} \\ & \text{subject to } x_i \in [0, 1] \text{ for all } i \end{aligned}$$

94 Then, every threshold of the solution \mathbf{x} is examined and a partition (called an *optimal*
 95 *threshold*) with least isoperimetric ratio among them is chosen. The authors of [20, 21] directly
 96 proceed to a modification of problem (5), without discussing it. Before going to their proposal,
 97 we provide hereafter a formal analysis of (5).

98 An issue with optimization problem (5) is the fact that at $\mathbf{x} = t \mathbf{1}$, where $0 < t < 1$, the
 99 cost function takes the minimum value of 0. Thus, optimal solutions to optimization problem
 100 (5) are degenerate.

101 Adding to the above issue, given any partition of $V = S \cup \bar{S}$, one can find a suitable
 102 \mathbf{x} , which when thresholded results in this partition and is close to the optimal solution.
 103 That is, solutions to (5) are not robust. This implies that solving (5) cannot be used to
 104 obtain meaningful partitions. This is stated rigorously in proposition 2.2. Before stating the
 105 proposition, we need to define the notion of ϵ -optimal solution.

106 **Definition 2.1.** Let P be the minimization problem with loss function L on the constraint
 107 set $S \subset \mathbb{R}^n$. Let $\mathbf{x}^* \in S$ be an optimal solution and $\epsilon > 0$ denote a constant. If $\mathbf{x} \in S$ satisfies
 108 $|L(\mathbf{x}) - L(\mathbf{x}^*)| < \epsilon$ then \mathbf{x} is said to be ϵ -optimal for P .

109 **Proposition 2.2.** *Let $\epsilon > 0$ be some constant and let $V = S \cup \bar{S}$ be any partition. Then,*
 110 *given the notation as before, one can find $\mathbf{x} \in [0, 1]^{|V|}$ such that*

$$111 \quad (6) \quad \frac{\mathbf{x}^t L \mathbf{x}}{\min\{\mathbf{x}^t \mathbf{1}, (\mathbf{1} - \mathbf{x})^t \mathbf{1}\}} < \epsilon$$

112 *and an optimal threshold of \mathbf{x} results in the partition $S \cup \bar{S}$.*

113 One way to rectify this is to consider the seeded version of the problem, i.e. set the value
 114 of one of the vertices to be 0. In the context of image segmentation, this can be interpreted
 115 as setting one of the vertices to be in the background of the object. This is the approach
 116 proposed in [20, 21], however the authors of [20, 21] do not clearly state that this problem is
 117 actually different from the unseeded version.

118 Note that one cannot a-priori know a pixel which would belong to the background without
 119 extra knowledge. Thus, in practice, either a seed must be given or one can solve the problem
 120 for all possible seeds and pick the best solution (an approach not very practical).

121 The relaxed seeded isoperimetric graph partitioning problem is stated as

$$122 \quad (7) \quad \begin{aligned} & \text{Find } \arg \min_{\mathbf{x}} \frac{\mathbf{x}^t L \mathbf{x}}{\min\{\mathbf{x}^t \mathbf{1}, (\mathbf{1} - \mathbf{x})^t \mathbf{1}\}} \\ & \text{subject to} \quad x_j = 0 \text{ for some } j \\ & \quad \quad \quad x_i \in [0, 1], \text{ for all } i \neq j \end{aligned}$$

123 which can equivalently be stated as

$$124 \quad (8) \quad \begin{aligned} & \text{Find } \arg \min_{\mathbf{x}} \frac{\mathbf{x}^t L \mathbf{x}}{\mathbf{x}^t \mathbf{1}} \\ & \text{subject to} \quad x_j = 0 \text{ for some } j \\ & \quad \quad \quad x_i \in [0, 1] \text{ for all } i \neq j \\ & \quad \quad \quad \mathbf{x}^t \mathbf{1} \leq \frac{|V|}{2} \end{aligned}$$

125 Note that using the constraint $x_j = 0$ and slack variables, the above problem can be further
 126 simplified to

$$127 \quad (9) \quad \begin{aligned} & \text{Find } \arg \min_x \mathbf{x}_{-j}^t L_{(-j, -j)} \mathbf{x}_{-j} \\ & \text{subject to} \quad (\mathbf{x}_{-j})_i \in [0, 1] \text{ for all } i \\ & \quad \quad \quad \mathbf{x}_{-j}^t \mathbf{1} = \frac{|V|}{2} \end{aligned}$$

128 where $L_{(-j, -j)}$ is the Laplacian of the graph with j^{th} column and row removed, and \mathbf{x}_{-j} is the
 129 vector with j^{th} entry removed. Optimization problem (9) is henceforth referred to the *relaxed*
 130 *seeded isoperimetric partitioning problem*.

131 Using the idea of Lagrange multipliers, one can find the solution to the above problem as
 132 proposed in [21] by solving

$$133 \quad (10) \quad L_{(-j, -j)} \mathbf{x}_{-j} = \mathbf{1}$$

134 The constants are ignored since, only relative values of the solution are of interest. Thus,
 135 finding a solution to the relaxed seeded isoperimetric partitioning problem is reduced to solving
 136 (10)².

137 **Remark:** An important property of the solution to seeded isoperimetric partitioning
 138 problem is a *continuity* property (discussed in detail in [21]). It states that, for any vertex
 139 v , there exists a path to the seed g (say) - $\langle v = v_0, v_1, \dots, g \rangle$, such that the solution x
 140 satisfies

$$141 \quad (11) \quad x(v_i) \geq x(v_{i+1})$$

142 i.e. there exists a descending path from the vertex v to the seed ($x(v)$ denotes the value of
 143 the solution at the vertex v). This property implies that the optimal component containing
 144 the seed is connected, which is important for practical purposes.

145 **3. Calculating the limit of minimizers.** In this section, we are going to compute the limit
 146 of the minimizers of (9). The maximum spanning tree is instrumental in doing so. Recall
 147 that a maximum spanning tree (MST) of a graph $G = (V, E, w)$ is a connected subgraph of
 148 G spanning V , with no cycles such that

$$149 \quad (12) \quad \text{weight of the MST} = \sum_{e_{ij} \in \text{MST}} w_{ij}$$

150 is maximized. The UMST is the weighted graph induced by the union of all the maximum
 151 spanning trees. In [19] the authors claim that instead of solving (10) on the Laplacian of the
 152 original graph, it is sufficient to solve the problem using the Laplacian of a MST of the graph.
 153 As a MST does not have any cycles, this allows for obtaining fast solution to (10). This was
 154 verified empirically in [19] but a detailed analysis of that claim is currently missing. We are
 155 going to undertake such an analysis using the Power Watershed framework.

156 Given $G = (V, E, w)$, a finite edge weighted graph, define an *exponentiated graph* by
 157 $G^{(p)} = (V, E, w^{(p)})$, where $w^{(p)}(e_{ij}) = (w(e_{ij}))^p$. In the rest of the article we assume that G
 158 has k distinct weights $w_1 < w_2 < \dots < w_k$. Also assume that G is connected. G^{umst} denotes
 159 the graph (weighted) induced by the UMST.

160 **3.1. Power Watershed Framework.** Let $\{Q_i(\cdot)\}$ be a set of cost functions on \mathbb{R}^n and
 161 $0 < \lambda_1 < \lambda_2 < \dots < \lambda_k$ a set of constants. Define

$$162 \quad (13) \quad Q(\mathbf{x}) = \sum_{i=1}^k \lambda_i Q_i(\mathbf{x})$$

163 and

$$164 \quad (14) \quad Q^{(p)}(\mathbf{x}) = \sum_{i=1}^k \lambda_i^p Q_i(\mathbf{x})$$

² Note that for the unseeded version, the equivalent of (10) is $L\mathbf{x} = \mathbf{1}$. In [20] it was stated that the reason
 to consider the seeded problem is - $L\mathbf{x} = \mathbf{1}$ has several solutions. However, multiplying with $\mathbf{1}^t$ on both sides
 of $L\mathbf{x} = \mathbf{1}$, one can easily see that the LHS is equal to 0, while the RHS is greater than 0. This implies that the
 system of equations $L\mathbf{x} = \mathbf{1}$ has no solutions. Here we provide a better understanding for solving the relaxed
 seeded isoperimetric partitioning problem.

165 Let \mathbf{x}_p^* denote a minimizer of $Q^{(p)}$. We are interested in computing a limit \mathbf{x}^* of minimizers
 166 $(\mathbf{x}_p^*)_{p>0}$ as $p \rightarrow \infty$.

167 In [27], the author provides a theory to calculate a limit of minimizers. In simple terms,
 168 given a specific structure of the cost function, one can compute a limit of minimizers by
 169 iteratively calculating the set of minimizers at every scale starting from the highest scale λ_k
 170 (λ_i provides the notion of scale here). Formally, the following result holds.

171 **Theorem 3.1.** [27] Let $Q^{(p)} := \sum_{i=1}^k \lambda_i^p Q_i$, where $(\lambda_i)_{1 \leq i \leq k} \in \mathbb{R}^k$ is such that $0 < \lambda_1 <$
 172 $\lambda_2 < \dots < \lambda_k \leq 1$, and $(Q_i)_{1 \leq i \leq k}$ are real-valued continuous functions defined on \mathbb{R}^n . Let M_k
 173 be the set of minimizers of Q_k , and for $1 \leq i < k$, M_i be recursively defined as follows:

$$174 \quad (15) \quad M_k = \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^n} Q_k(\mathbf{x})$$

$$175 \quad (16) \quad \forall 1 \leq i < k, M_i = \operatorname{argmin}_{\mathbf{x} \in M_{i+1}} Q_i(\mathbf{x})$$

176 Any convergent sequence $(\mathbf{x}_p)_{p>0}$ of minimizers of $Q^{(p)}$ converges to some point of M_1 . In
 177 particular, if for all $p > 0$, $(\mathbf{x}_p)_{p>0}$ is bounded (i.e. if there exists $C > 0$ such that for all $p > 0$,
 178 $\|\mathbf{x}_p\|_\infty \leq C$), then, up to a subsequence $(\mathbf{x}_p)_{p>0}$ is convergent to a point in M_1 . Further, we
 179 can estimate the minimum of $Q^{(p)}$ as follows:

$$180 \quad (17) \quad \min_{\mathbf{x} \in \mathbb{R}^n} Q^{(p)}(\mathbf{x}) = \sum_{1 \leq i \leq k} \lambda_i^p m_i + o(\lambda_1^p)$$

181 where $m_i = \min_{\mathbf{x} \in M_i} Q_i(\mathbf{x})$ and $o(\lambda_1^p)$ is the Landau notion of negligibility.

182 The following algorithm is readily derived from the theorem.

Algorithm 1 Calculating limit of minimizers [27]

Set $i = k$ and M_{i+1} is the entire space.

while $i > 0$ **do**

 Compute the set of minimizers $M_i = \operatorname{argmin}_{\mathbf{x} \in M_{i+1}} Q_i(\mathbf{x})$

end while

return Some $\mathbf{x} \in M_1$.

183 **3.2. Limit of Minimizers of Relaxed Seeded Isoperimetric Partitioning Problem.** As we
 184 are working with finite graphs, each of the edge weights can take values in a finite set. Hence,
 185 there are k distinct weights $w_1 < w_2 < w_3 < \dots < w_k$, the cost function of the isoperimetric
 186 partitioning problem can be written as

$$187 \quad (18) \quad Q(\mathbf{x}) = \mathbf{x}^t L \mathbf{x} = \sum_{i=1}^k w_i (\mathbf{x}^t L_i \mathbf{x})$$

188 where L_i denotes the Laplacian of the graph induced by edges with weight exactly equal to
 189 w_i . Observe that non-diagonal entries of L_i are either 0 or 1. It is easy to see that the cost
 190 function of the isoperimetric partitioning problem for the exponentiated graphs is equal to

$$191 \quad (19) \quad Q^{(p)}(\mathbf{x}) = \mathbf{x}^t L \mathbf{x} = \sum_{i=1}^k w_i^p (\mathbf{x}^t L_i \mathbf{x})$$

192 This allows us to use theorem 3.1 to calculate the limit of minimizers as $p \rightarrow \infty$. The
 193 following theorem holds.

194 **Theorem 3.2.** *Let G denote a finite edge-weighted graph and G^{umst} the weighted graph
 195 induced by the UMST. The limit of minimizers of the relaxed seeded isoperimetric partition-
 196 ing problem on $G^{(p)}$ is equal to the limit of minimizers of the relaxed seeded isoperimetric
 197 partitioning problem on $G^{(p),umst}$ as $p \rightarrow \infty$.*

198 The above theorem provides an initial step to explain the MST approximation in [19]. It
 199 has been shown earlier [7, 15, 11, 6, 14] that the limit of minimizers preserves the essential
 200 properties of solutions, thus giving useful results. Theorem 3.2 states that computing the
 201 limit of the minimizers of the relaxed seeded isoperimetric partitioning problem on UMST
 202 is same as on the original graph. Thus, assuming that the limit of minimizers yields useful
 203 solutions, theorem 3.2 allows us to solve the relaxed seeded isoperimetric partitioning problem
 204 on a smaller³ UMST graph instead of the original graph. While currently there is no formal
 205 statement justifying such approximation, section 5 provides some empirical evidences for this
 206 claim.

207 Note that algorithm 1 provides only a heuristic to calculate the limit of minimizers, which
 208 may not be implementable in practice. Theorem 3.3 provides a method of calculating the
 209 limit of minimizers of the relaxed seeded isoperimetric partitioning problem.

210 **Theorem 3.3.** *Let \mathbf{x}^* be a limit of minimizers of the relaxed seeded isoperimetric partition-
 211 ing problem on $G^{(p)}$ as $p \rightarrow \infty$. Assuming that x_j denotes the seed, $L_{(-j,-j)}^{umst}$ denotes the
 212 Laplacian of the UMST with j^{th} row and column removed. Then for some $\lambda \in \mathbb{R}$,*

$$213 \quad (20) \quad L_{(-j,-j)}^{umst} \mathbf{x}^* = \lambda \mathbf{1}$$

214 Since $L_{(-j,-j)}^{umst}$ is non-singular, solving the equation

$$215 \quad (21) \quad L_{(-j,-j)}^{umst} \mathbf{x} = \mathbf{1}$$

216 yields a solution to the relaxed seeded isoperimetric partitioning problem.

217 **3.3. Going from UMST to MST?.** The above theorems exhibit that in the limiting case,
 218 solving the relaxed seeded isoperimetric partitioning problem on the UMST is same as solving
 219 the problem on the original graph. However, in [19] the authors consider an arbitrary MST
 220 to solve the problem.

221 In general one cannot assure that solving the relaxed seeded isoperimetric partitioning
 222 problem on UMST and MST provide the same solution. Examples demonstrating this are
 223 discussed in the next section. However, there are few cases (not encountered often in practice)
 224 where it holds true.

- 225 • When all edges have distinct weights, UMST and MST are identical and hence they
 226 yield the same solution.
- 227 • Note that the final partition is obtained by thresholding the solution to the seeded
 228 isoperimetric partitioning problem. In the case when the ideal partition exists as a

³smaller in the sense of number of edges

229 threshold of the original graph, it is assured that both UMST and MST give the correct
 230 the partition⁴. This is because all the three structures MST, UMST and the original
 231 graph have the same components when thresholded.

232 Intuitively, UMST removes the ambiguity of choosing an arbitrary MST and considers a
 233 ‘union’ instead. And hence, it results in a more deterministic behavior. In the following part
 234 we answer the question - How different can the solutions of UMST and MST be?

235 Before proceeding, we need some more notions. Recall that it is assumed that there exists
 236 k distinct weights on the graph - $w_1 < w_2 < \dots < w_k$. For any subgraph one can assign a *weight-*
 237 *distribution* vector of length k - $[l_1, l_2, \dots, l_k]$, where l_i is an integer denoting the number of edges
 238 with weight w_i . Thus, the UMST and MST graphs also have such weight distributions which
 239 is denoted by $[u_1, u_2, \dots, u_k]$ and $[m_1, m_2, \dots, m_k]$ respectively. The following proposition
 240 holds.

241 **Proposition 3.4.** *Given an edge weighted graph $G = (V, E, w)$, all the MST’s have the same*
 242 *weight distribution.*

243 Recall that the solution is obtained by solving the following linear equation

$$244 \quad (22) \quad L_{(-j, -j)} \mathbf{x}_{-j} = \mathbf{1}$$

245 where $L_{(-j, -j)}$ is the reduced Laplacian. This implies that for each $i \neq j$ ($x_j = 0$ corresponds
 246 to the seed) the following equation holds.

$$247 \quad (23) \quad x_i = \sum_l \frac{w_{il}}{d_i} x_l + \frac{1}{d_i}$$

248 where $d_i = \sum_l w_{il}$ denotes the degree of the vertex i . Let D denote the matrix $\text{diag}(1/d_1, 1/d_2,$
 249 $\dots, 1/d_n)$, and assume W to indicate the adjacency matrix, hence W_{il} denotes the weight w_{il} .
 250 Also let f indicate the vector $[1/d_1, 1/d_2, \dots, 1/d_n]$. Using these notations, the solution to
 251 the relaxed seeded isoperimetric partitioning problem satisfies

$$252 \quad (24) \quad T(\mathbf{x}) = D^{-1}W \mathbf{x} + f = \mathbf{x}$$

253 In other words, the solution is a fixed point of the linear operator $T(\cdot)$.

254 Observe that, each adjacency matrix gives a different operator (the matrix D depends on
 255 the adjacency matrix). Thus, there are two operators - T_{umst} and T_{mst} , corresponding to the
 256 UMST and MST graphs respectively. To characterize the difference between the solutions of
 257 seeded isoperimetric partitioning problem on UMST and MST, it is enough to consider the
 258 distance between these two operators. In particular, the following theorem holds.

259 **Theorem 3.5.** *Let T_{umst} and T_{mst} denote the operators on UMST and MST respectively,*
 260 *as defined above. Then there exists two positive constants K_1 and K_2 such that*

$$261 \quad (25) \quad K_1 \sum_{i=1}^k (u_i - m_i)^2 w_i^2 \leq \|T_{umst} - T_{mst}\| \leq K_2 \sum_{i=1}^k (u_i - m_i)^2 w_i^2$$

⁴Assuming that the ideal partition is the one which minimizes the isoperimetric ratio.

262 The significance of theorem 3.5 is - it gives bounds on how different the solutions of UMST
 263 and MST can be for the seeded isoperimetric partitioning problem in terms of their weight
 264 distributions. As a consequence of proposition 3.4, these bounds can be calculated from the
 265 raw data without resorting to solving the linear equation or using an explicit structure of a
 266 MST. Note that in the case of all edge weights being distinct, the following holds true

$$267 \quad (26) \quad \sum_{i=1}^k (u_i - m_i)^2 w_i^2 = 0$$

268 since $u_i = m_i$ for all i . This implies that the bounds in theorem 3.5 are attained.

269 **4. Limit of Minimizers of Discrete Isoperimetric Partitioning Problem.** Given the in-
 270 terest of PW framework in solving the relaxed seeded isoperimetric partitioning problem, we
 271 explore the limit of the discrete isoperimetric partitioning problem (4) in the PW framework.
 272 In this section we characterize the limit of minimizers to the discrete problem (4) in the Power
 273 Watershed framework and establish links with other existing methods.

274 Theorem 4.1 shows the most important property of the limit of minimizers of the discrete
 275 isoperimetric partitioning problem. Recall the assumption that the edge weights can attain
 276 one of the k distinct weights $w_1 < w_2 < w_3 < \dots < w_k$. Also let $G_{\geq w}$ indicate the graph
 277 induced by the edges in G whose weight is at least w .

278 **Theorem 4.1.** *Let x^* be a limit of minimizers of the discrete isoperimetric partitioning
 279 problem. If $G_{\geq w}$ is disconnected, i.e. it has at least two connected components, then x^* is
 280 constant on each of these components.*

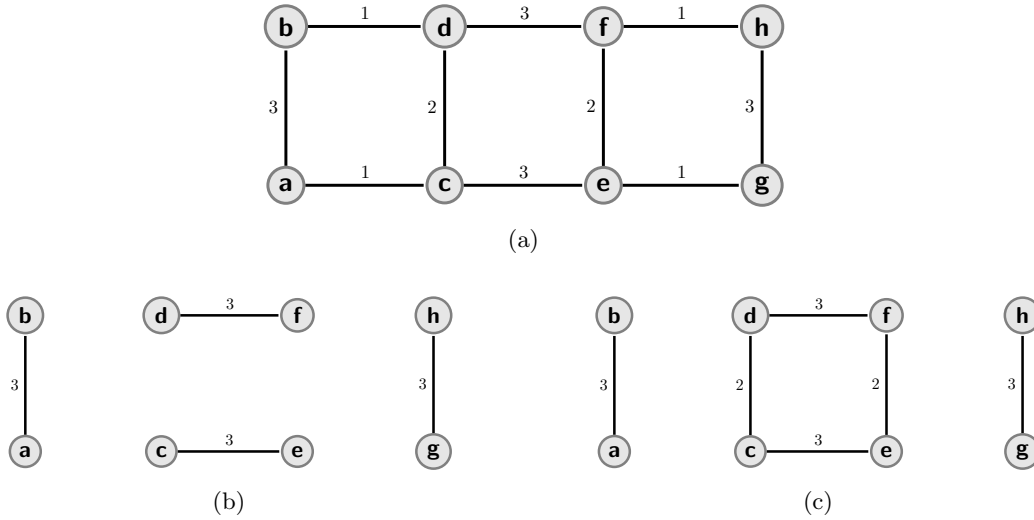


Figure 1. (a) A synthetic graph, G (b) Graph in (a) thresholded at 3, $G_{\geq 3}$. (c) Graph in (a) thresholded at 2, $G_{\geq 2}$. Note that the graph thresholded at 1 is the original graph.

281 To illustrate theorem 4.1, consider the graph G as shown in figure 1. The theorem implies
 282 that x^* (solution to limit of minimizers of the discrete isoperimetric partitioning problem)

283 has constant values within each of the connected components of $G_{\geq 3}$ i.e. $x^*(a) = x^*(b)$,
 284 $x^*(c) = x^*(e)$, $x^*(d) = x^*(f)$, $x^*(h) = x^*(g)$. Further, theorem 4.1 applied to $G_{\geq 2}$ implies
 285 that the equalities $x^*(c) = x^*(d) = x^*(e) = x^*(f)$ hold.

286 The above illustration indicates that one can define a *critical weight*, which is the largest
 287 weight w such that $G_{>w}$ (the subgraph induced by edges of G with weights greater than w) is
 288 disconnected while $G_{\geq w}$ is connected. Theorem 4.1 implies that, x^* attains a constant value
 289 on each component of $G_{>w}$.

290 Recall that a subgraph induced by a subset of edges $E_1 \subset E$ of an edge-weighted graph
 291 $G = (V, E, w)$ is said to be a *maximum* if: every edge in E_1 has the same weight; any edge
 292 in $E \setminus E_1$ adjacent to an edge in E_1 has strictly lesser weight; and E_1 induces a connected
 293 subgraph. The *watershed cut* [12, 13] of an edge-weighted graph is a maximum spanning forest
 294 relative to its maxima (when the edge weights represent similarity measure). This allows us
 295 to make the following interesting observation:

296 *The partition corresponding to the PW limit of minimizers of the discrete*
 297 *isoperimetric partitioning problem can be obtained by successively adding edges*
 298 *to a watershed cut in decreasing order of their weights until the resulting graph*
 299 *contains two connected components.*

300 In the case where $G_{>w}$ has exactly two components, the limit of the discrete isoperimetric
 301 partitioning problem is exactly the same as a watershed cut. For further details on watershed
 302 cuts, the reader may refer to [12, 13].

303 **Remark:** Note that the discrete isoperimetric partitioning problem is NP-hard. This
 304 property holds when computing the limit of minimizers as well, i.e. there does not exist a
 305 polynomial time algorithm to calculate the limit of minimizers to the discrete isoperimetric
 306 partitioning problem in general.

307 **5. Empirical Analysis.** To recap, we have shown in the Power Watershed framework that
 308 the limits of the relaxed seeded isoperimetric partitioning problem on the original graph and
 309 UMST are identical. Also, we have analyzed the situation if MST was used in place of
 310 UMST to solve the relaxed seeded isoperimetric partitioning problem as proposed in [19]. In
 311 this section we provide several examples and experimental results to further understand the
 312 relation between the solutions of the relaxed seeded isoperimetric partitioning problem on the
 313 original graph, UMST and MST.

314 **Remark:** Note from earlier that only the relative values of the solution are of interest,
 315 since after the calculation of the solution, each threshold is evaluated to obtain the optimal
 316 partition (See [20] for details). Thus, in this section two solutions \mathbf{x} and \mathbf{y} are considered to
 317 be *equivalent* if they have the same order, that is

$$318 \quad (27) \quad x_i \leq x_j \Leftrightarrow y_i \leq y_j \text{ for all } i, j$$

319 A sufficient condition to achieve the same partition is provided by this equivalence condi-
 320 tion, and the thresholding step is not considered hereafter.

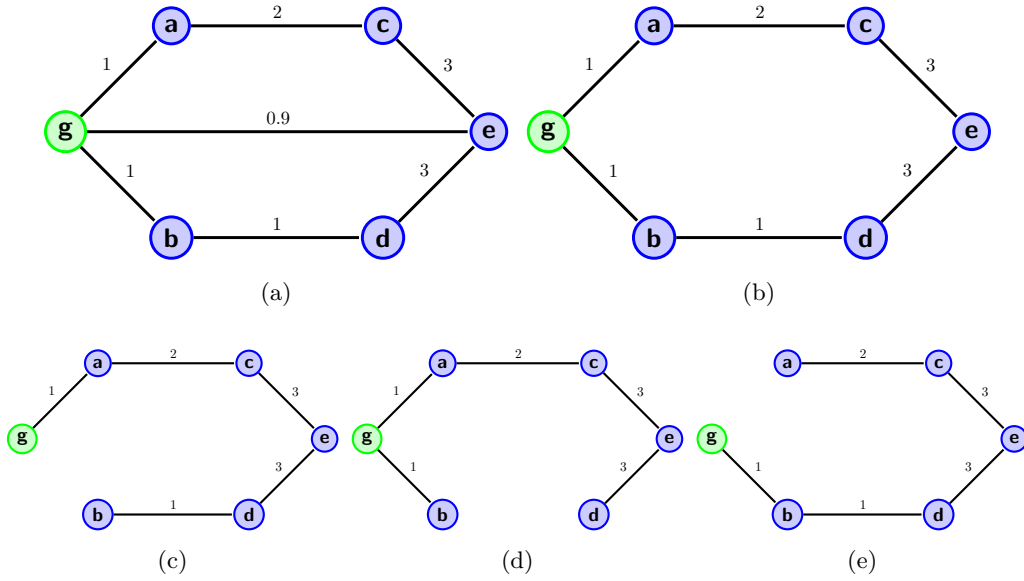


Figure 2. Example to illustrate the differences between the solutions to the relaxed seeded isoperimetric partitioning problem on the original graph, UMST and MST. (a) Original graph. (b) UMST. (c) MST_1 (d) MST_2 (e) MST_3 . (c) - (e) shows all possible MST's for graph in (a). vertex labelled 'g' indicates the seed vertex in all cases. The edge weights are as shown on the edges.

Table 1

Solutions to relaxed seeded isoperimetric partitioning problem with graphs in figure 2

Node	Original	UMST	MST_1	MST_2	MST_3
<i>g</i>	0.00	0.00	0.00	0.00	0.00
<i>a</i>	1.69	2.68	5.00	4.00	11.16
<i>b</i>	1.54	2.32	9.66	1.00	5.00
<i>c</i>	2.04	3.52	7.00	5.50	10.66
<i>d</i>	2.09	3.64	8.66	6.50	9.00
<i>e</i>	1.94	3.74	8.00	6.16	10.00

321 **5.1. Finding a suitable MST?** One question which naturally arises is - Does there exist
 322 a MST on which the solution of the relaxed seeded isoperimetric partitioning problem is
 323 equivalent to the solution obtained with the original graph? What about UMST?

324 In general it is not assured that such a MST exists. Consider a simple graph as shown in
 325 figure 2. Corresponding UMST and all possible MST's are also shown. Vertex 'g' denotes the
 326 seed. The solution to the relaxed seeded isoperimetric partitioning problem for each of these
 327 graphs is given in table 1.

328 The following conclusions can be drawn from the results:

- 329 1. Note that the relative ordering of the co-ordinates of solution to the seeded isoperimet-
 330 ric partitioning problem on original graph does not match with any of MST's. Hence
 331 this provides a counter example.
- 332 2. The relative ordering of the co-ordinates of the solutions for UMST is different from

333 those obtained from any of the MST's.

334 3. Moreover the relative ordering of the co-ordinates of solutions of UMST and the orig-
335 inal graph also do not match.

336 Example in figure 2 conclusively shows that, in general one cannot expect a relation
337 between the solutions of the relaxed seeded isoperimetric partitioning problem on the original
338 graph, UMST and MST. However, in several practical cases, one can expect them to be 'close'.
339 One such application is that of image segmentation. In this case, most of the edges in the
340 UMST and MST are within the object and hence might give similar results. This is discussed
341 in detail in the next part of the section.

342 Another important observation from the above example is that the values of solutions on
343 MST are widely fluctuating. That is, the solution changes with respect to the choice of MST.
344 This ambiguity is not present when considering the UMST.

345 **5.2. Results in Practice.** In this part we focus on how the solutions of the relaxed seeded
346 isoperimetric partitioning problem on the original graph, UMST and MST behave in practice.
347 Let \mathbf{x} , \mathbf{x}^{umst} , \mathbf{x}^{mst} indicate the solution to the original graph, the UMST solution and MST
348 solution respectively. As the seed, a vertex in the interior of an object is randomly picked,
349 and the same seed is used for all three solutions. The datasets considered are the Weizmann
350 1-Object and 2-Object datasets [2] and BSDS500 dataset [3]. We select to use the classic
351 4-adjacency graph constructed from the image⁵.

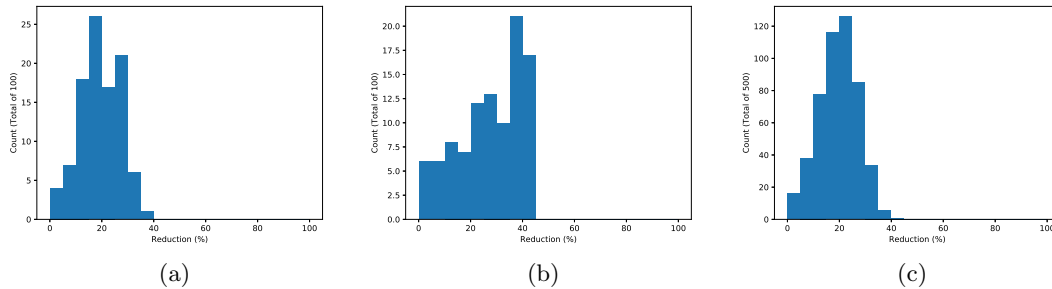


Figure 3. Histograms indicating the amount of reduction in number of edges obtained when constructing the UMST. x -axis represents the percentage reduction obtained. y -axis represents the number of images achieving the given amount of reduction. The results are computed on (a) Weizmann 1-Object dataset, (b) Weizmann 2-Object dataset and (c) BSDS500 dataset.

352 **Implementation Note:** Recall the assumption that there exists k distinct values for the
353 edge weights. In practice, the edge weights are represented by floating point numbers and
354 hence 'equality' cannot be judged. To overcome this, we consider an ϵ -precision where the
355 weights, w_{ij} are modified as below.

$$(28) \quad w_{ij} \rightarrow \text{int}(w_{ij}/\epsilon) \times \epsilon$$

⁵The isoperimetric graph partitioning problem is no longer NP-hard on a 4-adjacency graph. However, the number of partitions of the vertex set V into two subsets is $\mathcal{O}(|V|^{\frac{3}{2}})$ [1]. Hence, solving the discrete isoperimetric graph partitioning directly is inefficient.

357 Intuitively, this operation restricts the precision of a floating point number.

358 Firstly, observe that the reduction in the complexity is by reducing the number of non-zero
 359 entries of the matrix L in (10). Thus one question to ask is - How much reduction in the
 360 number of edges is achieved when considering the reduction to UMST or MST? In the case
 361 of using a MST instead of the original graph, the number of edges is simply $n - 1$ where n
 362 indicates the number of vertices in the graph (which is $\approx 50\%$ reduction on a 4-adjacency
 363 graph). In the case of UMST, in general it is not possible to predict the amount of reduction
 364 in number of edges. Figure 3 shows the histogram of the percentage of reduction achieved
 365 on the Weizmann and BSDS datasets. Observe that, on average we achieve 20% reduction,
 366 which can go up to 40%. Intuitively, UMST only removes ‘non-informative’ edges from the
 367 graph. This is dependent on the image under consideration.

368 **5.2.1. Accuracy of \mathbf{x} , \mathbf{x}^{umst} , \mathbf{x}^{mst} .** We now inspect how the different solutions affect the
 369 accuracy of segmentation. For the results to be as precise as possible we consider *recursive*
 370 *partitioning* - that is, each of the components of the partition is further partitioned, until a
 371 stopping criterion is met. We consider the stopping criterion to be when isoperimetric ratio
 372 crosses a given threshold.

Table 2

Accuracy Measures used in figure 4

Measure	Description
Adjusted Rand Index (ARI) [23]	Rand Index adjusted for chance.
Adjusted Mutual Information (AMI) [29]	Mutual Information adjusted for chance.
Precision (P_r in [28])	Reflects the probability that a pair of pixels predicted to have same label does indeed have same label.
Recall (R_r in [28])	Reflects the probability that a pair of pixels having the same label is predicted to have same label.
F-Score (F_r in [28])	Summary Measure given by (2 * Precision * Recall)/(Precision + Recall)

373 The accuracy measures considered are described in table 2. For each image in Weizmann
 374 1-Object dataset, we compare the recursive partition obtained using the solution to the relaxed
 375 seeded isoperimetric partitioning problem on UMST/MST with the solution to the relaxed
 376 seeded isoperimetric partitioning problem on original graph. These results are plotted as a
 377 scatter plot in figure 4. Note that the results on UMST and original graph are almost similar.
 378 However, when considering MST, sometimes the results are better and sometimes worse. This
 379 can, once again be attributed to the previous observation that MST loses information.

380 **Relative ordering of the co-ordinates in \mathbf{x} , \mathbf{x}^{umst} , \mathbf{x}^{mst} .** Recall that two solutions are
 381 considered equivalent if they have the same order (see (27)). Here we consider how different
 382 are the orders of \mathbf{x}^{umst} , \mathbf{x}^{mst} with respect to \mathbf{x} . In figures 5a, 5b the scatter plot is used to
 383 demonstrate the differences between \mathbf{x}^{umst} and \mathbf{x}^{mst} . The scatter plot is between the values of
 384 the solutions \mathbf{x}^{umst} and \mathbf{x}^{mst} , with respect to \mathbf{x} , at several random vertices across few random
 385 images. In the ideal case of the order being perfectly preserved, we expect the plot to follow
 386 a strictly increasing function. The size of deviation from the increasing function reflects how
 387 different the orders of the solution are. In figure 5a, observe that the UMST preserves the
 388 order quite well, while figure 5b suggests that MST does not preserve the order so well. This

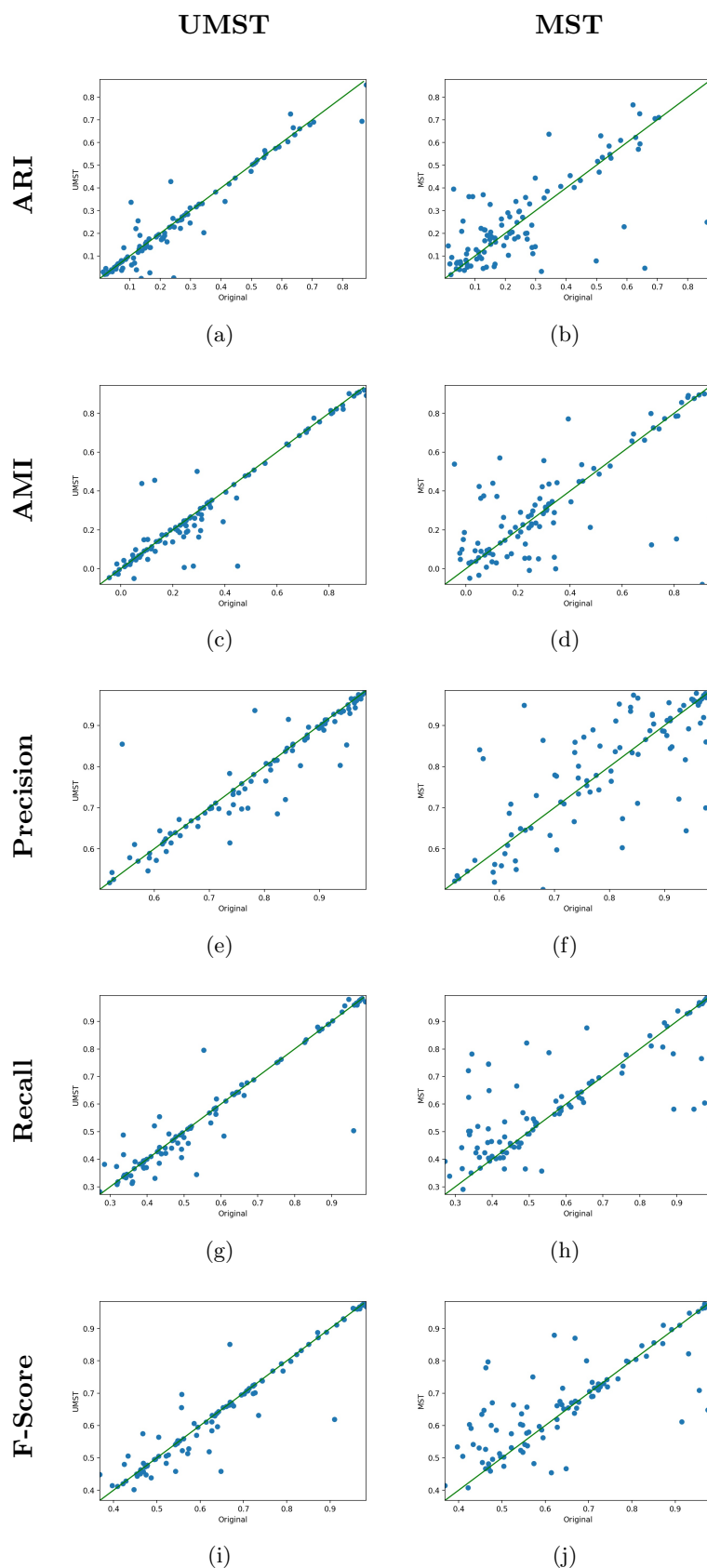


Figure 4. Scatter plots between measures obtained using UMST/MST and the original graph, on images from Weizmann 1-Object dataset, for several different measures described in table 2. Observe that the results for UMST (first column) are very close to the original graph, while results obtained using MST (second column) have large deviations. This is especially evident when considering the measure ‘Precision’.

This manuscript is for review purposes only.

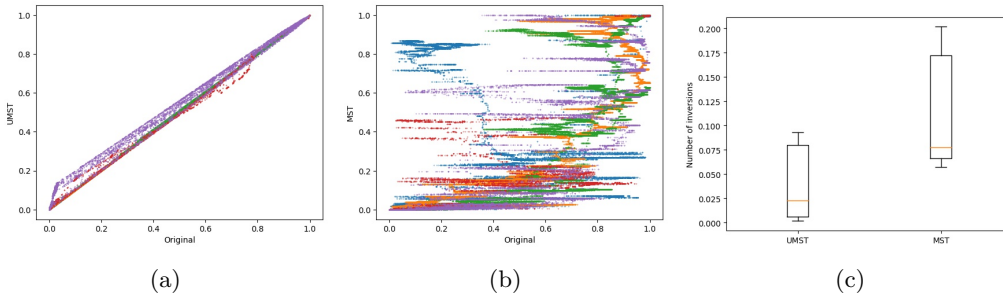


Figure 5. Relative ordering of the solutions of seeded isoperimetric partitioning problem on the UMST and MST with respect to the original graph. (a) Shows the scatter plot of the solutions at the vertices between \mathbf{x}^{umst} and \mathbf{x} for several images. Observe that the ordering relation (27) is preserved well. (b) Shows the scatter plot of the solutions at the vertices between \mathbf{x}^{mst} and \mathbf{x} for several images. Observe that the ordering relation (27) is **not** preserved as well as UMST. (c) Box plot indicating the number of inversions obtained, normalized with number of pairs, for \mathbf{x}^{umst} and \mathbf{x}^{mst} with respect to \mathbf{x} .

389 indicates that MST loses much more relevant information with respect to the original graph
390 than UMST.

391 Another metric to measure the amount of difference is by calculating the *number of in-*
392 *versions* between the solutions. This is calculated as follows - Order the solution \mathbf{x}^{umst} or
393 \mathbf{x}^{mst} with respect to the order \mathbf{x} . Then count the number of inverted pairs - (x_i, x_j) such that
394 $i < j$ and $x_i > x_j$. Normalize with respect to the total number of pairs possible, to obtain
395 consistency across differently sized images. This is measured on each image of Weizmann
396 1-Object dataset and a box plot is plotted in figure 5c. This substantiates the evidence that
397 MST loses a lot of relevant information while UMST preserves it.

398 **6. Conclusions and Perspectives.** In this article we have revisited the NP-hard isoperi-
399 metric graph partitioning problem. We have presented a detailed analysis of the continuous
400 relaxation of the problem, clarifying the construction followed in [20, 21]. In [19] the author
401 exhibited empirically that - solving the relaxed seeded isoperimetric partitioning problem on a
402 much smaller graph (MST) yields a good approximation to the solution on the original graph.
403 We provided an alternative explanation for this approximation by considering the limit of
404 minimizers in the Power Watershed framework. We have shown that, in the limiting case,
405 solving the problem on UMST is equivalent to solving the problem on the original graph.
406 We have established bounds on the difference between the solutions of the relaxed seeded
407 isoperimetric partitioning problem on UMST and MST graphs. Empirical experiments were
408 conducted to analyse these techniques in practice.

409 It is also possible to characterize the limit of minimizers of the solutions of the discrete
410 isoperimetric partitioning problem in the Power Watershed framework. Although, the compu-
411 tation of the exact limit still remains NP-hard, we have shown that these solutions are ‘close’
412 to watershed cuts. Further analysis of the limit of minimizers to the discrete isoperimetric
413 partitioning problem is a subject of future research. We mention here two possible directions:
414 (1) MST has been proved a good heuristic in solving the NP-hard travelling salesman problem
415 [17, 16]. Would it be possible to go along the same lines, and prove some theoretical bounds

416 on the solutions to the isoperimetric graph partitioning problem? (2) In the theory of scale-set
 417 analysis [22], it is shown that the celebrated Mumford-Shah functional [26] can be solved in
 418 linear time on a tree of segmentations, and that the persistence of regions is a good indicator
 419 of their relevance with respect to image segmentation. We can envision using the Cheeger
 420 constant in a similar way to what is done in [22]. Extensions to such ideas have been proposed
 421 in the shaping framework [31], and can be adapted to the case of the Cheeger constant. Would
 422 that be possible to estimate bounds on the solutions in such frameworks?

423 Beyond segmentation, filtering images is another possible direction of research. As the
 424 Cheeger cut problem is closely related to total variation (TV) minimization, it would be inter-
 425 esting to explore the utility of PW framework for solving TV minimization problems. More
 426 generally, going beyond images, is to explore the application of the UMST-based algorithm
 427 as a fast clustering technique for data analysis.

428 As a final note, as demonstrated in this paper and others [15, 7, 11, 14, 6], Power Watershed
 429 framework has proved to be very useful. From a theoretical standpoint, understanding the
 430 working principle behind the Power Watershed framework is still an open problem.

431 **7. Acknowledgments.** The authors would like to thank Leo Grady for a fruitful discussion
 432 that improved the manuscript. The first author would like to thank Vance Faber for providing
 433 some useful references. The second author would like to thank CEFIPRA for providing Raman
 434 Charpak Fellowship to support the research. This work is partially funded by CEFIPRA,
 435 EMR/2015/000853 SERB, ISRO/SSPO/Ch-1/2016-17 ISRO, ANR-15-CE40-0006 CoMeDiC
 436 and ANR-14-CE27-0001 GRAPHSIP research grants.

437 **Appendix A. Proof of proposition 2.2 .**

438 *Proof.* Given a subset S , define $\mathbf{x}(S) = (x_1, x_2, \dots, x_n)$ by

$$439 \quad (29) \quad x_i = \begin{cases} 1/2 - \delta & \text{if } i \in S \\ 1/2 + \delta & \text{otherwise} \end{cases}$$

440 Then,

$$441 \quad (30) \quad \mathbf{x}^t L \mathbf{x} = \sum_{ij} w_{ij} (x_i - x_j)^2 = \sum_{e_{ij} \in \partial S} w_{ij} (2 * \delta)^2$$

442 Without loss of generality, assume that $|S| \leq (1/2)|V|$ (Otherwise, take the complement).
 443 Then the denominator is equal to

$$444 \quad (31) \quad \frac{|V|}{2} - \delta|S| + \delta|V \setminus S|$$

445 Thus the cost of this vector is given by

$$446 \quad (32) \quad \frac{\sum_{e_{ij} \in \partial S} w_{ij} (2 * \delta)^2}{\frac{|V|}{2} - \delta|S| + \delta|V \setminus S|}$$

447 Note that the above cost converges to 0 as $\delta \rightarrow 0$. Hence one can find a $\delta > 0$ such that the
 448 cost is less than ϵ . ■

449 **Appendix B. Proof of theorem 3.2.** Let $G = (V, E, w)$ denote the graph, and G^{umst}
 450 denote the weighted graph induced by the UMST. Let L and L^{umst} denote the Laplacian of
 451 the graphs G and G^{umst} respectively. Denote $L^{other} = L - L^{umst}$ to indicate the Laplacian
 452 of the graph induced by the edges which do not belong to the UMST. Let L_i denote the
 453 Laplacian of the graph induced by edges with weight equal to w_i . Recall that- it is assumed
 454 there are k distinct weights $w_1 < w_2 < w_3 < \dots < w_k$. This notation is also compounded,
 455 in the sense that- L_k^{umst} indicates the Laplacian of the subgraph of UMST graph induced by
 456 edges with weight equal to w_k .

457 The idea of the proof is to show that at every level of the algorithm 1, the minimizers
 458 obtained with respect to the graph also minimize the corresponding optimization problem
 459 with respect to the UMST graph. This implies that the output of the algorithm is same for
 460 both the graph and its corresponding UMST, which proves the theorem.

461 Following the steps algorithm 1, starting at the highest level k and filter the set of mini-
 462 mizers to obtain the limit of minimizers.

463 **At level k** solve the following optimization problem.

$$\begin{aligned}
 & \arg \min_x \quad \mathbf{x}_{-j}^t L_{k,(-j,-j)} \mathbf{x}_{-j} \\
 & \text{subject to} \quad (\mathbf{x}_{-j})_i \in [0, 1] \forall i \\
 & \quad \quad \quad \mathbf{x}_{-j}^t \mathbf{1} = \mu
 \end{aligned}
 \tag{33}$$

465 The above optimization problem is stated for the graph. The equivalent optimization problem
 466 for the UMST is obtained by replacing the Laplacian of the graph with the Laplacian of the
 467 UMST. Note that it is assumed the seed is placed at some arbitrary vertex x_j . From above,
 468 the set of solutions to this optimization problem is the set of solutions obtained by solving

$$L_{k,(-j,-j)} \mathbf{x}_{-j} = \lambda_k \mathbf{1}
 \tag{34}$$

470 **Lemma B.1.** *Given a graph G and the corresponding UMST graph G^{umst} , if w_k denotes*
 471 *the highest weight, then*

$$L_{k,(-j,-j)} = L_{k,(-j,-j)}^{umst}
 \tag{35}$$

473 The lemma B.1 essentially tells that at the highest level, the Laplacian for the graph and
 474 the corresponding UMST are the same. And hence, the set of minimizers obtained after level
 475 k is same for both the graph and its UMST.

476 Let A_k indicate the matrix obtained by stacking the indicator vectors of the components
 477 of $G_{\geq w_k}$. Clearly, the following relation holds

$$L_{k,(-j,-j)} A_k = \mathbf{0}
 \tag{36}$$

479 This follows from the properties of the Laplacian [10]. Hence the following lemma holds.

480 **Lemma B.2.** *Let M_k denote the set of minimizers obtained by solving (34). If \mathbf{x}_k^* denotes*
 481 *some solution to the equation, then the set M_k is characterized by*

$$\mathbf{x}_k^* + A_k \mathbf{y}
 \tag{37}$$

483 *for any y .*

484 **At level $k - 1$** , thanks to the above results, solve the optimization problem

$$\begin{aligned}
& \arg \min_{\mathbf{x}} \quad \mathbf{x}_{-j}^t L_{k-1,(-j,-j)} \mathbf{x}_{-j} \\
& \text{subject to} \quad (\mathbf{x}_{-j})_i \in [0, 1] \forall i \\
& \quad \quad \quad \mathbf{x}_{-j}^t \mathbf{1} = \mu \\
& \quad \quad \quad \mathbf{x}_{-j}^t \approx \mathbf{x}_k^* + A_k \mathbf{y}
\end{aligned}
\tag{38}$$

486 where \approx is to be read as - ‘is of the form’. Once again, note that the above optimization
487 problem is stated for the graph. To obtain the equivalent optimization problem for the UMST
488 graph one simply replaces the Laplacian of the graph with the Laplacian of the UMST graph.
489 Simplifying the above optimization problem,

$$\begin{aligned}
& \text{minimize}_{\mathbf{y}} \quad (\mathbf{x}_k^*)^t L_{k-1,(-i,-i)} \mathbf{x}_k^* + 2(\mathbf{x}_k^*)^t L_{k-1,(-i,-i)} A_k \mathbf{y} + \mathbf{y}^t A_k^t L_{k-1,(-i,-i)} A_k \mathbf{y} \\
& \text{subject to} \quad (\mathbf{x}_k^* + A_k \mathbf{y})^t \mathbf{1} = \mu
\end{aligned}
\tag{39}$$

491 The following lemma holds.

492 **Lemma B.3.** *Recall that $L_{k-1} = L_{k-1}^{other} + L_{k-1}^{umst}$. Hence, $L_{k-1}^{other} A_k = 0$.*

493 The proof of the above lemma is seen by noting that - ‘other’ edges apart from UMST are intra-
494 component edges in the components of $\mathcal{G}_{\geq w_k}$. So, components of the graph whose Laplacian
495 would be L_{k-1}^{other} are subsets of the components of $\mathcal{G}_{\geq w_k}$. Thus, from the properties of the
496 Laplacian, the above lemma holds true.

497 Thanks to the above lemma, the optimization problem reduces to

$$\begin{aligned}
& \text{minimize}_{\mathbf{y}} \quad (\mathbf{x}_k^*)^t L_{k-1,(-i,-i)}^{umst} \mathbf{x}_k^* + 2(\mathbf{x}_k^*)^t L_{k-1,(-i,-i)}^{umst} A_k \mathbf{y} + \mathbf{y}^t A_k^t L_{k-1,(-i,-i)}^{umst} A_k \mathbf{y} \\
& \text{subject to} \quad (\mathbf{x}_k^* + A_k \mathbf{y})^t \mathbf{1} = \mu
\end{aligned}
\tag{40}$$

499 Observe that the above optimization problem is nothing but the optimization problem
500 at level $k - 1$ of the UMST graph. Hence, at level $k - 1$, the solutions to the optimization
501 problem of the graph and the solutions to the optimization problem of the UMST graph are
502 equal.

503 Continuing this argument, one can easily see that the output of the algorithm 1 for both
504 the graph and its UMST is the same. Hence the theorem is proved.

505 **Appendix C. Proof of theorem 3.3.** Observe that from above, the limit of minimizers
506 can be written as

$$\mathbf{x}^* = \mathbf{x}_k^* + A_k \mathbf{x}_{k-1}^* + A_k A_{k-1} \mathbf{x}_{k-2}^* + \cdots + A_k A_{k-1} A_{k-2} \cdots \mathbf{x}_1^*
\tag{41}$$

508 Note that $L^{umst} = L_1^{umst} + L_2^{umst} + L_3^{umst} + \cdots + L_k^{umst}$. So,

$$L^{umst} \mathbf{x}^* = (L_k^{umst} + L_{k-1}^{umst} + \cdots + L_1^{umst})(\mathbf{x}_k^* + A_k \mathbf{x}_{k-1}^* + \cdots + A_k A_{k-1} \cdots \mathbf{x}_1^*)
\tag{42}$$

$$= L_k^{umst} \mathbf{x}_k^* + L_{k-1}^{umst} (\mathbf{x}_k^* + A_k \mathbf{x}_{k-1}^*) + \cdots
\tag{43}$$

511 The second step follows from noting that $L_k A_k = 0$, $L_{k-1} A_k A_{k-1} = 0$, and so on. This,
512 inturn, follows from the following lemma.

513 **Lemma C.1.** *The matrix $A_1 A_2$ consists of the indicator vectors of the connected compo-*
 514 *nents of $\mathcal{G}_{\leq w_2}$.*

515 Note that $L_k^{umst} \mathbf{x}_k^* = \lambda_k \mathbf{1}$. Now, the solutions at level 2, optimization problem in (40)
 516 satisfy

$$517 \quad (44) \quad A_k L_{k-1}^{umst} (\mathbf{x}_k^* + A_k \mathbf{y}) = A_k \lambda_{k-1} \mathbf{1}$$

518 **Lemma C.2.** *If given $AB \mathbf{x} = A \mathbf{y}$, then $B \mathbf{x} = \mathbf{y}$ when A is non singular (has an inverse)*
 519 *or if the intersection of column space of B and the null space of A is the zero vector.*

520 The proof of the above lemma is standard. From lemma C.2, it can be deduced that
 521 $L_{k-1}^{umst} (\mathbf{x}_k^* + A_k \mathbf{y}) = \lambda_{k-1} \mathbf{1}$. Thus, continuing from (43)

$$522 \quad (45) \quad L^{umst} \mathbf{x}^* = L_k^{umst} \mathbf{x}_k^* + L_{k-1}^{umst} (\mathbf{x}_k^* + A_1 \mathbf{x}_{k-1}^*) + \dots$$

$$523 \quad (46) \quad = \lambda_k \mathbf{1} + \lambda_{k-1} \mathbf{1} \dots$$

$$524 \quad (47) \quad = \Lambda \mathbf{1}$$

525 **Appendix D. Proof of Proposition 3.4.** For this proof, we need the cut-property of MST.
 526

527 **Lemma D.1 (Cut Property [4]).** *A spanning tree T of a connected graph G is a MST if and*
 528 *only if for every edge e in T , any edge e' in the cut of $T \setminus \{e\}$ satisfies $w(e') \geq w(e)$*

529 It is enough to show that one can transform an MST T to any other MST T' via a sequence
 530 of operations that keep the edge-weight distribution invariant.

531 Suppose T is different from T' then let $e \in T \setminus T'$. Now consider the cut edges of $T \setminus \{e\}$.
 532 It is evident from the cut property that $w(e) \geq w(f)$ for any cut-edge f of $T \setminus \{e\}$. Firstly,
 533 T' being a spanning tree has to contain a cut-edge of $T \setminus \{e\}$. Secondly total weight of T'
 534 and T are same as both are MST's and $e \notin T'$ implies existence of a cut-edge $e' \neq e$ of
 535 $T \setminus \{e\}$ with $w(e) = w(e')$ and $e' \in T'$. Now observe that $T \setminus \{e\} \cup \{e'\}$ is a MST with same
 536 edge-weight distribution as that of T . Since, we are working on finite graphs, after a finite
 537 sequence of steps, we would end up with T' starting from T . We remark that at every step,
 538 the edge-weight distribution of the MST remains invariant and hence the proof.

539 **Appendix E. Proof of theorem 3.5.** Recall that,

$$540 \quad (48) \quad T(\mathbf{x}) = D^{-1} W \mathbf{x} + f$$

541 Assuming that f in column space of $D^{-1} W$, which is the case if the graph is connected, one
 542 can rewrite the operator as,

$$543 \quad (49) \quad T(\mathbf{x}) = D^{-1} W (\mathbf{x} + \bar{f})$$

544 for some \bar{f} . Hence we have that

$$545 \quad \|T_{umst} - T_{mst}\| = \|(D^{-1} W)_{umst} - (D^{-1} W)_{mst}\|$$

$$546 \quad = \sum_{i,j} \left(\frac{w_{ij,umst}}{d_{i,umst}} - \frac{w_{ij,mst}}{d_{i,mst}} \right)^2$$

547 Let,

$$548 \quad K_1 = \frac{1}{(\max_i \{d_{i,umst}\})^2}$$

$$549 \quad K_2 = \frac{1}{(\min_i \{d_{i,mst}\})^2}$$

550 Note that,

$$551 \quad (50) \quad \sum_{i,j} \left(\frac{w_{ij,umst}}{d_{i,umst}} - \frac{w_{ij,mst}}{d_{i,mst}} \right)^2 \leq \sum_{i,j} \frac{1}{d_{i,umst}^2} (w_{ij,umst} - w_{ij,mst})^2$$

$$552 \quad (51) \quad \leq \frac{1}{(\min_i \{d_{i,umst}\})^2} \sum_{i,j} (w_{ij,umst} - w_{ij,mst})^2$$

$$553 \quad (52) \quad = \frac{1}{(\min_i \{d_{i,umst}\})^2} \sum_i (n_i - m_i)^2 w_i^2$$

$$554 \quad (53)$$

555 Similarly, one can obtain the other relation as well.

556 **Appendix F. Proof (sketch) of theorem 4.1.** Recall that the edge weights are assumed
557 to take one of k distinct values $w_1 < w_2 < \dots < w_k$.

558 The proof of this theorem follows the similar lines as that of theorem 3.2. That is, we
559 trace the working of the algorithm 1 to obtain the proof. The main difference is that, at level
560 m we solve the problem

$$561 \quad (54) \quad \begin{aligned} & \text{Find } \arg \min_{\mathbf{x}} \frac{\mathbf{x}^t L_m \mathbf{x}}{\min\{\mathbf{x}^t \mathbf{1}, (\mathbf{1} - \mathbf{x})^t \mathbf{1}\}} \\ & \text{subject to } x_i \in \{0, 1\} \text{ for all } i \\ & \mathbf{x} \text{ is a minimizer at all levels } n > m \end{aligned}$$

562 If $G_{\geq w_m}$ has components $\{C_1, C_2, \dots, C_l\}$, with $l > 1$ consider the following vector

$$563 \quad (55) \quad \mathbf{x}^*(i) = \begin{cases} 1 & \text{if } i \in C_1 \\ 0 & \text{otherwise} \end{cases}$$

564 Then it is easy to see that for all $n > m$, we have

$$565 \quad (56) \quad (\mathbf{x}^*)^t L_n \mathbf{x}^* = 0$$

566 and hence \mathbf{x}^* is a solution to (54).

567 The theorem follows from extending this to all levels.

568

REFERENCES

- 569 [1] V. E. ALEKSEEV, R. BOLIAC, D. V. KOROBITSYN, AND V. V. LOZIN, *NP-hard graph problems and*
570 *boundary classes of graphs*, Theoretical Computer Science, 389 (2007), pp. 219–236.
- 571 [2] S. ALPERT, M. GALUN, R. BASRI, AND A. BRANDT”, *Image segmentation by probabilistic bottom-up*
572 *aggregation and cue integration.*, in Proceedings of the IEEE Conference on Computer Vision and
573 Pattern Recognition, June 2007.
- 574 [3] P. ARBELAEZ, M. MAIRE, C. FOWLKES, AND J. MALIK, *Contour detection and hierarchical image seg-*
575 *mentation*, IEEE transactions on pattern analysis and machine intelligence, 33 (2011), pp. 898–916.
- 576 [4] K. BERNHARD AND J. VYGEN, *Combinatorial optimization: Theory and algorithms*, Springer, Third
577 Edition, 2005., (2008).
- 578 [5] X. BRESSON, X.-C. TAI, T. F. CHAN, AND A. SZLAM, *Multi-class transductive learning based on 1*
579 *relaxations of Cheeger cut and Mumford-Shah-Potts model*, Journal of mathematical imaging and
580 vision, 49 (2014), pp. 191–201.
- 581 [6] A. CHALLA, S. DANDA, B. S. D. SAGAR, AND L. NAJMAN, *An introduction to Gamma-convergence for*
582 *spectral clustering*, in Discrete Geometry for Computer Imagery - 20th IAPR International Conference,
583 DGCI 2017, Vienna, Austria, September 19-21, 2017, Proceedings, W. G. Kropatsch, N. M. Artner,
584 and I. Janusch, eds., vol. 10502 of Lecture Notes in Computer Science, Springer, 2017, pp. 185–196,
585 https://doi.org/10.1007/978-3-319-66272-5_16, https://doi.org/10.1007/978-3-319-66272-5_16.
- 586 [7] A. S. CHALLA, S. DANDA, B. S. S. DAYA SAGAR, AND L. NAJMAN, *Power Spectral Clustering*. working
587 paper or preprint, Jan. 2018, <https://hal.archives-ouvertes.fr/hal-01516649>.
- 588 [8] A. CHAMBOLLE, V. CASELLES, D. CREMERS, M. NOVAGA, AND T. POCK, *An introduction to Total*
589 *Variation for image analysis*, Theoretical foundations and numerical methods for sparse recovery, 9
590 (2010), p. 227.
- 591 [9] K. CHANG, S. SHAO, AND D. ZHANG, *The 1-Laplacian Cheeger cut: Theory and algorithms*, arXiv
592 preprint arXiv:1603.01687, (2016).
- 593 [10] F. R. K. CHUNG, *Spectral Graph Theory*, American Mathematical Society, 1997.
- 594 [11] C. COUPRIE, L. J. GRADY, L. NAJMAN, AND H. TALBOT, *Power watershed: A unifying graph-based*
595 *optimization framework*, IEEE Trans. Pattern Anal. Mach. Intell., 33 (2011), pp. 1384–1399, <https://doi.org/10.1109/TPAMI.2010.200>, <https://doi.org/10.1109/TPAMI.2010.200>.
- 596 [12] J. COUSTY, G. BERTRAND, L. NAJMAN, AND M. COUPRIE, *Watershed cuts: Minimum spanning forests*
597 *and the drop of water principle*, IEEE Trans. Pattern Anal. Mach. Intell., 31 (2009), pp. 1362–1374,
598 <https://doi.org/10.1109/TPAMI.2008.173>, <https://doi.org/10.1109/TPAMI.2008.173>.
- 599 [13] J. COUSTY, G. BERTRAND, L. NAJMAN, AND M. COUPRIE, *Watershed cuts: Thinnings, shortest path*
600 *forests, and topological watersheds*, IEEE Trans. Pattern Anal. Mach. Intell., 32 (2010), pp. 925–939,
601 <https://doi.org/10.1109/TPAMI.2009.71>, <https://doi.org/10.1109/TPAMI.2009.71>.
- 602 [14] S. DANDA, A. CHALLA, B. S. D. SAGAR, AND L. NAJMAN, *Power tree filter: A theoretical frame-*
603 *work linking shortest path filters and minimum spanning tree filters*, in Mathematical Morphology
604 and Its Applications to Signal and Image Processing - 13th International Symposium, ISMM 2017,
605 Fontainebleau, France, May 15-17, 2017, Proceedings, J. Angulo, S. Velasco-Forero, and F. Meyer,
606 eds., vol. 10225 of Lecture Notes in Computer Science, 2017, pp. 199–210, [https://doi.org/10.1007/](https://doi.org/10.1007/978-3-319-57240-6_16)
607 [978-3-319-57240-6_16](https://doi.org/10.1007/978-3-319-57240-6_16), https://doi.org/10.1007/978-3-319-57240-6_16.
- 608 [15] S. DANDA, A. S. CHALLA, B. S. DAYA SAGAR, AND L. NAJMAN, *Some Theoretical Links Between*
609 *Shortest Path Filters and Minimum Spanning Tree Filters*. working paper or preprint, May 2018,
610 <https://hal.archives-ouvertes.fr/hal-01617799>.
- 611 [16] D.-Z. DU AND F. K. HWANG, *A proof of the gilbert-pollak conjecture on the steiner ratio*, Algorithmica,
612 7 (1992), pp. 121–135.
- 613 [17] E. N. GILBERT AND H. O. POLLAK, *Steiner minimal trees*, SIAM Journal on Applied Mathematics, 16
614 (1968), pp. 1–29.
- 615 [18] G. GILBOA, *Graph and nonlocal framework*, in Nonlinear Eigenproblems in Image Processing and Com-
616 *puter Vision*, Springer, 2018, pp. 123–132.
- 617 [19] L. GRADY, *Fast, quality, segmentation of large volumes - isoperimetric distance trees*, in Computer
618 *Vision - ECCV 2006, 9th European Conference on Computer Vision, Graz, Austria, May 7-13,*
619 *2006, Proceedings, Part III, A. Leonardis, H. Bischof, and A. Pinz, eds., vol. 3953 of Lecture*
620 *Notes in Computer Science, Springer, 2006, pp. 449–462, https://doi.org/10.1007/11744078_35,*
621 https://doi.org/10.1007/11744078_35,
622 https://doi.org/10.1007/11744078_35.

- 623 [20] L. GRADY AND E. L. SCHWARTZ, *Isoperimetric graph partitioning for image segmentation*, IEEE Trans.
624 Pattern Anal. Mach. Intell., 28 (2006), pp. 469–475, <https://doi.org/10.1109/TPAMI.2006.57>, <https://doi.org/10.1109/TPAMI.2006.57>.
- 626 [21] L. GRADY AND E. L. SCHWARTZ, *Isoperimetric partitioning: A new algorithm for graph partitioning*,
627 SIAM J. Scientific Computing, 27 (2006), pp. 1844–1866, <https://doi.org/10.1137/040609008>, <https://doi.org/10.1137/040609008>.
- 629 [22] L. GUIGUES, J. P. COCQUEREZ, AND H. LE MEN, *Scale-sets image analysis*, International Journal of
630 Computer Vision, 68 (2006), pp. 289–317.
- 631 [23] L. HUBERT AND P. ARABIE, *Comparing partitions*, Journal of Classification, 2 (1985), pp. 193–218,
632 <https://doi.org/10.1007/BF01908075>, <https://doi.org/10.1007/BF01908075>.
- 633 [24] E. MERKURJEV, A. BERTOZZI, X. YAN, AND K. LERMAN, *Modified Cheeger and ratio cut methods*
634 *using the Ginzburg–Landau functional for classification of high-dimensional data*, Inverse Problems,
635 33 (2017), p. 074003.
- 636 [25] B. MOHAR, *Isoperimetric numbers of graphs*, J. Comb. Theory Ser. B, 47 (1989), pp. 274–291, [https://doi.org/10.1016/0095-8956\(89\)90029-4](https://doi.org/10.1016/0095-8956(89)90029-4), [http://dx.doi.org/10.1016/0095-8956\(89\)90029-4](http://dx.doi.org/10.1016/0095-8956(89)90029-4).
- 638 [26] J.-M. MOREL AND S. SOLIMINI, *Variational methods in image segmentation: with seven image processing*
639 *experiments*, vol. 14, Springer Science & Business Media, 2012.
- 640 [27] L. NAJMAN, *Extending the power watershed framework thanks to Γ -convergence*, SIAM J. Imaging
641 Sciences, 10 (2017), pp. 2275–2292, <https://doi.org/10.1137/17M1118580>, <https://doi.org/10.1137/17M1118580>.
- 643 [28] J. PONT-TUSET AND F. MARQUÉS, *Supervised evaluation of image segmentation and object proposal*
644 *techniques*, IEEE Trans. Pattern Anal. Mach. Intell., 38 (2016), pp. 1465–1478, <https://doi.org/10.1109/TPAMI.2015.2481406>, <https://doi.org/10.1109/TPAMI.2015.2481406>.
- 646 [29] N. X. VINH, J. EPPS, AND J. BAILEY, *Information theoretic measures for clusterings comparison: Vari-*
647 *ants, properties, normalization and correction for chance*, J. Mach. Learn. Res., 11 (2010), pp. 2837–
648 2854, <http://dl.acm.org/citation.cfm?id=1756006.1953024>.
- 649 [30] U. VON LUXBURG, *A tutorial on spectral clustering*, Statistics and computing, 17 (2007), pp. 395–416.
- 650 [31] Y. XU, E. CARLINET, T. GÉRAUD, AND L. NAJMAN, *Hierarchical segmentation using tree-based shape*
651 *spaces*, IEEE transactions on pattern analysis and machine intelligence, 39 (2017), pp. 457–469.