Submodularity Beyond Submodular Energies: Coupling Edges in Graph Cuts

Stefanie Jegelka
Max Planck Institutes Tübingen
jegelka@tuebingen.mpg.de

Jeff Bilmes
University of Washington
bilmes@ee.washington.edu

Abstract

We propose a new family of non-submodular global energy functions that still use submodularity internally to couple edges in a graph cut. We show it is possible to develop an efficient approximation algorithm that, thanks to the internal submodularity, can use standard graph cuts as a subroutine. We demonstrate the advantages of edge coupling in a natural setting, namely image segmentation. In particular, for fine-structured objects and objects with shading variation, our structured edge coupling leads to significant improvements over standard approaches.

1. Introduction

For many years, Markov random fields (MRF) have been seen as a natural fit to solve various problems in computer vision [12]. In such models, finding a maximizing assignment of random variable values corresponds to minimizing a Gibbs energy. This minimization is in general not only NP-hard, but some models do not even admit any nontrivial approximation guarantees [5]. Consequently, for image processing, some early researchers considered MRFs destined for no more than a theoretical curiosity.

Recently, subclasses of MRFs were shown to be not only easy to exactly optimize (without a tree-width restriction) but also quite naturally applicable to many computer vision problems [4, 12, 23]. Specifically, finding the minimum energy configuration is very efficient for those Gibbs energy functions whose variable assignment costs correspond exactly to cut costs in an appropriate graph [8, 23]. Graph cuts are now successfully used in segmentation, stereo matching, and texture synthesis, among others.

Inspired by these results, a principal goal has become identifying the most general classes of energies that can be exactly optimized either directly or indirectly via graph cuts. For example, while some binary pairwise potential functions can be solved exactly using graph cuts, in many cases higher order (*e.g.*, k-ary) potential functions [8, 23, 33] and potentials functions over non-binary variables [4] can also be solved efficiently. In all cases, a critical property known as “regularity” [23] (more generally, submodularity [9]) is used.

Unfortunately, there are critical deficiencies when graph cuts are used in practice, partly stemming from their inability to represent more than only a limited class of energies [8, 10, 23, 33]. The core issue is that graph cuts model an energy that decomposes into *pairwise* terms with *nonnegative* weights. The direct use of such energies can cause insurmountable over-smoothing in image segmentation. While some higher order energies are graph-representable, this representation might regrettably require additional variables which also might not remain computationally feasible [33]. Recent research, therefore, has aimed to identify practically manageable higher order energies [15, 18, 19, 24], and to develop efficient optimization methods for non-submodular potentials [22].

In this work, we define a new powerful class of arbitrarily high order non-submodular energy functions that abandons neither the existence of an underlying graph, nor the use of submodularity, nor practical efficiency. This class is structurally and conceptually very different from the recently considered potentials in [15, 18, 24]. To wit, we make the following critical observation: graph cut based energy functions can be significantly enhanced if the cost of the edges that constitute a cut is measured **not** merely based on the sum of the edge weights. Rather, in our work, any or all the edges in a graph may interplay in complex ways. Formally, let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph where each (s, t)-cut induces an assignment of pixel labels. We replace the usual cut cost (the sum of edge weights) by a submodular cut cost, and we therefore say that the edges themselves may *cooperate* [30]. Doing so introduces the following problem:

Definition 1 (Minimum Cooperative Cut)**.** *Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a nondecreasing submodular function $f : 2^{\mathcal{E}} \to \mathbb{R}_+$ defined on subsets of* **edges** *$\mathcal{E}$, find an (s, t)-cut $\Gamma \subseteq \mathcal{E}$ having minimum cost $f(\Gamma)$.*

As shown below, the equivalent energy functions are not in general submodular and cooperative cut is NP-hard, even though submodularity is, in a sense, “internal” to our problem as will be seen. The graph structure is key to obtain an efficient approximation algorithm.

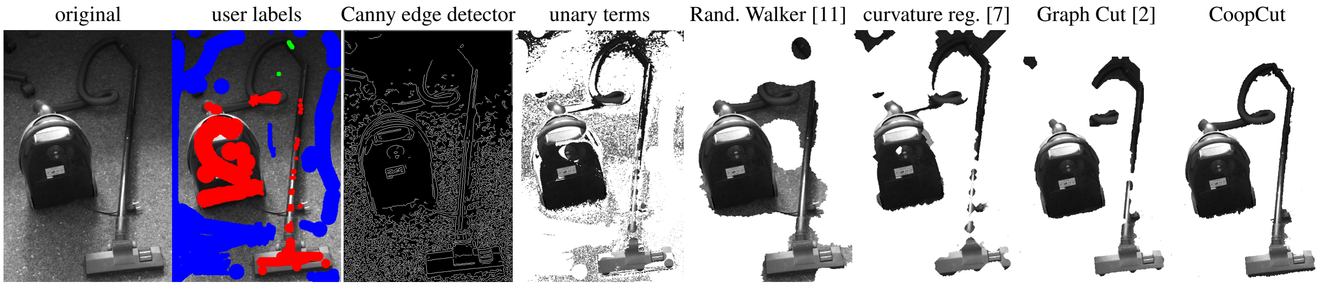


Figure 1. Segmentation results for an image with shading. The task is difficult despite many labels. All algorithms used the same unary terms, except for the Random Walker, which got enhanced seeds (green). Column 4 is the segmentation obtained from unary terms alone.

Edge cooperation naturally captures information that is missed by many existing approaches, such as global features of object boundaries in a segmentation. For example, consider the vacuum cleaner in Figure 1 (left): low contrast makes the tube difficult to identify, so that either background is included in the foreground, or parts of the tube are cut off. Such incorrect boundaries (chosen due to shortcutting) are qualitatively different from the correct boundary between the patterned carpet and the vacuum cleaner. We maintain that the boundary is "congruous": along it lies a repetitive pattern made up from the few adjoining texture types that, if properly represented, can help boundary identification. By globally coupling boundary sub-segments across lighted and shaded regions, such congruity can be exploited, and this is easily expressible by cooperative cuts as we show below.

More specifically, we show: 1) a new class of powerful energy functions having arbitrarily high order, where the potential functions and maximum order may automatically and efficiently adapt to each image; 2) an optimization method that is remarkably efficient and practical, and that uses standard graph cuts as a subroutine; 3) theoretical approximation guarantees for our optimization method; 4) a specific edge-cooperative potential for segmenting considerably difficult images that, compared to graph cuts, reduces the segmentation error by up to 70%. In particular, we show significant improvements on images having the potential for a severe shrinking bias problem, on images that possess light intensity gradients and shadows, and on images with both these difficulties simultaneously. Finally, we relate edge cooperation to other recent approaches in computer vision.

2. Background: Graph Cuts & Gibbs Energies

Before describing cooperative cuts, we recall the relationship between graph cuts and energies and in doing so define our notation. Labeling image pixels is often formulated as inference in an MRF. For each pixel i in an image I, a random variable takes values from a set $\mathcal{L}$ of labels. For simplicity, we consider only binary labels ($|\mathcal{L}| = 2$). A Gibbs energy $E(\mathbf{x}; \bar{\mathbf{z}})$ over labelings $\mathbf{x} = \{x_1, \ldots, x_{|I|}\} \in \mathcal{L}^{|I|}$ defines the probability $p(\mathbf{x}|\bar{\mathbf{z}}) \propto \exp(-E(\mathbf{x}; \bar{\mathbf{z}}))$ of a labeling given observed pixel values $\bar{\mathbf{z}}$. The energy decomposes into a sum of unary potential functions, making a connection to the image $\bar{\mathbf{z}}$, and a sum of clique potentials $\{\psi_C : C \in \mathcal{C}\}$, where $\mathcal{C}$ is the set of maximal cliques in the MRF. That is,

$$E(\mathbf{x}; \bar{\mathbf{z}}) = \sum\nolimits_i \psi_i(x_i, \bar{\mathbf{z}}_i) + \sum\nolimits_{C \in \mathcal{C}} \psi_C(\mathbf{x}_C). \quad (1)$$

As $\bar{\mathbf{z}}$ is constant, using $E(\mathbf{x}) = E(\mathbf{x}; \bar{\mathbf{z}})$ simplifies notation.

A pixel labeling is produced by finding a maximum a posteriori (MAP) (equivalently, energy minimizing) variable assignment, *i.e.*, $\mathbf{x}^* \in \mathrm{argmax}_{\mathbf{x}}\, p(\mathbf{x}|\bar{\mathbf{z}}) = \mathrm{argmin}_{\mathbf{x}}\, E(\mathbf{x})$. For a sub-family of energies, energy minimization is equivalent to a minimum (s,t)-cut in a corresponding graph [12]. A key graph-cut-enabling ingredient is "regularity," defined as follows in the pairwise case ($|C| = 2, \forall C$): for all $\{i, j\} = C \in \mathcal{C}$, we have

$$\psi_{i,j}(0,1) + \psi_{i,j}(1,0) \geq \psi_{i,j}(0,0) + \psi_{i,j}(1,1). \quad (2)$$

Graph cuts arise naturally via the relationship between energy functions and set functions on nodes of a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Given a set $\mathcal{V} = \{v_1, v_2, \ldots, v_{|I|}\}$, one element per pixel, define the mapping $X(\mathbf{x}) = \{v_i \in \mathcal{V} : x_i = 1\}$ from labelings to sets. Then, the energy $E(\mathbf{x}) = \Psi(X(\mathbf{x}))$ has a corresponding set function Ψ, and regularity of E is equivalent to submodularity of $\Psi(X)$. A function $\Psi : 2^{\mathcal{V}} \to \mathbb{R}$ is *submodular* if for all $X, Y \subseteq \mathcal{V}$, we have $\Psi(X) + \Psi(Y) \geq \Psi(X \cap Y) + \Psi(X \cup Y)$ [9]. If this condition holds everywhere with equality, then Ψ is called *modular* (*i.e.*, $\Psi(X) = \sum_{x \in X} a_x$ for some $\mathbf{a} \in \mathbb{R}^{|I|}$).

To represent a pairwise submodular energy $E(\mathbf{x})$ as a graph cut, define a weighted directed graph $\mathcal{G} = (\mathcal{V} \cup \{s,t\}, \mathcal{E}, w)$ having a node $v_i \in \mathcal{V}$ for each image pixel, and two terminal nodes s, t. The edges $\mathcal{E}$ consist of inter-pixel edges $\mathcal{E}_n$ and terminal edges $\mathcal{E}_t$. Each potential $\psi_{ij}(x_i, x_j)$ corresponds to two edges $(v_i, v_j), (v_j, v_i) \in \mathcal{E}_n$, and each unary potential $\psi_i(x_i)$ corresponds to the edges $(s, v_i), (v_i, t) \in \mathcal{E}_t$ (although this is often done using undirected graphs, directed graphs better suit our needs). A minimal (s,t)-cut $\Gamma \subseteq \mathcal{E}$ defines a labeling by assigning 1 to x_i if v_i is uncut from s, and 0 otherwise. Equally, an assignment $\mathbf{x}$ defines an (s,t)-cut in $\mathcal{G}$. Let $X_1 = X(\mathbf{x}) \cup \{s\}$ and $X_0 = (\mathcal{V} \setminus X(\mathbf{x})) \cup \{t\}$, then $\Gamma(X(\mathbf{x})) = \mathcal{E} \cap (X_1 \times X_0)$

is a set of edges defining a cut. Given edge weights $w : 2^{\mathcal{E}} \to \mathbb{R}_+$, the cost of a cut Γ is usually the sum of weights $w(\Gamma) = \sum_{e \in \Gamma} w(e)$, which is a modular function of *edge* sets. This cut cost, if seen as a function of sets of *nodes*, is $\Psi_w(X) \triangleq w(\Gamma(X))$, a function well known to be submodular on $2^{\mathcal{V}}$. Moreover, for the pairwise regular energies $E(\mathbf{x})$ there exists a w such that [23]

$$E(\mathbf{x}) + \text{const} = w(\Gamma(X(\mathbf{x}))) = \sum_{e \in \Gamma(X(\mathbf{x}))} w(e). \tag{3}$$

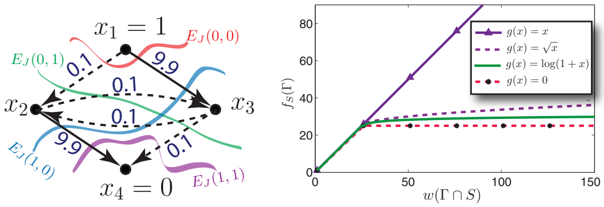
3. Cooperative Graph Cuts

The modularity of the edge weights w in the cut cost (3) is a critical structural limitation: cutting one edge has no effect on the cost of cutting a different edge. Modular edge weights allow efficient graph cut algorithms but can also have deleterious effects on computer vision results.

In our approach, by contrast, cutting an edge may influence the cost of cutting other edges. We express this influence by measuring the cost of a cut using a nondecreasing nonnegative submodular function $f : 2^{\mathcal{E}} \to \mathbb{R}_+$ defined on subsets of **edges** $\mathcal{E}$ (in stark contradistinction to Section 2, where submodular functions are defined on subsets of nodes $\mathcal{V}$). Because f is submodular and nonnegative, it is also *subadditive*: $f(A \cup B) \leq f(A) + f(B)$. If the inequality is strict, we will say that edges in A and B *cooperate* [30].

The weight of a *cooperative cut* between nodes X and $\mathcal{V} \setminus X$ can be expressed as the node function $\Psi_f(X) = f(\Gamma(X))$. Thus, cooperative cut leads to a family of energies of the form $E_f(\mathbf{x}) \triangleq f(\Gamma(X(\mathbf{x})))$. This has three consequences. First, MAP inference for E_f reduces to *Minimum Cooperative Cut*. Second, depending on f, there can be cooperation between arbitrarily large edge sets *anywhere* in the graph. Since this couples all nodes adjacent to the cooperating edges, E_f has arbitrarily high order. Third, E_f is not necessarily regular (equivalently, Ψ_f is not necessarily submodular). Figure 2(a) shows a cooperative energy E_f that violates regularity. A higher order energy E is *regular* [23] if all of its projections on any pair of variables i, j are regular. Let $J = I \setminus \{i, j\}$. A projection E_J of $E : \{0,1\}^I \to \mathbb{R}_+$ on i, j is obtained by fixing the values of $\mathbf{x}_J$ to some $\bar{\mathbf{x}}_J$, and setting $E_J(x_i, x_j) = E(x_i, x_j, \bar{\mathbf{x}}_J)$.

Submodular functions can reward the *co-occurrence* of certain elements (here, edges of a graph cut). Useful submodular functions include (i) $f(A) = g(\sum_{e \in A} w(e))$ for nonnegative weights w and any concave, nondecreasing function $g : \mathbb{R}_+ \to \mathbb{R}_+$ [30]; (ii) cover-type functions $f(A) = |\bigcup_{e \in A} S_e|$, where each e has an associated set or area S_e; (iii) entropy; and (iv) neighborhood functions in bipartite graphs. Moreover, the sum of submodular functions is submodular. Additional flexibility is gained by the graph structure, as will be seen in Sections 5 and 6.



(a) non-regular E_f

(b) examples of g

Figure 2. (a) Example of a non-regular energy E_f with $f(A) = \left(\sum_{e \in A} w(e)\right)^{1/2}$. Edge weights are as indicated. An edge is cut if its tail has label 1 and its head label 0. Consider the projection $E_J(x_2, x_3)$ for $J = (1,4)$ and $\bar{\mathbf{x}}_J = (1,0)$. Then $E_J(1,0) + E_J(0,1) = \sqrt{9.9 + 0.1 + 9.9} + \sqrt{0.1 + 0.1 + 0.1} < 5.01 < 6.32 < \sqrt{0.1 + 9.9} + \sqrt{0.1 + 9.9} = E_J(0,0) + E_J(1,1)$, violating regularity. (b) Effect of different gs in Eqn. (13).

4. Optimization

To minimize E_f, we must solve a Minimum Cooperative Cut. While coupling edges allows E_f to be non-regular, this also makes the problem NP-hard:

Theorem 1. *Minimum Cooperative Cut is NP-hard.*

The proof is a reduction from Graph Bisection [17]. On the other hand, the graph structure provides a definitive advantage over general higher order potentials — for some global energies, no algorithm can provide quality guarantees on the solutions it finds [5, 16]. In contrast, we now derive a practical and efficacious approximation algorithm for cooperative cuts that does have an approximation guarantee. It iteratively minimizes an upper bound on $E_f(\Gamma)$.

The simplest upper bound on a submodular function $f(A)$ is its modular counterpart $\hat{f}(A) = \sum_{e \in A} f(e)$, but this ignores all coupling inherent in f. We instead develop an *adjusting* bound that largely retains cooperation. Define, for $B \subseteq \mathcal{E}$ and an edge $e \in \mathcal{E}$, the *marginal cost* of e with respect to B as $\rho_e(B) = f(B \cup e) - f(B)$. Submodularity implies *diminishing marginal costs:* $\rho_e(B) \leq \rho_e(A)$ for all $A \subseteq B \subseteq \mathcal{E} \setminus \{e\}$.

Lemma 1. *For a submodular $f : 2^{\mathcal{E}} \to \mathbb{R}_+$, and an arbitrary $B \subseteq \mathcal{E}$, define $h_{B,f} : 2^{\mathcal{E}} \to \mathbb{R}_+$ as*

$$h_{B,f}(A) \triangleq f(B) + \sum_{e \in A \setminus B} \rho_e(B) - \sum_{e \in B \setminus A} \rho_e(\mathcal{E} \setminus \{e\}). \tag{4}$$

The function $h_{B,f}$ is a modular upper bound on f.

Proof. For any sets $A, B \subseteq \mathcal{E}$, it holds that [26]

$$f(A) \leq f(B) + \sum_{e \in A \setminus B} \rho_e(B) - \sum_{e \in B \setminus A} \rho_e((A \cup B) \setminus \{e\}). \tag{5}$$

Bound (4) follows by diminishing marginal costs: $\rho_e(\mathcal{E} \setminus \{e\}) \leq \rho_e((A \cup B) \setminus \{e\})$. Modularity is immediate. □

This bound adds an upper bound on the cost of $A \setminus B$ and subtracts a lower bound on the cost of $B \setminus A$. The bound, moreover, is tight at $A = B$, *i.e.*, $h_{B,f}(B) = f(B)$.

Importantly, the cut cost $h_{B,f}$ is efficient to minimize using standard minimum cut, thanks to its modularity. For $\mathcal{G} = (\mathcal{V}, \mathcal{E}, f)$, define $\mathcal{G}_B = (\mathcal{V}, \mathcal{E}, w_B)$ with edge weights

$$w_B(e) = \begin{cases} \rho_e(\mathcal{E} \setminus \{e\}) & \text{if } e \in B \\ \rho_e(B) & \text{otherwise.} \end{cases} \tag{6}$$

For a nondecreasing f, the weights w_B are nonnegative.

Lemma 2. *The minimum (s,t)-cut in $\mathcal{G}_B$ is a minimizing cut for the bound $h_{B,f}$.*

Proof. With weights w_B, the cost of a cut $\Gamma \subseteq \mathcal{E}$ is

$$\sum_{e\in\Gamma} w_B(e) = \sum_{e\in B\cap\Gamma} \rho_e(\mathcal{E} \setminus \{e\}) + \sum_{e\in\Gamma\setminus B} \rho_e(B) \tag{7}$$

$$= h_{B,f}(\Gamma) - f(B) + \sum_{e\in B} \rho_e(\mathcal{E} \setminus \{e\}). \tag{8}$$

Since $f(B)$ and the sum are constant for a fixed B, $w_B(\Gamma) = h_{B,f}(\Gamma) + \mathit{const}$ for any edge set $\Gamma \subseteq \mathcal{E}$. □

Using $h_{B,f}$, we derive an iterative minimization procedure (Algorithm 1). Given an initial reference set B, we find the minimum cut Γ with respect to $h_{B,f}$. Then we adjust the bound to be tight at Γ and repeat. Thus, $h_{B,f}$ is always tight at the currently best solution. The algorithm starts with an initial reference set $I_j \in \mathcal{I}$, the simplest case of which is $\mathcal{I} = \{\emptyset\}$. For further improvements, other options include setting $\mathcal{I}$ to the elements of a cut basis, *e.g.*, the cuts induced by cutting edges of a spanning tree. For our experiments in Section 8, however, $\mathcal{I} = \{\emptyset\}$ was sufficient, and the algorithm converged in less than 10 iterations.

Algorithm 1: Iterative bound minimization

```
Input: $G = (\mathcal{V}, \mathcal{E})$; submodular cost $f\colon 2^{\mathcal{E}} \to \mathbb{R}_0^+$;
       reference initialization set $\mathcal{I} = \{I_1, \dots, I_k\}$,
       $I_j \subseteq \mathcal{E}$; source /sink $s, t \in \mathcal{V}$
Output: cut $B \subseteq E$
for $j = 1$ to $k$ do
    find $(s,t)$-mincut $\Gamma$ for edge weights $w_{I_j}$;
    repeat
        $B_j = \Gamma$;
        find $(s,t)$-mincut $\Gamma$ for edge weights $w_{B_j}$;
    until $f(\Gamma) \geq f(B_j)$;
return $B = \mathrm{argmin}_{B_1,\dots,B_k} f(B_j)$;
```

As a result of Lemma 2, the algorithm alternates between adjusting weights and computing a minimum cut. Implementation efficiency can be improved by noting that the marginal cost of an edge e depends only on edges that cooperate with e. The weights w_B show how h_B captures the cost-reducing effect of f: $\rho_e(B) < f(e)$ if e cooperates with B. For a modular function $f = f_m$, $\rho_e(B) = f_m(e)$ and Algorithm 1 becomes the standard minimum cut.

Lemma 3 gives an approximation bound for the initial solution $\Gamma_\emptyset$ for $h_{\emptyset,f}$, which improves in subsequent iterations.

Lemma 3. *Let $\Gamma_\emptyset \in \mathrm{argmin}\{h_{\emptyset,f}(\Gamma) \mid \Gamma \subseteq \mathcal{E}$ an (s,t)-cut$\}$ be a minimum cut for $h_{\emptyset,f}$, and $\Gamma^* \in \mathrm{argmin}\{f(\Gamma) \mid \Gamma \subseteq \mathcal{E}$ an (s,t)-cut$\}$ an optimal solution. Let $\nu(\Gamma^*) = \min_{e\in\Gamma^*} \rho_e(\Gamma^* \setminus \{e\}) / \max_{e\in\Gamma^*} f(e)$. Then*

$$f(\Gamma_\emptyset) \leq \frac{|\Gamma^*|}{1 + (|\Gamma^*| - 1)\nu(\Gamma^*)} f(\Gamma^*) \quad \leq \quad |\Gamma^*| f(\Gamma^*).$$

The proof is deferred to [16]. For the functions we use in Section 5, the term $\nu(\Gamma^*)$ is always nonzero and the second inequality is strict. Lemma 3 is a worst case bound and holds for *any* nondecreasing submodular f. In practice, the algorithm usually performs much better [17].

5. Structured cooperation for segmentation

We now apply edge cooperation to interactive figure-ground segmentation, where, given initial user input, the remaining pixels are to be labeled as object or background. In particular, we address the problems shown in Figure 1: while graph cuts have been used successfully for this task, they are known to shortcut elongated boundaries, especially in low contrast, shaded regions (see also Figs. 3, 4, 5). These failures are caused by the commonly used pairwise energy inherent to standard grid-structured MRFs:

$$E(\mathbf{x}) = \sum_{i\in I} \psi_i(x_i) \quad + \quad \lambda \sum_{(i,j)\in\mathcal{E}_n} \psi_{ij}(x_i, x_j) \tag{9}$$

$$= w(\Gamma(X(\mathbf{x})) \cap \mathcal{E}_t) + \lambda w(\Gamma(X(\mathbf{x})) \cap \mathcal{E}_n) + \text{const.}$$

The associated graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ has terminal edges $\mathcal{E}_t$, and a grid of inter-pixel edges $\mathcal{E}_n$ expressing the pairwise potentials. While the former integrate user interaction, the latter enforce smoothness and coherency. The edge weights on $\mathcal{E}_n$ are a function of the intensity gradient; their sum may be seen as the weighted length of the boundary. This penalty favors short boundaries, and thus results in the aforementioned shortcutting. Lowering the coefficient λ is not a solution since boundaries become noisy and true background is included into the hypothesized foreground (Fig. 5).

Instead, we utilize edge cooperation to selectively reward global features of true boundaries. Specifically, we retain $\mathcal{G}$ and replace only the over-smoothing inter-pixel cut by a cooperative cut:

$$E_f(\mathbf{x}) = w(\Gamma(X(\mathbf{x})) \cap \mathcal{E}_t) + \lambda f(\Gamma(X(\mathbf{x})) \cap \mathcal{E}_n) \tag{10}$$

$$= \sum\nolimits_{i\in I} \psi_i(\mathbf{x}_i) \quad + \quad \lambda f(\Gamma(X(\mathbf{x})) \cap \mathcal{E}_n). \tag{11}$$

Since an object boundary consists of cut edges in $\mathcal{G}$, we desire a submodular edge cost f that captures desirable boundary features. We observe that, along true object boundaries, many images possess a certain *congruence*, and this may be true globally throughout the image. Boundary congruity materializes in a number of contexts. For example, of the many inter-pixel color gradients in Figures 3, 5, only few occur along the true boundary in difficult regions; and shortcutting introduces new, incongruous, boundary types. Moreover, the repetitiveness of patterned backgrounds retains congruity to a large extent. Similarly, there is congruity between lighted and shaded regions in Figure 1, if shade is neutralized. In this latter case, we thus need a *shade-invariant* congruity criterion.

Consequently, f should (i) decrease the penalty for globally congruous boundaries, (ii) retain the common smoothing effect of pairwise potentials for incongruous boundaries, and (iii) allow automatic and efficient adaptation of the congruence criterion to each image.

We define congruity in terms of classes of similar edges, $\mathcal{S}(\bar{\mathbf{z}}) = \{S_1, S_2, \ldots, S_\ell\}$, $S_i \subseteq \mathcal{E}_n$ and $\mathcal{E}_n = \bigcup_i S_i$. Congruous boundaries use few classes. Submodularity may selectively reward congruity since it possesses diminishing marginal costs. We make f submodular, however, only *within* classes, and modular across classes:

$$f(\Gamma) = \sum\nolimits_{S \in \mathcal{S}(\bar{\mathbf{z}})} f_S(\Gamma \cap S). \tag{12}$$

As a result, (i) the marginal cost of an edge decreases only when enough edges from the same class are cut. The discount increases with the number of edges included from that class. On the other hand, (ii) there is no discount for cuts that use edges from many classes, *i.e.*, incongruous cuts. The class costs f_S are thresholded discount functions,

$$f_S(\Gamma) = \begin{cases} w(\Gamma \cap S) & \text{if } w(\Gamma \cap S) \leq \theta_S \\ \theta_S + g(w(\Gamma \cap S) - \theta_S) & \text{if } w(\Gamma \cap S) > \theta_S \end{cases}, \tag{13}$$

for any nondecreasing, nonnegative concave function $g : \mathbb{R} \to \mathbb{R}$. For our experiments, we chose $g(x) = \sqrt{x}$. Alternatives include $g(x) = \log(1+x)$ or roots $g(x) = x^{1/p}$ (Figure 2(b)). The modular case (9) corresponds to $g(x) = x$.

To adapt f, we infer the classes by clustering edges $\mathcal{E}_n$ for each image. Furthermore, the discount only sets in after a threshold θ_S is reached, and we adapt θ_S to the total weight of the class, *i.e.*, $\theta_S = \vartheta w(S)$ for $\vartheta \in [0, 1]$, which improves scale-invariance. For large objects or images, more edges are in a class, requiring more cutting to observe a discount. The factor ϑ trades off between completely modular cuts ($\vartheta = 1$) and completely cooperative cuts ($\vartheta = 0$).

The quantitative gauge of "congruence" depends on the distance measure used to cluster the edges. For an edge $e = (v_i, v_j)$ with observed pixel values z_i, z_j, we define two possible feature vectors $\phi(e)$: (i) for uniformly lit images, we use linear color gradients, $\phi_l(e) = z_j - z_i$, and squared Euclidean distance for clustering; and (ii) for shading, we use log intensity ratios $\phi_r(e) = \log(z_j/z_i)$ (channel-wise for color images) which are approximately invariant to shading, and ℓ_1 distance for clustering. In each case, we use the features ($\phi_l(e)$, or $\phi_r(e)$) for clustering edges, and use the standard weights $w(e)$ inside of f.

potential	cooperating edges	$f(\Gamma)$
Graph Cut	$\mathcal{E}$	$w(\Gamma)$
congruence (Sec. 5)	groups of $\mathcal{E}_n$	$\sum_S g_\theta(w(\Gamma \cap S))$
(binary) P^n [19]	$\mathcal{E}$ in $\mathcal{G}'$	$g(\lvert\Gamma\rvert)$
rand. walker [11]	$\mathcal{E}$	$\sqrt{w^2(\Gamma)}$
ℓ_∞ [31]	$\mathcal{E}_n$	$\max_{e \in \Gamma \cap \mathcal{E}_n} w(e)$
class labels [25]	$\mathcal{E}_t$	$f_L(\bigcup_{e \in \Gamma} l(e))$

Table 1. Examples of cooperative cuts; $l(e)$ is the label of edge e [16], and $w(\Gamma)$ ($w^2(\Gamma)$) the sum of (squared) weights.

6. The expressive power of cooperative cuts

Cooperative cuts cover (and strictly generalize) a number of recent approaches in computer vision (summarized in Table 1). Note, however, that cooperative cut is not a special case of any these methods (*e.g.*, some are not NP-hard).

Kohli *et al.* [19] consider $\mathcal{P}^{|C|}$ potentials of the form $\psi_C(\mathbf{x}_C) = g(\sum_{i,j \in C} \tilde{\psi}_{C,i,j}(x_i, x_j))$ for a concave, nondecreasing function g and clique C. Because of the structure of their ψ_C, in the binary case, the sum of pairwise potentials $\tilde{\psi}_{C,i,j}$ is representable as cuts in a graph. These potentials are special cases of cooperative cut potentials that remain regular [19], unlike the example in Figure 2. Similarly, the $\alpha\beta$ swap for a multi-label ψ_C is a cooperative cut, as is the α expansion if $\tilde{\psi}$ is a metric [16]. The $\mathcal{P}^n$ *Potts model* [19] and robust P^n potentials [20] are regular special cases of cooperative cut as well [16].

In class-based image segmentation, each pixel must be labeled by an object class. Ladický *et al.* [25] suggest a global potential $f_L(L(\mathbf{x}))$ on the set of class labels $L(\mathbf{x})$ used in $\mathbf{x}$. If $f_L : 2^{\mathcal{L}} \to \mathbb{R}_+$ is nondecreasing and submodular, then the α expansion can be formulated as a cooperative cut on $\mathcal{E}_t$ [16]. The co-occurrence function $f_L(L)$ in [25] is not submodular with respect to class labels. An alternative, submodular f_L could count the number of training images whose labels do not contain the entire set $L(\mathbf{x})$. The label-cost function in [6] is submodular and thus a cooperative cut on $\mathcal{E}_t$, as it corresponds to a neighborhood function in a bipartite graph [16].

Lastly, Sinop and Grady [31] express an objective for variants of the Random Walker algorithm [11] as $E(\mathbf{x}) = (\sum_{(i,j) \in \mathcal{E}} w_{ij}^q |x_i - x_j|^q)^{1/q}$. In a discrete version, $|x_i - x_j| = 1$ if and only if the edge (v_i, v_j) is cut. Since the qth root is concave for $q \geq 1$, $f(\Gamma) = (w^q(\Gamma))^{1/q}$ is submodular. The same holds for the $q \to \infty$ version $E(\mathbf{x}) = \max_{(i,j) \in \mathcal{E}} w_{ij}|x_i - x_j|$. Therefore, the discrete case of [31] is a cooperative cut as well.

7. Other related work

Starting with [12], graph cuts have become standard in computer vision, with many applications [2, 3, 4, 14, 13]. In the standard case, the cut represents a pairwise, regular energy, but graph cuts can also be used for non-submodular pairwise potentials [22] and ratio problems [21].

Beyond pairwise energies, efficiently optimizable higher order potentials have been the subject of many recent studies ([15, 18, 24] and references therein), but the structure of those potentials is very different from edge cooperation. Examples of higher order constraints include single global constraints such as connectivity [27], statistical constraints [25], or clique potentials enforcing homogeneity for groups of nodes [19]. While user-interactive connectivity has been used to tackle shrinking bias [32], it may become tedious for trees (Fig. 5), and does not address holes (Fig. 4).

8. Experiments

For the task of interactive figure-ground segmentation, our experiments address three main questions: (i) What is the effect of coupling edges, and does this strengthen correct boundaries? We compare E_f (CoopCut) from Section 5 to the standard graph cut (GC) [2] for pairwise potentials. (ii) What is the effect of the structure of coupling, *i.e.*, the classes S_i? (iii) Does edge cooperation harm the segmentation of objects requiring standard smoothing?

We use color and grayscale images of complicated objects, with and without shading. Since, to our knowledge, no public database exists for such images, we created our own hand-labeled collection. Images and code are available at `ssli.ee.washington.edu/~jegelka/cc`. For (iii), we use the Grabcut data [1, 28].

Both methods use the same 8-neighbor graph structure, the same unary potentials and inter-pixel edge weights $w(e) = 2.5 + 47.5 \exp(-0.5\|z_i - z_j\|^2/\sigma)$ for edges $e = (v_i, v_j)$ and variance σ of color gradients (parameters as in [32]). The unary potentials are either based on color histograms [2] or on Gaussian mixture models (GMMs) with 5 components [28, 32]. Edge classes are inferred by k-means clustering, and edges between identically colored pixels form an extra class S' with no discount ($\vartheta_{S'} = 1$). Errors are the percentage of wrongly assigned unlabeled pixels. To quantify the recovery of fine object parts that only make up a small fraction of the pixels, we compute the "twig error" on these delicate parts only. We chose good parameters for each method. In Tables 2, 3, parameters are the same on all images. All algorithms were implemented in C++, using the graph cut code [3], OpenCV, and some Matlab pre-processing. For details and more results, see [16].

The results show that (i) cooperation helps to track boundaries into shaded regions, and preserves fine segments; (ii) what matters is the *structure of cooperation*; and (iii) the improvements on complicated boundaries do not harm the results for "standard" boundaries.

Experiment 1: Shading gradient. Table 2 shows segmentation errors on shaded objects in (a) 8 grayscale and (b) 7 color images. On such images, the unary terms are very noisy (Figs. 1,3). Coupling edges using ϕ_r reduces the error, compared to GC, by up to two thirds. Figures 3 and 4 show that CoopCut recovers the object shape much better. To ensure that not the mere ratio information but *cooperation* makes the difference, we ran GC with ("log") edge weights derived from ϕ_r: the errors improve only slightly. To probe the effect of the classes S_i, we compare against a cooperative cut with only one class (plus the class S'). Such uniform, unstructured coupling is much less effective, *i.e.*, the structure implied by the S_i is crucial.

CoopCut does not model shading explicitly, but cancels shading effects via ϕ_r. Thus, it also improves results if the shade varies locally with higher frequency. We artificially shaded images from Expt. 2, by multiplying the pixel at location (x, y) by $0.4(1 + \sin(2\pi y/\gamma))$ ($\gamma \in [10, 120]$). Unary terms were computed from the modified image. Figure 4 shows an example, and Table 2 lists average errors over 18 such images. Indeed, CoopCut halves the total error of GC, and preserves delicate structures much better than GC.

Experiment 2: Thin, elongated parts and holes. To examine the effect of coupling in uniformly lit images, we compute the total and twig error for 17 images with delicate objects. Table 2(d) and Figure 5 show results for two parameter settings: (1) low overall error, and (2) low twig error. Graph cut roughly recovers fine structures if the smoothness term is reduced, but at the price of a high overall segmentation error. CoopCut preserves fine parts without including pieces of background. Total and twig error are minimized *simultaneously*. In comparison, curvature regularization (as in [7], with our unary terms) is more sensitive to noise in the unary terms (which are less noisy in [7, 29]).

Experiment 3: Grabcut data. As a "sanity check", we address the effect of cooperation with objects that are rounder and need regularization. Table 3 displays the errors for GC and CoopCut on the 50 images of the Grabcut data set [1, 28] with the "Lasso" labeling. Even here, CoopCut slightly improves the results on both color models. Figure 3 shows segmentations for two objects where GC faces the shrinking bias and CoopCut recovers the shape.

The optimal parameter choice varies slightly with the setting, like with standard graph cuts, but the errors show that one choice is reasonable for a wide range of images.

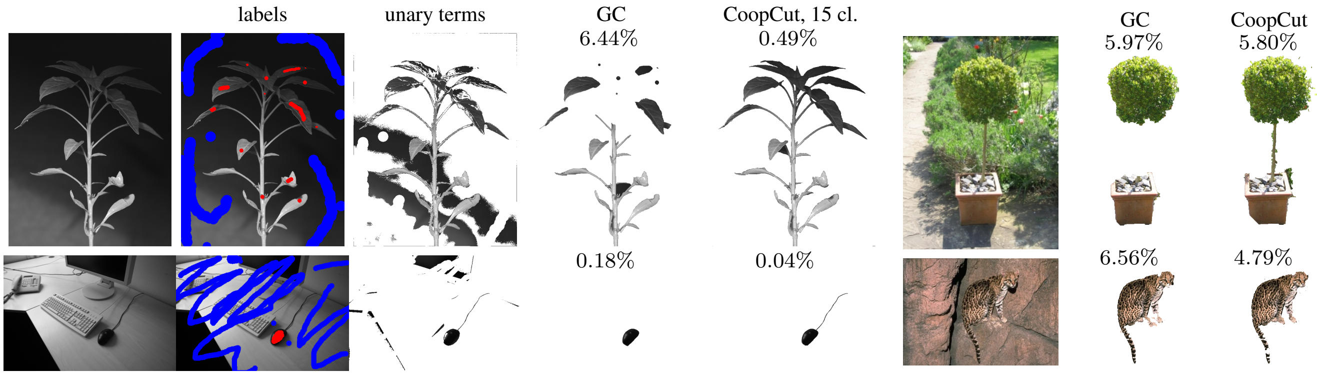


Figure 3. Example results and errors for Expt. 1 (left) and the Grabcut data (right). GC has minimum-error parameters $\lambda = 1.2, 1.0$; CoopCut $(\lambda, 10^4\vartheta) = (8, 5)$ on both images. Grabcut data: GC $\lambda = 1.3, 0.05$, CoopCut (15 & 10 classes): $(\lambda, 10^4\vartheta) = (12, 3), (0.4, 7)$.

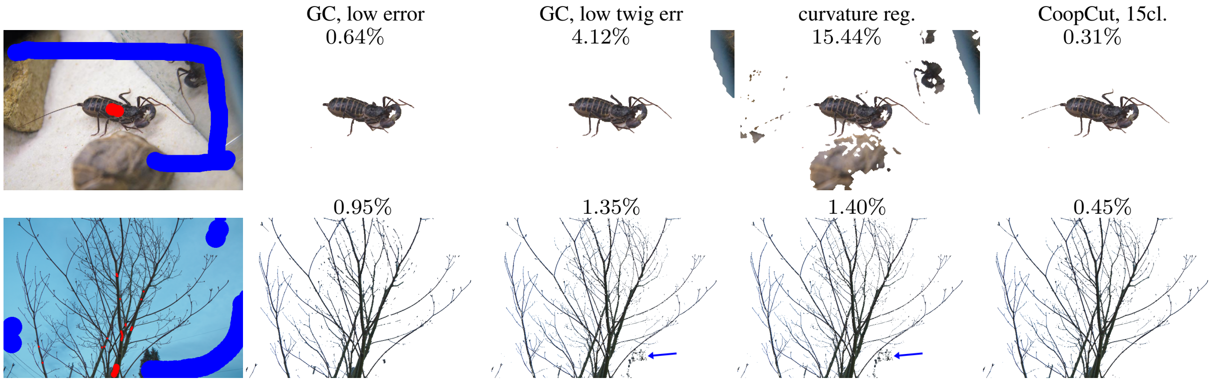


Figure 5. Example results for Experiment 2. Cooperation preserves legs and fine twigs without including pieces of background (arrows). Parameters: GC low err $\lambda = 1.5, 0.05$, GC low $\mathrm{err}_{\mathrm{twig}}$ $\lambda = 1.0, 0.001$; curv. $\lambda = 0.03, 0.002$, CoopCut: $(\lambda, 10^4\vartheta) = (1.5, 9), (1.8, 10)$.

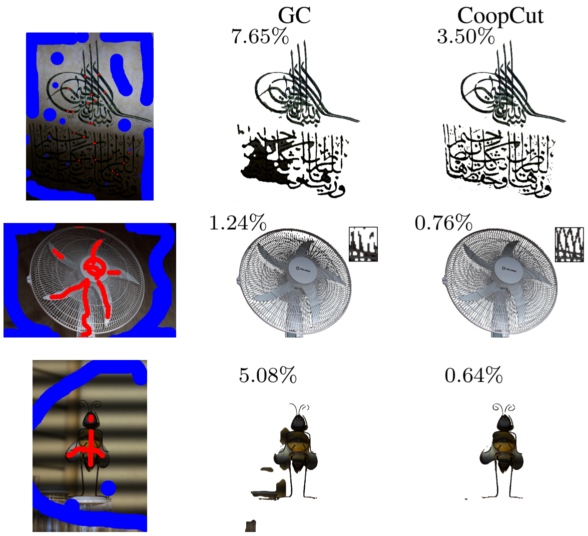


Figure 4. Results on shaded color images for 15, 20 and 25 classes (top to bottom). Parameters chosen for low total and twig error; GC: $\lambda = 0.1, 0.05, 0.1$; CoopCut: $(\lambda, 10^4\vartheta) = (4.5, 6), (7.0, 3), (1.5, 50)$. The zoom-in shows a part of the grid.

	GMM	hist.
GC	5.33 ± 3.7	6.88 ± 5.0
CoopCut ϕ_l, 20 cl.	4.95 ± 3.2	6.25 ± 4.3
CoopCut ϕ_r, 20 cl.	4.79 ± 3.1	6.12 ± 4.0

Table 3. Errors on the Grabcut data with both feature types.

9. Discussion

We introduced a general model, cooperative cuts, that can express a family of global potentials and reward co-occurrences, while still being approximable efficiently. We demonstrated its effectiveness for image segmentation, where we reward the co-occurrence of local boundary features. Key to this is a new class-structured cooperation that drives to globally cut *similar* edges, instead of merely *few* edges. Our approach can thus be viewed as a discrete structured sparsity. Furthermore, it can be extended to multiple labels. Swap or expansion moves then become cooperative cuts. Finally, the relations to other recent models imply that segmentation is only one possible application of the rich modeling capabilities of cooperative cuts.

Acknowledgments. We thank Sebastian Nowozin, Peter Gehler and Christoph Lampert for comments, and Richard

	(a) shading graysc.		(b) shading color		(c) high-frequ. shading (GMM)				(d) Expt. 2 (GMM)			
	GMM	hist	GMM	hist	(1) tot	twig	(2) tot	twig	(1) tot	twig	(2) tot	twig
unary terms	15.66	17.42	4.42	8.18	5.50	14.55	5.50	14.55	5.73	15.47	5.73	15.47
GC	14.03	14.71	3.41	6.49	2.56	20.96	3.43	13.54	2.10	34.40	3.78	18.08
GC,log weights	13.67	14.13	3.63	6.54	2.58	23.21	4.11	13.52	n/a	n/a	n/a	n/a
CoopCut,1	11.58	10.61	2.95	5.31	1.49	33.03	3.10	12.53	1.25	34.35	4.73	15.60
CoopCut,10	4.39	5.02	1.67	3.05	**1.26**	**14.79**	1.65	12.47	1.01	18.27	1.17	16.43
CoopCut,15	**3.63**	**4.27**	1.69	**2.94**	1.27	14.69	1.73	12.39	1.01	26.32	1.02	16.36
CoopCut,20	4.33	4.48	**1.62**	3.00	1.29	18.10	**1.62**	**12.01**	**0.98**	**17.78**	**1.16**	**15.91**
curvature reg.	17.40	19.48	3.93	7.37	3.38	34.50	4.70	14.08	3.82	56.09	5.73	16.00

Table 2. Average error (percent mispredicted pixels) for Expt. 1 and 2. GC: Graph Cut, Coopcut: Cooperative Cut with 1-20 classes. (a), (b) total error across 8 and 7 images; (c), (d) total and twig error across 18 and 17 images, respectively; (c), (d) results for parameters with (1) minimum total error, and (2) minimum joint error ($2\text{err}_{\text{total}} + \text{err}_{\text{twig}}$). CoopCut achieves low total *and* twig error, whereas GC can only minimize one of those. Twig error is overall higher since it counts fewer pixels. Results with histogram unary terms are similar [16].

Karp for the name "cooperative cut."

References

[1] A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr. Interactive image segmentation using an adaptive GMMRF model. In *ECCV*, 2004.

[2] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *ICCV*, 2001.

[3] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE TPAMI*, 26(9):1124–1137, 2004.

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE TPAMI*, 23, 2001.

[5] P. Dagum and M. Luby. Approximating probabilistic inference in Bayesian belief networks is NP-hard. *Artificial Intelligence*, 60(1):141 – 153, 1993.

[6] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov. Fast approximate energy minimization with label costs. *IJCV*, 2011.

[7] N. El-Zehiry and L. Grady. Fast global optimization of curvature. In *CVPR*, 2010.

[8] D. Freedman and P. Drineas. Energy minimization via graph cuts: Settling what is possible. In *CVPR*, 2005.

[9] S. Fujishige. *Submodular Functions and Optimization*. Ann. of Discr. Math. Elsevier Science, 2nd edition, 2005.

[10] S. Fujishige and S. B. Patkar. Realization of set functions as cut functions of graphs and hypergraphs. *Discr. Math.*, 226:199–210, 2001.

[11] L. Grady. Random walks for image segmentation. *IEEE TPAMI*, 28(11), 2006.

[12] D. M. Greig, B. T. Porteous, and A. H. Seheult. Exact maximum a posteriori estimation for binary images. *J. R. Stat. Soc.*, 51(2), 1989.

[13] D. Hochbaum and V. Singh. An efficient algorithm for cosegmentation. In *ICCV*, 2009.

[14] H. Ishikawa. Exact optimization for Markov random fields with convex priors. *IEEE TPAMI*, 25(10):1333–1336, 2003.

[15] H. Ishikawa. Higher-order clique reduction in binary cut. In *CVPR*, 2009.

[16] S. Jegelka and J. Bilmes. Supplementary material. `ssli.ee.washington.edu/~jegelka/cc/supp.pdf`.

[17] S. Jegelka and J. Bilmes. Cooperative cuts: graph cuts with submodular edge weights. Technical Report TR-189, Max Planck Institute for Biological Cybernetics, 2010.

[18] P. Kohli and M. Kumar. Energy minimization for linear envelope MRFs. In *CVPR*, 2010.

[19] P. Kohli, M. P. Kumar, and P. Torr. P3 & beyond: solving energies with higher-order cliques. In *CVPR*, 2007.

[20] P. Kohli, L. Ladický, and P. Torr. Robust higher order potentials for enforcing label consistency. *Int. J. Comp. Vision*, 82(3):302–324, 2009.

[21] V. Kolmogorov, Y. Boykov, and C. Rother. Applications of parametric maxflow in computer vision. In *ICCV*, 2007.

[22] V. Kolmogorov and C. Rother. Minimizing nonsubmodular functions with graph cuts–a review. *IEEE TPAMI*, 29(7):1274–1279, 2007.

[23] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE TPAMI*, 26(2):147–159, 2004.

[24] N. Komodakis and N. Paragios. Beyond pairwise energies: efficient optimization for higher-order MRFs. In *CVPR*, 2009.

[25] L. Ladický, C. Russell, P. Kohli, and P. Torr. Graph cut based inference with co-occurrence statistics. In *ECCV*, 2010.

[26] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular functions - I. *Math. Program.*, 14:265–294, 1978.

[27] S. Nowozin and C. H. Lampert. Global connectivity potentials for random field models. In *CVPR*, 2009.

[28] C. Rother, V. Kolmogorov, and A. Blake. Grabcut – interactive foreground extraction using iterated graph cuts. In *SIGGRAPH*, 2004.

[29] T. Schoenemann, F. Kahl, and D. Cremers. Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In *ICCV*, 2009.

[30] L. S. Shapley. Cores of convex games. *Int. J. Game Theory*, 1(1):11–26, 1971.

[31] A. Sinop and L. Grady. A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm. In *ICCV*, 2007.

[32] S. Vicente, V. Kolmogorov, and C. Rother. Graph cut based image segmentation with connectivity priors. In *CVPR*, 2008.

[33] S. Živný and P. Jeavons. Classes of submodular constraints expressible by graph cuts. *Constraints*, 15:430–452, 2010.