Spectral Methods in Image Segmentation: A Combined Approach

Fernando C. Monteiro[1,2] and Aurélio C. Campilho[1,3]

[1] INEB - Instituto de Engenharia Biomédica
[2] Escola Superior de Tecnologia e de Gestão de Bragança
Campus de Santa Apolónia, Apartado 134, 5301-857 Bragança, Portugal
monteiro@ipb.pt
[3] FEUP - Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
campilho@fe.up.pt

Abstract. Grouping and segmentation of images remains a challenging problem in computer vision. Recently, a number of authors have demonstrated a good performance on this task using spectral methods that are based on the eigensolution of a similarity matrix. In this paper, we implement a variation of the existing methods that combines aspects from several of the best-known eigenvector segmentation algorithms to produce a discrete optimal solution of the relaxed continuous eigensolution.

1 Introduction

The natural ability of the human visual system to separate an image into coherent segments or groups is extraordinary. This important phenomenon was studied extensively by the Gestalt psychologists, nearly a century ago [10]. They identified several key factors that contribute to human perceptual grouping process, including cues such as proximity, similarity, symmetry, continuity, common fate and familiarity.

An auspicious approach that has recently emerged uses spectral methods for image segmentation. These methods use the eigenvectors of a matrix representation of a graph to partition image into disjoint clusters with pixels in the same cluster having high similarity and points in different clusters having low similarity. A common characteristic among these techniques is the idea of clustering/separating pixels or other image elements using the dominant eigenvectors of a $n \times n$ matrix derived from the pair-wise affinities between pixels, where n denotes the number of pixels in the image. The affinity computed between pixels captures their degree of similarity as measured by one or more cues.

The general belief that these methods work is based on proofs that if segments are very dissimilar, spectral methods will be able to separate them [5]. In addition to, there is accumulated evidence that spectral methods find good or acceptable segmentation as judged by human on a variety of real data sets [3], i.e. these methods are effective in capturing perceptual organization features [2]. In spite of these facts, different authors still disagree on exactly which matrix and which eigenvectors they should use and how to proceed from the continuous eigenvectors to the discrete segmentation.

In section 3 we propose a new multiclass spectral algorithm that combines aspects from a set of algorithms to produce a discrete solution. The discretization is efficiently computed in an iterative way using singular value decomposition and non-maximum suppression. Mostly of previous works ([4], [5]) use a k-means clustering to get a discrete solution from eigenvectors. Although these methods can produce similar results to our approach, they may take twice as long to converge. Moreover, while for k-means a good initial estimation is crucial our method is robust to a random initialization.

2 Spectral Segmentation

2.1 Notation

We introduce some notation, before describing the algorithm in more detail. Let the symmetric matrix $W \in R^{n \times n}$ denote the weighted adjacency matrix for a graph $G = (V, E)$ with nodes V representing pixels and edges E whose weights capture the pair-wise affinities between pixels. Let A and B represent a bipartition of V, i.e., $A \cup B = V$ and $A \cap B = \emptyset$. The degree of dissimilarity between these two groups can be computed as total weight of the edges that must be removed to separate the groups. In graph theoretic language, it is called the *cut*:

$$cut\,(A, B) = \sum_{i \in A, j \in B} W\,(i, j) \ . \tag{1}$$

Although there are efficient computational algorithms to find partitions that minimizes the *cut* value, this criterion favours partitions which have small sizes [11]. Shi and Malik [8] presented an extension of the *cut* criterion, called *normalized cut* criterion:

$$ncut\,(A, B) = \frac{cut\,(A, B)}{links\,(A, V)} + \frac{cut\,(A, B)}{links\,(B, V)} \ , \tag{2}$$

where $links(A, V)$ is the total edges weights connecting nodes of A to all nodes in the graph, and $links(B, V)$ is similarly defined. This new criterion avoids the segmentation of separated nodes. If we define $links(A, A)$ as the total weights of edges connecting nodes within A, we can also define a measure for the degree of similarity within groups for a given partition. Using $links(A, V)$ as a normalization term, we can get *normalized links* such as:

$$nlinks\,(A, B) = \frac{links\,(A, A)}{links\,(A, V)} + \frac{links\,(B, B)}{links\,(B, V)} \ . \tag{3}$$

A simple calculation shows that $ncut\,(A, B) = 2 - nlinks\,(A, B)$. Hence minimizing the degree of dissimilarity between the groups and maximizing the degree of similarity within the group, can be satisfied simultaneously by the *normalized* *cut*. Therefore, this criterion favours both tight connections within partitions and loose connections between partitions. Among numerous partitioning criterion only *minimum cut* [11] and *normalized cut* have this duality property.

A common matrix representation of graphs is the Laplacian. Let D be the degree diagonal matrix of W such that $D_{ii} = \sum_j W_{ij}$, i.e. D_{ii} is the sum of the weights of the connections from node i to all other nodes in the graph W. Then the Laplacian of W is the matrix $L = (D - W)$.

2.2 Previous Works

We can classify spectral methods in two classes: recursive spectral segmentation [8] - these algorithms try to split the data into two partitions based on a single eigenvector and are then recursively used to generate more partitions; and multi-way spectral segmentation ([4], [5], [12]) - these algorithms use information from multiple eigenvectors to do a direct multi-way partition of data. Experimentally it has been observed that using more eigenvectors and directly computing a k way partitioning produces better results (e.g. [2], [4], [5]).

As we saw above, a good segmentation corresponds to a partitioning scheme that separates all the nodes of a graph by cutting off the weakest links among them, i.e. minimizes the *cut* value. Wu and Leahy [11] proposed a clustering method based on the minimum criterion that minimizes (1). However, as the authors also noted in their work, and since the *cut* increases with the number of edges going across the two clusters, the minimum cut criteria favours cutting small sets of isolated nodes in the graph.

Shi and Malik proposed to use a normalized similarity criterion to evaluate a partition. One key advantage of using the *normalized cut* is that it makes possible to find a good approximation to the optimal partition[1]. The approximation to the optimal partition can be found by computing:

$$min_x ncut\left(x\right) = \min_y \frac{y^T\left(D - W\right)y}{y^T D y} \quad , \tag{4}$$

subject to the constraints that $y\left(i\right) \in \{-1, 1\}$ and $y^T D\mathbf{1} = 0$. y is a binary indicator vector specifying the group identity for each pixel and $\mathbf{1}$ is the vector of all ones. Notice that the above expression is a Rayleigh quotient, so if we relax y to take on real values (instead of two discrete values), the minimization becomes equivalent to solving the generalized eigenvalue system,

$$D^{-1/2}\left(D - W\right)D^{-1/2}z = \mu z \quad , \tag{5}$$

where $z = D^{1/2}y$. Shi and Malik verified that for the two-class *normalized cut* criterion, the global optimum in the relaxed continuous domain is given by the second smallest generalized eigenvector. This eigenvector of W is thresholded in order to cut the image into two parts. This process can be continued recursively

[1] Minimizing *normalized cut* exactly is a NP-complete problem.

as desired. However, as Shi and Malik noted, there is no guarantee that the solution obtained will have any relationship to the correct discrete solution.

The Scott and Longuet-Higgins algorithm [7] constructs a matrix M whose columns are the first k eigenvectors of W, normalizes the rows of M and constructs a matrix $Q = MM^T$. It produces a good segmentation if Q has only 1's or 0's. They use a not normalized similarity matrix. In [9], Weiss proposed an interesting combination of the Shi and Malik algorithm and the Scott and Longuet-Higgins algorithm and proved that it produces the best result. Meila and Shi algorithm [4] uses a random walk view in terms of the stochastic matrix P, with elements P_{ij}, obtained by normalizing the rows of W to sum 1. $P = D^{-1}W$ or $P_{ij} = W_{ij}/D_i$. This matrix can be viewed as defining a Markov random walk over nodes V, with P_{ij} being the transition probability $p\,[i \to j|\, i]$.

Equation (5) can be solved by a simpler eigensystem:

$$Px = \lambda x \quad . \tag{6}$$

The eigenvalues of P are $1 = \lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n \geq -1$ and the corresponding eigenvectors are $x_1, x_2, ..., x_n$. Then from (5) we get,

$$\mu_i = 1 - \lambda_i \text{ and } z_i = D^{1/2} x_i \quad . \tag{7}$$

for all $i = 1, ..., n$. Note that this ensures that the eigenvalues of P are always real and the eigenvectors are linearly independent. Meila and Shi [4] form a matrix X whose columns are the eigenvectors corresponding to the k largest eigenvalues of P and then cluster the rows of X as points in a k-dimensional space.

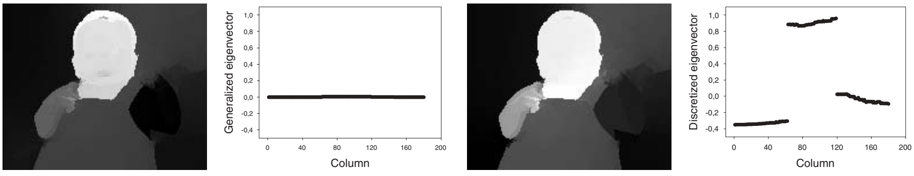
Ng *et al.* [5] use a different spectral mapping that behaves very similar to the Meila and Shi algorithm. It is proved that if the regions are well separated in the sense that the similarity matrix W is almost block diagonal, and if the sizes of the regions and the degrees of individual nodes don't vary too much, the rows of the X matrix cluster near k orthogonal vectors in R^k . This fact suggested the orthogonal initialization presented by Yu and Shi in [12].

3 Our Approach

We propose a multiclass algorithm based on a combined approach that uses random walk approach proposed by Meila and Shi [4] to create a normalized weight matrix P; Then, it solves an eigensystem and generates a matrix X, in the same manner as proposed by Ng *et al.* [5]; Finally, it uses a discretization process, proposed by Yu and Shi [12], more efficient than the k-means method, since it is robust to random initialization and converges faster.

3.1 The Algorithm

In an ideal case, the eigenvectors should only take on discrete values and the signs of the values can tell us exactly how to partition the graph. However, the eigenvectors can take on continuous values with very small variation among



(a) (b) (c) (d)

Fig. 1. Continuous vs. discretized eigenvectors: **a.** A generalized continuous eigenvector of W. **b.** A horizontal cross section through the pixels in a. **c.** The discrete solution of the same eigenvector. **d.** A cross section through the pixels in c.

them. Figure 1 shows the relation between continuous and discretized eigenvectors. Although there is correct information in this continuous solution, it could be very hard to split the pixels into segments.

Our goal is to find the right orthogonal transform that leads to a discrete solution that satisfies the binary constraints of (4), yet it is closest to the continuous optimum. The result of such discrete solution is presented in Fig. 1.d. Note that pixels referring to the head are nearly all 1, while others are much smaller. From this result it is very easy to segment the image.

To obtain a discrete solution we follow the heuristics presented by Yu and Shi in [12]. Due to the orthogonal invariance of the eigenvectors, any continuous solution can be replaced by $\widetilde{Y}R$ for any orthogonal matrix $R \in \mathrm{R}^{k \times k}$. An optimal partition Y should satisfy the following conditions:

$$min\ \phi\left(Y,R\right) = \left\|Y - \widetilde{Y}R\right\|^2 \text{ with } Y \in \{0,1\}_{n\times k}\,,\ Y\mathbf{1}_k = \mathbf{1}_n,\ R^TR = I_k \ . \quad (8)$$

where $\mathbf{1}_k$ and $\mathbf{1}_n$ are vectors of all ones, and I_k is the identity matrix.

This can be solved by an iterative optimization process:

- Given R, we want to minimize $\phi\left(Y\right) = \left\|Y - \widetilde{Y}R\right\|^2$. The optimal solution is given by non-maximum suppression:

$$Y\left(i,m\right) = istrue\left(m = \arg\max\left[\widetilde{Y}\left(i,k\right)\right]\right),\ i \in V,\ m \in \{1..k\} \ . \quad (9)$$

We let the first cluster centroid be a randomly chosen row of the continuous solution $\widetilde{Y}$, and then repeatedly choose as the next centroid the row of $\widetilde{Y}$ that is closest to being 90^0 from all the centroids already picked.

- Given Y, we want to minimize $\phi\left(R\right) = \left\|Y - \widetilde{Y}R\right\|^2$. The solution is given by singular value decomposition (SVD):

$$U \cdot \Omega \cdot \widetilde{U}^T = SVD\left(Y^T\widetilde{Y}\right) \ . \quad (10)$$

So, we can get,

$$R = \widetilde{U}U^T \text{ with } \min\phi\left(R\right) = 2\left(n - tr\left(\Omega\right)\right) \ . \quad (11)$$

Such iterations monotonously decrease the distance between a discrete solution and the continuous optimum. The larger $tr\left(\Omega\right)$ is, the closer Y is to $\widetilde{Y}R$.

Our segmentation algorithm consists of the following steps:

1. Set the diagonal elements W_{ii} to 0 and compute the normalized matrix P.
2. Let $1 = \lambda_1 \geq ... \geq \lambda_k$ be the k largest eigenvalues of P and $x_1, ..., x_k$ the corresponding eigenvectors. Form the matrix X by stacking the eigenvectors in columns.
3. Form the matrix $\widetilde{Y}$ from X by renormalizing each of X's rows to have unit length: $\widetilde{Y} = X \cdot Diag^{-1/2}\left(XX^T\right)$.
4. Initialize orthogonal matrix R with random lines of $\widetilde{Y}$.
5. Find the optimal discrete solution Y by (9).
6. Find the optimal orthogonal matrix R by (11).
7. While $|tr(\Omega) - \phi| > eps$ go to step 5.
8. Merge very similar neighbour regions which don't have edges among them.

3.2 Initialization of Affinity Matrix *W*

The quality of a segmentation based on the pair-wise similarities fundamentally depends on the weights that are provided as input. The weights should be large for pixels that should belong to the same group and small otherwise.

We associate to each pixel in the image a descriptor that captures brightness in a neighbourhood of the pixel. The similarity between two pixels is a function of the difference in their descriptors. Images are first convolved with oriented filter pairs (Fig. 2.b) to extract the magnitude of orientation energy (OE) of edge responses, as used by Malik *et al.* in [2]. At each pixel i, we can define the dominant orientation as $\theta^* = \arg\max OE_\theta$ and OE^* as the corresponding energy. The value OE^* is kept at the location of i only if it is greater than or equal to the neighbouring values. Otherwise it is replaced with a value of zero.

For each pair of pixels, pixel affinity is inversely correlated with the maximum contour energy encountered along the path connecting the pixels (Eq. 12). A large magnitude indicates the presence of an *intervening contour* and suggests that the pixels do not belong to the same segment.

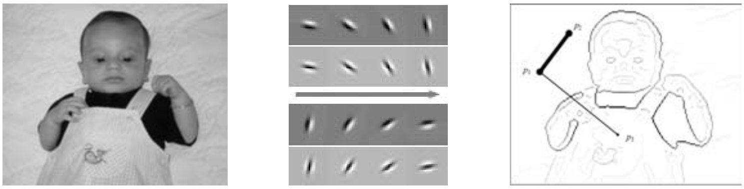
$$W(i,j) = \begin{cases} \exp\left[-\frac{\max_{t\in(0,1)} OE^*(s_i + t\cdot s_j)}{2\sigma_e^2 \cdot \max_l OE^*(s_l)}\right] & \text{if } \|s_i - s_j\| < r \\ 0 & \text{otherwise} \end{cases} , \qquad (12)$$

where s_i denotes the spatial location of pixel i, l is the straight line between pixels, t is a binary value which takes value '1' if the phases of the pixels are different, and r defines the city-block distance.

Figure 2 illustrates the intuition behind this idea. The intensity values of pixels p_1, p_2 and p_3 are very similar. However, there is a contour among them, which suggests that p_1 and p_2 belong to one group while p_3 belongs to another.

3.3 Experiments

To test our algorithm, we applied it to a set of images from the Berkeley Segmentation Dataset [3]. It contains 12.000 manual segmentations of 1.000 images by 30 human subjects. Each image has been segmented by at least 5 subjects,

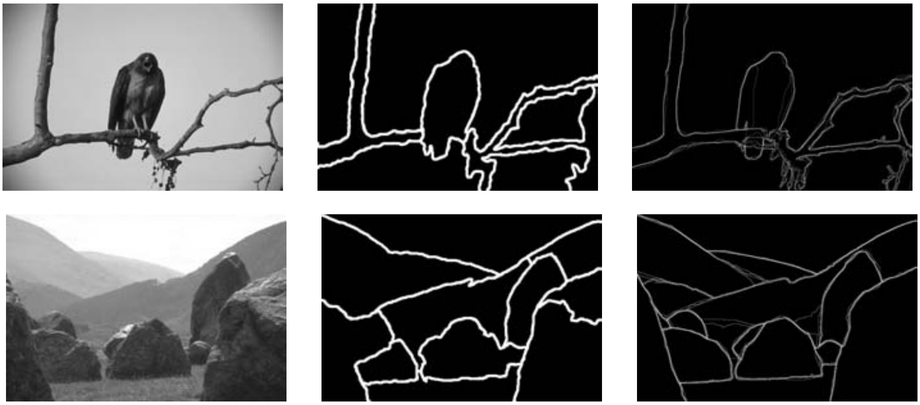


(a) (b) (c)

Fig. 2. Similarity matrix W is computed based on intensity edge magnitudes: **a.** The original image. **b.** The oriented filter pairs. **c.** Orientation energy.

so the ground truth is defined by a set of human segmentations. Martin *et al.* [3] declare two pixels to lie in the same segment only if all subjects declare that.

Figure 3 compares our results with the ground truth defined by Martin *et al.*. The column (Fig. 3.b) shows the results of some experiments with our algorithm and column (Fig. 3.c) represent the ground truth. Note that these ground truth images represent the probability that a segment will be chosen, if analysed by a person. We can see that our method reliably finds segments consistent with that an human would have chosen.



(a) (b) (c)

Fig. 3. Results of some experiments with the proposed algorithm: **a.** The original image. **b.** Our results. **c.** Berkeley ground truth.

3.4 Computation Time

The most time consuming part of the method is step 2, with a time complexity of $O\left(n^{3/2}k\right)$ using a Lanczos eigensolver [6]. The total time complexity of the algorithm is around $O\left(n^{3/2}k + 2nk^2\right)$. On a 1.4GHz Intel® Centrino™ processor, our method takes about 3 seconds on segmenting an 180×120 image, with $k = 10$, in C. This time could be greatly reduced by using the Nyström method proposed by Fowlkes *et al.* [1].

4 Conclusion

In this paper, we have presented a variation of the existing methods that combines aspects from different eigenvector segmentation algorithms. The heuristics are simple to implement as well as computationally efficient. Experimentally, we have demonstrated the potential of our approach for brightness and proximity image segmentation. However this model is general and can also be applied in a variety of image analysis. The improvement of the methodology can be achieved by designing better similarity distances between pixels. This can be done by using other cues such as texture or colour. However, good ways of combining these cues into one similarity matrix is still an open issue. Nevertheless, in the context of a specific application, dedicated similarity distances could be defined and lead to more precise segmentation results.

References

1. Fowlkes, C., Belongie, S., Chung, F., Malik, J.: Spectral Grouping Using Nyström Method, Trans. on Pattern Analysis and Machine Intelligence, 26(2), (2004) 214-225
2. Malik, J., Belongie, S., Leung, T., Shi, T.: Contour and Texture Analysis for Image Segmentation, International Journal of Computer Vision, 43(1), (2001) 7-27
3. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics, IEEE Int. Conf. on Computer Vision, Volume 2, (2001) 416-424
4. Meila, M., Shi, J.: Learning Segmentation by Random Walks, In Neural Information Processing Systems, (2000) 873-879
5. Ng, A., Jordan, M., Weiss, Y.: On Spectral Clustering: Analysis and an Algorithm, Advances in Neural Information Processing Systems 14, (2002) 849-856
6. Scott, D.S.: Solving Sparse Symmetric Generalized Eigenvalue without Factorization, SIAM Journal of Numerical Analisys, Volume 18, (1981) 102-110. Software available at: *http://www.netlib.org/laso/snlaso.f*
7. Scott, G.L., Longuet-Higgins, H.C.: Feature Grouping by "Relocalisation" of Eigenvectors of the Proximity Matrix, Proc. British Machine Vision Conf., (1990) 103-108
8. Shi, J., Malik, J.: Normalized Cuts and Image Segmentation, IEEE Trans. on Pattern Analysis and Machine Intelligence, 22(8), (2000) 888-905
9. Weiss, Y.: Segmentation Using Eigenvectors: A Unifying View, In International Conference on Computer Vision, (1999) 975-982
10. Wertheimer, M.: Laws of Organization in Perceptual Forms (partial translation), W. B. Ellis Editor, A Sourcebook of Gestalt Psychology, (1938) 71-88
11. Wu, Z., Leahy, R.: An Optimal Graph Theoretic Approach to Data Clustering: Theory and its Application to Image Segmentation, IEEE Trans. on Pattern Analysis and Machine Intelligence, Volume 15, (1993) 1101-1113
12. Yu, S., Shi, J.: Multiclass Spectral Clustering, International Conference on Computer Vision, Nice, France, (2003) 313-319