Graph Partitioning by Spectral Rounding: Applications in Image Segmentation and Clustering *

David A. Tolliver
Robotics Institute,
Carnegie Mellon University,
Pittsburgh, PA 15213.
tolliver@ri.cmu.edu

Gary L. Miller
Computer Science Department,
Carnegie Mellon University,
Pittsburgh, PA 15213.
glmiller@cs.cmu.edu

Abstract

*We introduce a family of spectral partitioning methods. Edge separators of a graph are produced by iteratively reweighting the edges until the graph disconnects into the prescribed number of components. At each iteration a small number of eigenvectors with small eigenvalue are computed and used to determine the reweighting. In this way spectral rounding directly produces discrete solutions where as current spectral algorithms must map the continuous eigenvectors to discrete solutions by employing a heuristic geometric separator (*e.g. *k-means). We show that spectral rounding compares favorably to current spectral approximations on the Normalized Cut criterion (NCut). Results are given for natural image segmentation, medical image segmentation, and clustering. A practical version is shown to converge.*

1. Introduction

Several problems in machine perception and pattern recognition can be formulated as partitioning a graph. In general, most formulations of partition quality yield NP-hard optimization problems. This raises two important questions. First, does a particular optimization problem capture good partitions for the image segmentation domain, especially in light of the optimization being NP-hard and thus we may never know the true optimum anyway. Second, given that the optimal value is a good characterization are approximations quickly constructible and do they return good partitions?

One popular formulation, used in image processing and clustering, is the normalized cut (NCut) of a graph introduced by Shi and Malik [16]. The ideas contained therein were further explored by Ng *et al*. [14] and Yu and Shi [20] both of whom motivated multi-way partitioning algorithms. In part, our method was motivated by observations made in [14, 20]. Now, how does the NCut optimization problem fare against our two questions?

It is not difficult to construct image examples for which common image percepts do not correspond to the optimal NCut of the image (*e.g.* see Shental *et al*.'s example [15] and see Figure 3 for similar sub-optima). This is unsurprising, and an acknowledged attribute of all objective measures of cluster or partition quality (see Kleinberg [6] and Meilă [10] for treatment of this issue). But, for many images, as we shall show, there are segmentations with a smaller normalized cut value than in those generated by earlier methods that are at the same time more pleasing. For example, one of the main empirical advantages of spectral rounding technique seems to be that it is less likely to split the image in homogeneous regions, see Figure 2, while returning smaller NCut values. Thus good image segmentations are generated as graph partitions without reformulating the underlying combinatorial problem.

The two common paradigms for approximating such objective functions are 1) linear or semidefinite programming [9, 19, 2]. and 2) spectral methods [4, 16]. In this paper we introduce a spectral technique that empirically improves upon existing spectral algorithms for quotient cuts. Earlier spectral methods consisted of a two stage algorithm. In the first stage a small collection of, say k, eigenvectors with small eigenvalues are computed. While in the second stage these vectors are used to map the graph vertices into $\mathbb{R}^k$ and a geometric separator is then applied [3].

Our approach is to skip the geometric separator step by iteratively reweighting the graph in such a fashion that it eventually disconnects. At each iteration we will use the eigenvalues and eigenvectors of the reweighted graph to determine new edge weights. At first hand this may seem very inefficient, since the most expensive step in the two stage method is the eigenvector calculation. By using the eigen-

*This work was supported in part by the National Science Foundation under grants CCR-9902091, CCR-9706572, ACI 0086093, CCR-0085982 and CCR-0122581.

vector from the prior step as a starting point, for finding the new eigenvector, simple powering methods seem to work in only a small number of steps.

1.1. Problems and Results

Let $G = (V, E, w)$ be a graph on n vertices with connectivity E; we wish to find a partitioning of vertices such that some cut metric is minimized. An apparently equitable measure of the quality of a cut is the **normalized cut** (NCut). As NCut minimizes the ratio cut cost over the balance in total degree of the partitions. The normalized cut criterion is:

$$nc(G) = \underset{V_1,..,V_p}{argmin} \; : \; \frac{1}{p}\sum_{i=1}^{p} \frac{vol(V_i, V \setminus V_i)}{vol(V_i)} \tag{1.1}$$

where $vol(V_i)$ is the sum of edge weights associated with the vertices in V_i, $vol(V_i, V \setminus V_i)$ is the sum of the edge weights connecting V_i to remainder of the graph, and $V_i \cap V_j = \emptyset$. The combinatorial problem in Equation 1.1 can be stated as a Quadratically Constrained Quadratic Program (QCQP)[20]. This QCQP admits a straightforward eigenvector relaxation, stated as minimization of the Rayleigh Quotient over orthogonal functions.

1.2. Quality of the Spectral Bound

Spectral methods are so named because the second smallest eigenvalue λ_2 of the **normalized Laplacian** bounds the best cut obtained from a continuous vector. The associated eigenvectors provide a means of obtaining a discrete solution that satisfies the eigenvalue bound. The eigenvalue bound on the **isoperimetric number** of a graph $\Phi(G)$, see [13, 4], can be ported to the normalized 2-cut as $\frac{1}{4}\lambda_2 \leq nc(G) \leq \Phi(G) \leq \sqrt{2\lambda_2}$, as $\Phi(G)$ is an upper bound on $nc(G)$. The upper bound on $\Phi(G)$ is loose in general, as demonstrated by the pathological graphs constructed by Guattery and Miller in [5]. While, guaranteed $O(\frac{1}{\sqrt{n}})$ cut bounds were exhibited for planar graphs in [17].

Equation 1.1 has been studied in the context of image segmentation in the vision community [16, 20] and clustering in the learning community [14, 19]. In all cases a standard spectral algorithm is used. The methods [16, 20, 14] differ primarily in how the eigenvectors are used to find a feasible solution satisfying the constraints in Equation 1.1.

2. Preliminaries

Throughout this paper we let $G = (V, E, w)$ denote an edge weighted undirected graph without multiple edges or selfloops, where V is a set of n vertices numbered from 1 to n, E is a set of m edges, and $w : E \to [0, 1]$ is the edge weighting.

We associate four matrices with the graph: First, W the weighted adjacency matrix,

$$W_{ij} = \begin{cases} w_{ij} = w(i,j) & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

The **weighted degree** of vertex i is $d_i = \sum_{j=1}^{n} w_{ij}$. We assume that no vertex has zero degree. Second, the **weighted degree matrix** D is

$$D_{ij} = \begin{cases} d_i & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} . \tag{2.2}$$

Third, the **generalized Laplacian** or simply the **Laplacian** of G is $L = D - W$. Finally, the **normalized Laplacian** of G is $\mathcal{L} = D^{-1/2}LD^{-1/2}$.

Rather than working directly with the normalized Laplacian we shall work with a similar system. Throughout this paper we let f and g be functions from the vertices to the reals. Such functions we will call **vertex valuations** or simply a valuations of G. We shall also view them as n by 1 column vectors. When convenient we may also view them as eigenvectors of L or $\mathcal{L}$.

If $D^{1/2}f = g$ and g is an eigenvector of $\mathcal{L}$ with eigenvalue λ, i.e., $\mathcal{L}g = \lambda g$ then it is an easy calculation to see that f is a generalized eigenvector for the pair (L, D) with eigenvalue λ. That is $Lf = \lambda Df$. It will be convenient to work with the generalized eigenvalues and vectors of (L, D). In this case the normalized Rayleigh quotient is $f^T Lf / f^T Df$ of the valuation f.

We make a simple, but important, observation about these Rayleigh quotients:

Lemma 1. *Given a weighted symmetric graph $G = (V, E, w)$ then the normalized Rayleigh quotient can be written as*

$$\frac{f^T Lf}{f^T Df} = \frac{\sum_{(i,j)\in E, i<j} (f_i - f_j)^2 w_{ij}}{\sum_{(i,j)\in E, i<j} ((f_i)^2 + (f_j)^2) w_{ij}} \tag{2.3}$$

where $f_i = f(v_i)$

The main importance of Lemma 1 is that for each valuation f and each edge e_{ij} we get the fraction $\frac{(f_i - f_j)^2}{(f_i)^2 + (f_j)^2}$. We will use this fraction to reweight the edge e_{ij}. The simplest reweighting scheme would be to replace the edge weight w_{ij} with the weight $\frac{(f_i)^2 + (f_j)^2}{(f_i - f_j)^2} w_{ij}$. There are several issues with this scheme that will be address in the next section.

3. Spectral Rounding

In this section spectral rounding is introduced as a procedure for obtaining graph cuts. At each iteration a small number of eigenvectors with small eigenvalue are computed and used to determine the reweighting w' for the graph

$G = (V, E, w)$. We show this process induces a k-way multiplicity in the k smallest eigenvalues of L (*i.e.* $\lambda_i(L) = 0$ for $1 \leq i \leq k$). By obtaining a Laplacian with this nullspace property we guarantee that the matrix represents k disconnected subgraphs, whose vertex membership can be read off directly from the first k eigenvectors.

§3.1 defines the spectral rounding algorithm. §3.2 connects decreasing a Rayleigh quotient to reweighting the graph. §3.6 the spectral rounding algorithm is shown to converge for a **reweighting scheme** proposed in §3.1.

3.1. The `SR-Algorithm`

For a graph $G = (V, E, w^0)$ prescribe the number of partitions k that the edge cut is to yield. Given a valid **reweighting scheme**, iteration of the `SR-Step` produces a sequence of N weightings $\{w^{(N)}\}$ such that the graph $G^N = (V, E, w^N)$ is disconnected into k components by the weighting w^N.

Algorithm 1 `SR-Step(w:` $||w||_k > 0$)
Let $F_k = [f_1 \ldots f_k]$ *denote the* k *generalized eigenvectors of* $L(G; w), D(G; w)$ *associated with the* k *smallest eigenvalues* $\Lambda_k = diag([\lambda_1 \ldots \lambda_k])$

1. `compute` $w_r = \mathcal{R}(F_k, \Lambda_k)$, `set` $\alpha = 1$ & $w' = w_r$
2. `while` $||w'||_k \geq ||w||_k$
   $\alpha \leftarrow \frac{1}{2}\alpha$, $w' = (1 - \alpha)w + \alpha w_r$
3. `return` w'

The function $\mathcal{R}$ computes a new weighting of the graph given the first k eigenpairs of L, D. The norm $||\cdot||_k$ is taken over weightings of the graph, such that $||w||_k = 0$ *iff* the weighting w disconnects the graph into at least k pieces. A pair $\mathcal{R}$, $||\cdot||_k$ is called a **reweighting scheme** if the `SR-Step` converges in a finite number of iterations. We define Algorithm 2, the `SR-Algorithm`, as the iteration of Algorithm 1 until $||w^{(N)}||_k \approxeq 0$. In the following sections we propose $\mathcal{R}$s and corresponding norms $||\cdot||_k$ such that the `SR-Step` and `SR-Algorithm` converge in the desired fashion.

In §3.6 the `SR-Algorithm` is shown to converge on graphs with $||\cdot||_k < 1$. In the case of a 2-cut this reduces to $\lambda_2(\mathcal{L}) < 1$. The class of graphs satisfying this spectral constraint is very general, excluding an uninteresting collection of graphs for our purposes. In particular, if $\lambda_2 \geq 1$ then no subset of the vertices exists with more than half of its edge volume contained within it (entailed by the Cheeger bound $2\Phi(G) \geq \lambda_2$ [4]). Such graphs are often called expander graphs.

3.2. Fractional Averages: a reweighting function

By Lemma 1 we saw that the Rayleigh quotient could be written as a sum of formal fractions where the numerators are added separately from the denominators. Define a **formal fraction** as a pair of real numbers $\frac{a}{b}$ and its **value** as the real number a/b. We call the average of a set of **formal fractions** the **fractional average**. We now prove a few simple but important facts about fractional averages.

Definition 1. *Given formal fractions*

$$\frac{a_1}{b_1}, \cdots, \frac{a_n}{b_n}$$

the **fractional average** *is the formal fraction*

$$\frac{\sum_{i=1}^n a_i}{\sum_{i=1}^n b_i}$$

where the a_i*'s and* b_i*'s are reals.*

We will simply call formal fractions fractions and only make a distinction between the formal fraction and its value when needed. In the case when the a_i's and b_i's are nonnegative we first observer that the fractional average is a convex combination of the fractions. That is we can rewrite the sum as

$$\sum_{i=1}^n \frac{b_i}{\bar{b}} \cdot \frac{a_i}{b_i}$$

where $\bar{b} = \sum_{i=1}^n b_i$. Thus fractional average lies between the largest and smallest fraction.

Possibly a more important interpretation is by viewing each fraction $\frac{a_i}{b_i}$ as the point $P_i = (b_i, a_i)$ in the plane and the value of the fraction is just its slope. The fractional average is just the vector sum of the points. Since we are only interested in the value of the fraction, the slope, we will think of the fractional average as the centroid of the points. If we multiply the numerator and denominator by a scalar w we shall say we **reweighted the fraction** by w. Geometrically, we are scaling the vectors or points P_i and then computing the centroid.

In the next lemma we show that we can control the slope of the fractional average by reweighting.

Lemma 2. *If* $\frac{a_1}{b_1} \leq \cdots \leq \frac{a_n}{b_n}$ *and* $w_1 \geq \cdots \geq w_n$ *then*

$$\frac{\sum_{i=1}^n a_i}{\sum_{i=1}^n b_i} \geq \frac{\sum_{i=1}^n a_i w_i}{\sum_{i=1}^n b_i w_i}$$

The inequality is strict if for some pair $1 \geq i < j \leq n$ *we have that* $\frac{a_i}{b_i} < \frac{a_j}{b_j}$ *and* $w_i > w_j$.

See [12] for the proof of Lemma 2.

The Rayleigh quotient in Lemma 1 associates a formal fraction with each edge in G. One of the simplest ways to get weights satisfying the hypothesis of Lemma 2, for such a system, is to pick $w_i = \frac{b_i}{a_i} = \frac{f(u_i)^2 + f(v_i)^2}{(f(u_i) - f(v_i))^2}$, if a_i is not zero. We shall call this **inverse fractional reweighting**.

This reweighting scheme gives very large values for small values of a_i. We have found that using the stereographic map to normalized the inverse fractions between zero and one works well.

Observation 1. *The stereographic projection $\Psi : \mathbb{R}^d \to \mathcal{S}^d$ preserves the order of points on the real line, mapping points at ∞ to 1 and points at 0 to 0. Thus the* **inverse weight ordering** *of the edge update values is preserved by the stereographic map.*

If we think of the Φ as mapping points in $\mathbb{R}^d$ to $\mathbb{R}^{d+1}$, where we are only interested in the value in the $d+1$ dimension, then the images of $v \in \mathbb{R}^d$ is $\frac{v^T v}{v^T v+1} \geq 0$. We use Ψ_h to denote the map which returns the value in this dimension (*i.e.* the "height" on the sphere).

3.3. Reweighting for Multiple Valuations (`step 1`)

In the discussion so far we have assumed only one valuation is being used. To produce cuts of higher degree (*i.e.* $k > 2$) it is crucial that we simultaneously handle multiple valuations in computing the reweighting. Given two valuations, say f_2 and f_3, we need to pick a single new weight per edge, *i.e.* a reweighting function $\mathcal{R}$. We suggest two approaches.

The first, $\mathcal{R}_{fa}$, takes the fractional sum of f_2 and f_3 per edge, yielding one fraction per edge. The second, $\mathcal{R}_{sa}$, applies the stereographic map to the vector of slopes of the individual Rayleigh quotients for f_2 and f_3. Formally, for an edge (uv), let $a_i(u,v) = \frac{1}{\lambda_i}(f_i(u) - f_i(v))^2$ and $b_i(u,v) = f_i(u)^2 + f_i(v)^2$. Define the reweighting functions as

$$\mathcal{R}_{fa}(F, \Lambda, w_{uv}) = \Psi_h\left(\frac{\sum_{i=1}^k b_i(u,v)}{\sum_{i=1}^k a_i(u,v)}\right) w_{uv} \quad (3.1)$$

$$\mathcal{R}_{sa}(F, \Lambda, w_{uv}) = \Psi_h\left(\left[\frac{b_1(u,v)}{a_1(u,v)} \ldots \frac{b_k(u,v)}{a_k(u,v)}\right]\right) w_{uv}$$

To successfully link the reweighting functions $\mathcal{R}$ to eigenvalue optimization we must specify a norm function $||\cdot||_k$ on weightings of G. For example, given the function $\mathcal{R}_{fa}$ we specify the norm $||w'||_k = \frac{\sum_{i=1}^k \sum_{(uv)\in E} w_{uv}(f_i'(u)-f'(v))^2}{\sum_{i=1}^k \sum_{(uv)\in E} w_{uv}(f_i'(u)^2+f_i'(v)^2)}$, *i.e.* the fractional average of the updated eigenvectors $f_i' \mid L' f_i' = \lambda_i' D' f_i'$. Given the reweighting function $\mathcal{R}_{sa}$ the norm $||w'||_k$ is the sum of k smallest eigenvalues of L', D'. We now connect linear combinations of weightings to their eigenvalues.

3.4. From Rayleigh Quotients to Eigenvalues

In §3.2 we showed how to, given a valuation or set of valuations of a graph, reweight the edges so as to reduce the Rayleigh quotient. In general this does not mean that if the valuation f is an eigenvector with eigenvalue λ of the old graph that the corresponding eigenpair f' and λ' of the new graph will have the property that $\lambda' \leq \lambda$.

Given a new edge weighting w' such that the fractional average of an eigenvector is decreased, we show that there is a linear combination of the weights of the form $w + t \cdot w'$ for $t > 0$ such that the associated eigenvalue is also decreased. This yields an algorithm which forces the target eigenvalue to zero. And motivates a matrix where the entries are linear functions of a scalar t and the following lemma.

Definition 2. *Given two weightings w and w' of G we define the* **matrix curve***, a $1-$dimensional family of matrices, as:*

$$W(t) = W + tW'$$

for $t \geq 0$.

A direct consequence of the scale invariance of the Rayleigh quotient $\frac{f^T L f}{f^T D f}$ is that any linear combination $W(t) = W + tW'$ may be expressed as a convex combination $W(\alpha) = (1-\alpha)W + \alpha W'$ on $0 \leq \alpha \leq 1$ (*i.e.* $\alpha = \frac{t}{t+1}$). The eigenstructure of normalized laplacians defined on $W(\alpha)$ and $W(t)$ are identical by the scale invariance of the Rayleigh quotient.

Lemma 3. *Given a weighted graph $G = (V, E, w)$, matrices L and D, the simple eigenpair $(f, \lambda) \mid Lf = \lambda Df$, and a new weighting w' such that $\frac{f^T L' f}{f^T D' f} < \frac{f^T L f}{f D f} = \lambda$ then the derivative of the eigenvalue function*[1] *$\lambda(t)$ of the matrix curve $W(t) = W + tW'$ is well defined for small t and*

$$\frac{d\lambda(t)}{dt} < 0$$

at $t = 0$.

See [12] for a detailed proof.

To handle multiple eigenvectors one hope might be to simultaneously bound the derivatives of the target eigenvalues $\{\lambda_2, ..., \lambda_k\}$ of $L(t), D(t)$. To do this one arrives at the update criteria for the re-weighting w'; select a w' such that $\frac{f_i^T L' f_i}{f_i^T D' f_i} < \frac{f_i^T L f_i}{f_i^T D f_i}$ insuring that $\lambda_i(L(t)) < \lambda_i(L)$ for $2 \leq i \leq k$. But in general this may be unrealistic since we must use the same weight per edge for all the eigenvectors f_i. In the case where the aforementioned inequalities do not hold, we determine w' so as to decrease the fractional average of the Rayleigh quotients (see $\mathcal{R}_{fa}$ in §3.3). The average of the target eigenvalues tends to zero as a consequence of the decrease in the fractional average.

[1] The proof that $d\lambda$ exists follows from properties of the characteristic polynomial of L, D and relies on the implicit function theorem. Details can be found on the differentiability of λ and f in Lax [8] chapter 9.

3.5. Termination of SR-Step

In order to prove that the algorithm SR-Algorithm converges to a k-way partition we first need to show that each step of SR-step terminates. In the next subsection we use this termination to show convergence. To simplify the discussion we only consider using one eigenvector. The SR-Algorithm has two majors steps. In the first step given a valuation f it computes a reweighting w_r. We claim with a reasonable assumption about the Rayleigh quotient that

$$\frac{f^T L f}{f^T D f} > \frac{f^T L_r f}{f^T D_r f}. \tag{3.2}$$

By lemma 2 we know that equation 3.2 is true as long as not all the fractions $\delta_{ij} = (f_i - f_j)^2/(f_i^2 + f_j^2)$ are equal. We show that if the fraction are all equal the Rayleigh quotient is at least one, contradicting how we construct affinity graphs.

Lemma 4. *Let $G = (V, E, w)$ be a connected graph and f a valuation such that all the fractions $\delta_{ij} = (f_i - f_j)^2/(f_i^2 + f_j^2)$ are equal then $\frac{f^T L f}{f^T D f} \geq 1$*

Proof. Let f be a valuation of G with Rayleigh quotient λ. Suppose all $\delta_{ij} = (f_i - f_j)^2/(f_i^2 + f_j^2)$ over all edges of G are the same value. Observe that: 1) $\delta_{ij} > 1$ if the sign of f_i and f_j differ, 2) $\delta_{ij} = 1$ if $f_i f_j = 0$, 3) $\delta_{ij} < 1$ if $f_i f_j > 0$. If we are not case 1) then the value of each fraction is at least one. Thus by Lemma 1 the Rayleigh quotient is at least one. □

The proof of termination for multiple eigenvectors is similar.

Lemma 5. *Algorithm SR-step terminates for graphs such that $\lambda_2 < 1$*

Proof. Follows from Lemma 3 and Lemma 4. □

3.6. Convergence of SR-Algorithm

By Lemma 5 we know that each step of procedure SR-step produces a new weighted graph such that the norm $||\ ||_k$ is strictly decreasing. We show that in fact the limit norm must be zero, i.e., SR-Algorithm converges.

Again for simplicity of the exposition we only consider the case of a single eigenvector. Let G^i, D^i, λ^i, and x^i be the ith graph, its degree matrix, its second eigenvalue, and eigenvector. That is $G^i x^i = \lambda^i D^i x^i$. We also assume that each x^i is a unit length vector. Thus all the eigenvectors all belong to a compact region in $\mathbb{R}^n$, the unit sphere. The Bolzano-Weierstrass theorem dictates such a space contains an accumulation point, say, x. Let $\{x^{(i)}\}$ be a subsequence of eigenvectors that converge to x, and let G', D', and λ' be the corresponding graph, degree matrix and eigenvalue. The eigenvectors converge to x and the eigenvalues converge to some value λ. To insure that the graph also converges to a unique graph we apply the Bolzano-Weierstrass theorem again to the weighted graphs which belong to a compact set in $\mathbb{R}^m$. Thus, we may also assume the graphs and degree matrix converge to G', D', and λ' such that $G'x' = \lambda' D'x'$.

Lemma 6. *The limit of the sequence of λ_i, as defined above, converges to $\lambda = 0$*

Proof. Suppose that $\lambda' > 0$. We know by Lemma 5 that if we run SR-step on G' we will get a new graph with $\lambda'' < \lambda'$. Let $\epsilon = \lambda' - \lambda''$. We can also run SR-Step on each G'_i getting a new eigenvalue λ''_i. Let $\epsilon_i = \lambda'_i - \lambda''_i$. Since SR-step is a continuous function in G'_i and x'_i we get that the ϵ_i converge to ϵ. For sufficiently large i it follows that $\epsilon_i \geq \epsilon/2$. But this implies that λ'_i goes to $-\infty$ which contradicts the fact the they a bounded below by zero. □

4. Empirical Evaluation

We show that spectral rounding compares favorably to recent spectral approximations for the Normalized Cut criterion (NCut). For the purpose of our evaluation we compared our method, denoted *SR*, to that of Yu and Shi [20], denoted *EIG*, as it reliably returns superior cuts relative to methods which use K-means and less of the eigenvectors, those based on [14]. In the following sections results are given in geometric clustering, natural image segmentation, and medical image segmentation. In all cases, the same graph $G = (V, E, w)$, is passed to both algorithms. To compare the partitioning we report the expected improvement in NCut value, on the initial graph $G = (V, E, w)$, and the **variation of information** between clusterings.

The first comparison, the expected improvement in value, can be expressed as a positive scaler $\mathbf{c}$ such that $nc(SR) \approxeq \mathbf{c} \cdot nc(EIG)$ on average. In the case where $\mathbf{c} < 1$ *spectral rounding* finds an improved cut on average.

Throughout this section we report the **variation of information** described in [11] to measure the difference between two partitionings, P_a and P_b of the vertex set V, defined as

$$\mathcal{D}_{vi}(P_a, P_b) = H(P_a) + H(P_b) - 2I(P_a, P_b) \tag{4.1}$$

with $I(P_a, P_b) = -\sum_{i=1}^{p}\sum_{j=1}^{p} \frac{n_{i,j}}{n} \log \frac{n_{i,j}}{n}$, $H(P) = -\sum_{k=1}^{p} \frac{n_k}{n} \log \frac{n_k}{n}$, where n_k is number of vertices in the k^{th} partition, $n_{i,j}$ the size of the overlap between the i^{th} partition in P_a and j^{th} partition in P_b, and $n = |V|$. The entropy term $H(P)$ can be thought of as the number of bits required to represent the distribution of partition sizes. Where as $I(P_a, P_b)$ is the mutual information between the two partitionings P_a and P_b. And so, $\mathcal{D}_{vi}$ can be thought of as the number of bits needed to represent the cumulative disagreement between the overlapping partitions of P_a and P_b. As expected $\mathcal{D}_{vi}(P_a, P_a) = 0$.

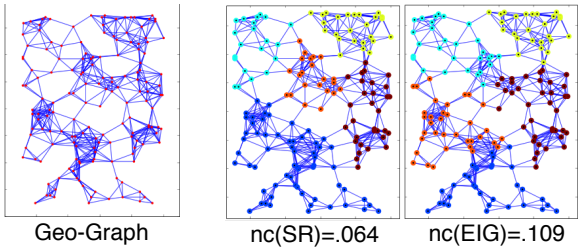

Figure 1. A ($|V| = 300$) geometric graph, and two 5-way cuts.

4.1. Random Geometric Graphs

We compare *SR* and *EIG* in the expected partition cost on a collection of random geometric graphs. The vertices of $G = (V, E, w)$ are associated with uniformly distributed coordinates in $\mathbb{R}^d$. The edge set of $E(G)$ is then constructed using the following rule, for $\{u, v \in V | u \neq v\}$, $(u, v) \in E \iff dist(u, v) < r$. We sampled 10000 graphs with 1000 vertices and chose the radius r such that the expected degree of each vertex was approximately $log(|V|)$. As shown in Figure 1 such graphs afford a large number of relatively small cuts. Table 1 contains the improvement factor, and the cluster divergence. We note that the divergence distance, relative to partition entropy $H(SR)$, highlights that the NCut improvements are not due to a small number of boundary vertex exchanges, but rather that SR and EIG return significantly different subgraphs.

	$\mathcal{D}_{vi}(SR, Eig)$	$nc(SR) = \mathbf{c} \cdot nc(EIG)$
geo-graph	$0.910 \pm .219$	$\mathbf{c} = .690 \pm .113$

Table 1. Comparison between spectral rounding *SR* and the multi-way cut algorithm of Yu and Shi [20] *EIG*. The partition entropy for SR was $H(SR) \approxeq 1.935$.

4.2. Image Segmentation

The parameters used in constructing a weighted graph from an image were fixed for all the results presented in this section. The graph $G = (V, E, w)$ represents an image as follows. For each pixel in the image a vertex in V is assigned. If two pixels are connected in E a weight in w is determined based on the image data. The graph connectivity, E, was generated by connecting pixels to 15% of their neighboring pixels in a 10 pixel radius. The initial weighting w of the graph $G = (V, E, w)$ was determined using the *Intervening Contour* cue described in [16]. This cue assigns small weights to pixels which lie on opposite sides of a strong image boundary, and large weights otherwise.

4.2.1 Natural Image Segmentation

We compiled a set of a 100 images from Google Images using the keywords *farm, sports, flowers, mountains, & pets*. Examples from this data set, and segmentations can be found in Figure 2. Again, we note that changes in the cut value often correlate with large changes in the co-membership relationships on the image pixels. To quantitatively compare the methods on natural images we report the divergence distance and NCut improvement factor **c**.

	$\mathcal{D}_{vi}(SR, Eig)$	$nc(SR) = \mathbf{c} \cdot nc(EIG)$
natural	$1.23 \pm .160$	$\mathbf{c} = .536 \pm .201$

Table 2. Comparison between spectral rounding *SR* and the multi-way cut algorithm of Yu and Shi [20] *EIG* on segmentations of natural images. The average cluster entropy over *SR*-segmentations of the image collection is $1.62 \pm .4$.

4.2.2 Medical Image Segmentation

To a degree, clustering methods are only successful in that are useful in servicing a particular task. We have selected a simple medical task, segmenting out the left ventricle (a fluid sack located in the brain), as it is well defined – *i.e.* the boundary of the percept is agreed upon by experts. While this task would seem to be relatively easy, a successful automatic method represents a significant reduction in human effort for a common labeling task.

The test set was constructed from a collection of 200 NMR images containing the left ventricle. The collection was built by taking 5 slices each from scans of 40 individuals. Images were selected randomly from slices containing the left ventricle. As shown in Figure 5 the appearance of the ventricle varies substantially in shape and size.

The comparison of segmentations obtained from spectral rounding and the eigenvector method of Yu and Shi [20] with the human labels is given in Table 3. The divergence distance and expected cut improvement are given in Table 4. The average cluster entropy for *SR* was $0.611 \pm .131$. As this is a two-class problem, this suggests that one of the segments tends to be much smaller than the other. This is due to the often small size of the ventricle in the image.

	$nc(SR)$	$nc(EIG)$[20]
$Pr(v \in T(Im))$	$.95 \pm .04$	$.37 \pm .12$

Table 3. The value $Pr(v \in T(Im))$ is reported over the population of images, where $T(Im)$ is the expert's hand segmentation and $Pr(v \in T(Im))$ is the probability that a pixel v in a segment is also contained in $T(Im)$ – this statistic was computed for the segment with the largest overlap with $T(Im)$.

	$\mathcal{D}_{vi}(SR, Eig)$	$nc(SR) = \mathbf{c} \cdot nc(EIG)$
medical	$1.856 \pm .192$	$\mathbf{c} = .598 \pm .237$

Table 4. The divergence and expected value improvement for the medical image data set. The average cluster entropy for *SR* segmentations on the medical data set was $0.611 \pm .131$.

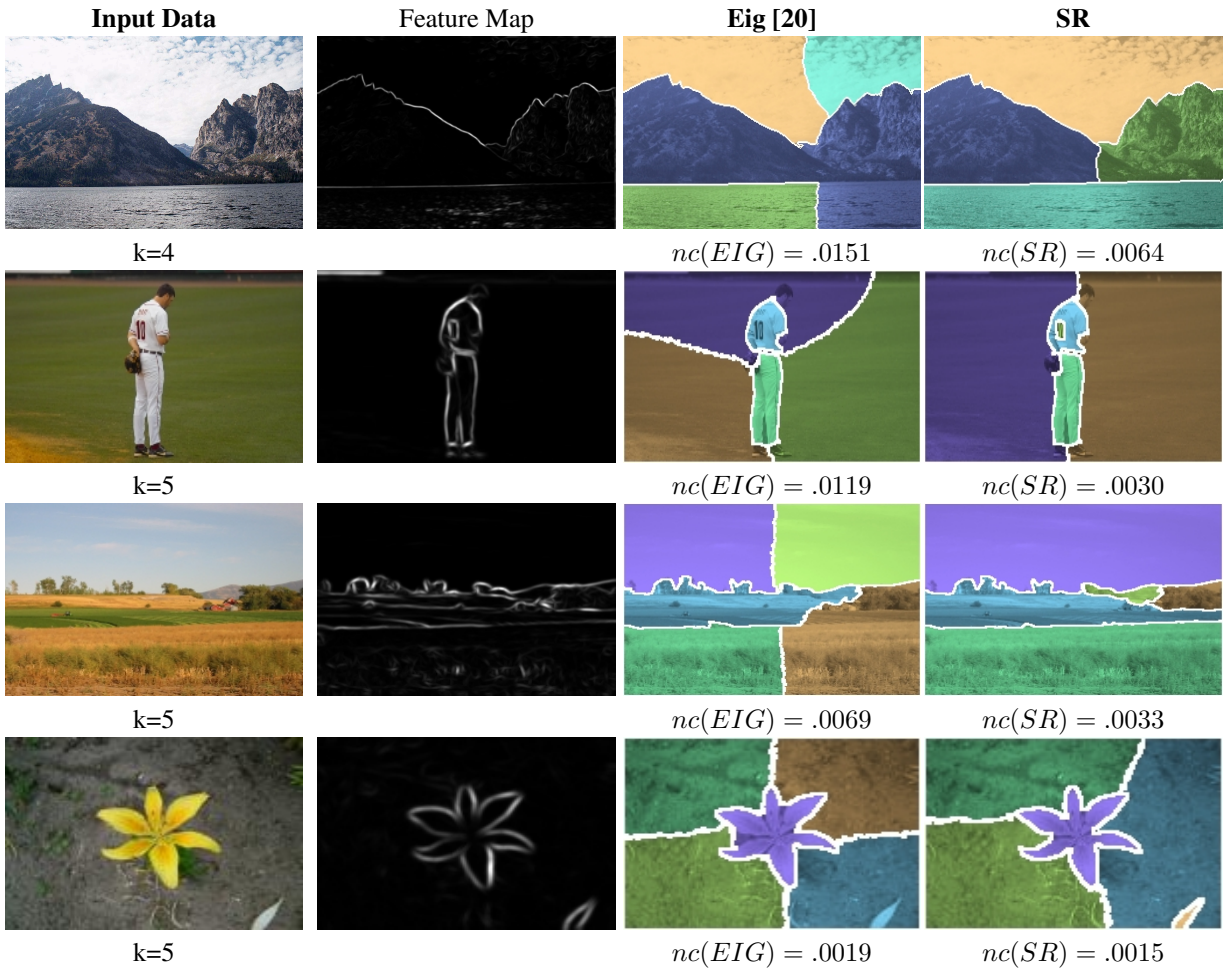

Figure 2. The first four rows illustrate with qualitative examples the improvements in NCut value for natural images. Column three contains segmentations generated by the published code of Yu and Shi [20]. Column four contains results for *Spectral Rounding*. The number of segments k is fixed for each comparison. We emphasize that the cut cost was evaluated on identical combinatorial problems (graphs).

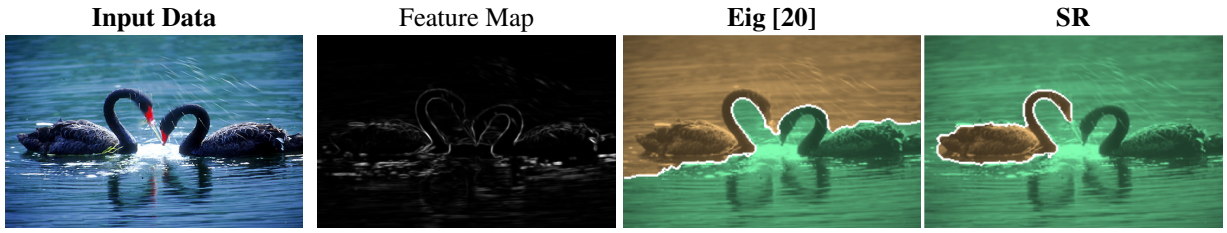

Figure 3. The second to last row illustrates a 2-way cut in which the NCut values are nearly identical, but which support very different percepts.

5. Discussion

We have presented a new rounding technique for the eigenvectors of graph laplacians in order to partition a graph. The method was shown to converge and demonstrated empirical improvements over the standard rounding strategy on a variety of problems. In ongoing work we are seeking a theoretical bound for the SR-algorithm.

The initial eigencomputation remains as a hurdle to the integration of spectral methods into large scale vision and data-mining systems. At present the best known [7, 1] average case time bound on $\epsilon-$approximate estimation of an extremal eigenpair is $\tilde{O}(m\sqrt{\frac{n}{\epsilon}})$, for generalized Laplacians

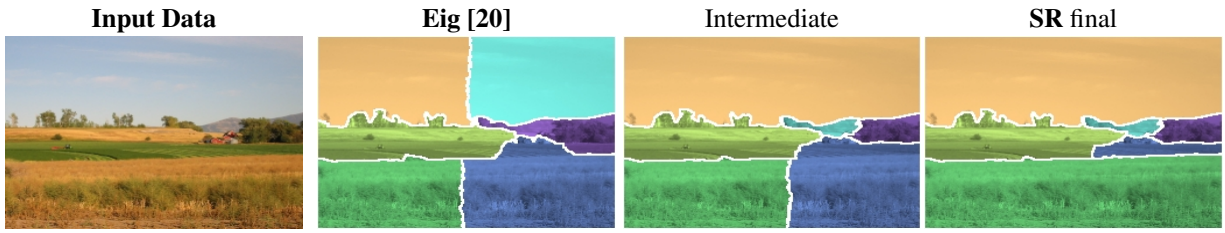

Figure 4. A sequence of iterations projected onto the feasible set, starting left with solution from Yu's method and ending with the fourth and final *SR* iteration on the right. Notice that the large cuts in the sky and field shift to small cuts in the area around the farm.

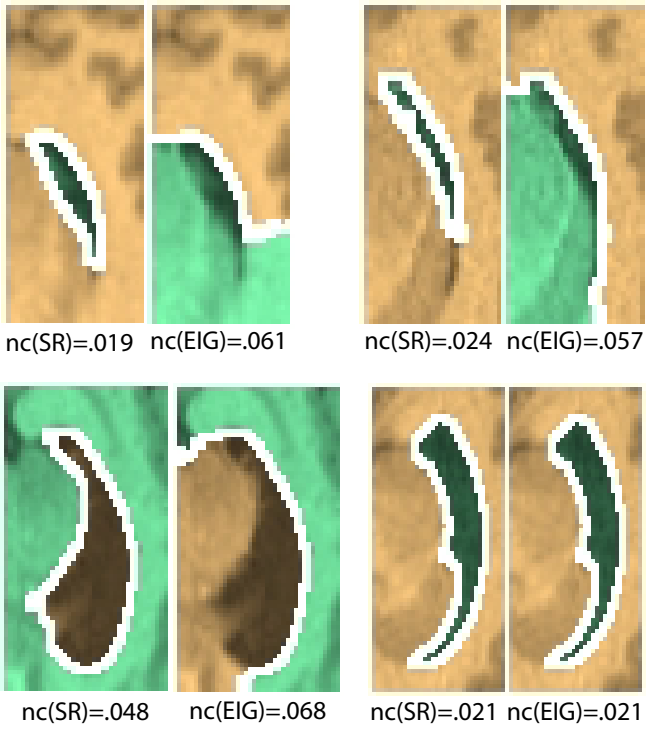

Figure 5. Examples of the left ventricle, and qualitative results for the *SR* and *EIG* algorithms. Segmentations required approximately 1.2 seconds for *EIG* and 1.9 seconds for *SR*.

with $n = |V|$ and $m = |E|$. Fortunately, recent theoretical results hold out the promise of nearly-linear time eigencomputation. Recent work by Spielman and Teng [18] on linear time linear solvers suggests that $\epsilon-$approximate eigencomputation may soon fall into same time complexity.

References

[1] S. Arora, E. Hazan, and S. Kale. Fast algorithms for approximate semidefinite programming using the multiplicative weights update method. *FOCS*, pages 339–348, 2005.

[2] S. Arora, S. Rao, and U. V. Vazirani. Expander flows, geometric embeddings and graph partitioning. In *STOC*, pages 222–231, 2004.

[3] T. F. Chan, J. R. Gilbert, and S.-H. Teng. Geometric spectral bisection. Manuscript, July 1994.

[4] F. R. K. Chung. *Spectral graph theory*. Number 92 in Regional Conference Series in Mathematics. Amer. Math. Soc, Providence, 1997.

[5] S. Guattery and G. L. Miller. On the quality of spectral separators. *Matrix Analysis & Applications*, 19(3), 1998.

[6] J. Kleinberg. An impossibility theorem for clustering. *Advances in Neural Information Processing Systems*, 2003.

[7] J. Kuczyński and H. Woźniakowski. Estimating the largest eigenvalue by the power and lanczos algorithms with a random start. *Matrix Analysis & Applications*, 4(13):1094–1122, 1992.

[8] P. D. Lax. *Linear Algebra*. Pure and Applied Mathematics. Wiley-Interscience, 1st edition, 1997.

[9] F. T. Leighton and S. Rao. An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms. In *FOCS*, pages 422–431. IEEE, Oct. 1988.

[10] M. Meilaă. Comparing clusterings – an axiomatic view. *ICML*, pages 577–584, 2005.

[11] M. Meilă. Comparing clusterings by the variation of information. *COLT*, pages 173–187, 2003.

[12] G. L. Miller and D. A. Tolliver. Graph partitioning by spectral rounding. *SCS Technical Report*, May 2006.

[13] B. Mohar. Isoperimetric numbers of graphs. *Journal of Combinatorial Theory, Series B*, 47:274–291, 1989.

[14] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, 2002.

[15] N. Shental, A. Zomet, T. Hertz, and Y. Weiss. Learning and inferring image segmentations with the gbp typical cut algorithm. *ICCV*, 2003.

[16] J. Shi and J. Malik. Normalized cuts and image segmentation. In *PAMI*, 2000.

[17] D. Spielman and S. Teng. Spectral partitioning works: Planar graphs and finite element meshes. *FOCS*, 1996.

[18] D. Spielman and S. Teng. Nearly linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. *STOC*, pages 81–90, 2004.

[19] E. P. Xing and M. I. Jordan. On semidefinte relaxtion for normalized k-cut and connections to spectral clustering. *TR-CSD-03-1265*, University of California Berkeley, 2003.

[20] S. Yu and J. Shi. Multiclass spectral clustering. In *ICCV*, October 2003.