

Image Segmentation using Linear Algebra and Graph-Based Techniques

Authors

Gauranga Kumar Baishya - MDS202325

Esha Bhattacharya - MDS202324

Hiba A P - MDS202326

Instructor

Dr. Priyavrat Deshpande

Associate Professor, Chennai Mathematical Institute

pdeshpande@cmi.ac.in

Acknowledgement

This work is based on the paper titled *Normalized Cuts and Image Segmentation*, authored by *Jianbo Shi* and *Jitendra Malik* published in IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL.22, NO.8, August 2000.

Our sincere gratitude goes to Professor Priyavrat Deshpande for his invaluable guidance and support throughout the project, shaping its direction and focus.

Our gratitude also goes to all the participants who generously shared their time and insights, making this collaborative endeavor possible. Their contributions enriched our research and facilitated meaningful conclusions.

Contents

1	Introduction	3
1.1	The perpetual grouping problem	3
1.2	Gestalt's Psychology	3
1.3	Philosophical Debates to Computational Approaches	6
2	Graph Partitions	6
2.1	Preliminaries	6
3	NCut Optimisation	7
3.1	Formulating the Minimisation Problem	7
3.2	Relaxing the Formulated Minimisation problem	8
4	Algorithms	9
4.1	Recursive Two-Way NCut	9
4.1.1	Constructing the graph	9
4.1.2	Solve for eigenvectors	9
4.1.3	Partition the graph	10
4.1.4	Recursion	10
4.2	Simultaneous K-Way Cut	10
5	Comparison with related Eigenvector based Models	11
5.1	First function	12
5.2	Second Function	13
5.3	Third Function	14
6	Implementation	15
6.1	Python Script	15
6.2	Description of the Segmentation Results	15
7	Conclusion	16
8	Bibliography	18

1 Introduction

1.1 The perpetual grouping problem

The process of projecting the three-dimensional world onto two-dimensional images inevitably results in the loss of information, particularly regarding depth. Consequently, the visual system must reconstruct the true 3D structure from countless possible interpretations. While geometric theory alone is essential, it may not suffice. Additional assumptions, tailored to specific cases, become necessary. These assumptions should not overly restrict the functionality domain but instead leverage the general properties of scenes and imaging. Several questions arise in this context:

1. What are the overarching properties of scenes that enable domain-independent interpretation of images?
2. Do these properties need to be explicitly defined in terms of entities within the scene and constraints imposed by projective geometry? Alternatively, can real-world properties be translated into their image-plane counterparts without direct reference to 3D geometry?
3. If so, what are the image-plane entities to which these properties pertain?
4. How should constraints be articulated if not in terms of projective geometry?
5. When does the explicit 3D nature of the scene become relevant in the interpretation process?

The significance of image-plane structure hinges on image-plane entities. Entities that are structurally related are grouped together, allowing for the organization of image data across different scales. Rules governing this grouping can be entirely formulated based on the intrinsic properties of the tokens being grouped and their relationships in the image plane. Thus, grouping serves as an early inference mechanism regarding the structure of scene objects, operating without explicit reliance on 3D or domain-specific knowledge. Tokens eligible for grouping encompass blobs, edge segments, and geometrical features within image regions. The perceptual grouping problem involves the ability of the visual system to group individual elements, such as edges or regions, into coherent objects and surfaces. The history of research on perceptual grouping can be traced back to the Gestalt psychologists in the early 20th century, as will be discussed in the following section.

1.2 Gestalt's Psychology

Gestalt psychology, also known as gestaltism or configurationism, is a school of psychology and a theory of perception that emphasizes the processing of entire patterns and configurations, rather than merely focusing on individual components. It emerged in the early twentieth century in Austria and Germany as a rejection of the basic principles of Wilhelm Wundt's and Edward Titchener's elementalist and structuralist psychology. The Gestaltists were the first psychologists to systematically study perceptual grouping. According to Gestalt psychologists, the fundamental principle of perceptual grouping is the law of Prägnanz, also known as the law of good Gestalt. Prägnanz is a German word that directly translates to "pithiness" and implies salience, conciseness, and orderliness.

The law of Prägnanz states that people tend to experience things as regular, orderly, symmetrical, and simple.

1. **Law of proximity:** Objects close to each other are perceived as forming a group. For example, clusters of similar objects are grouped together.

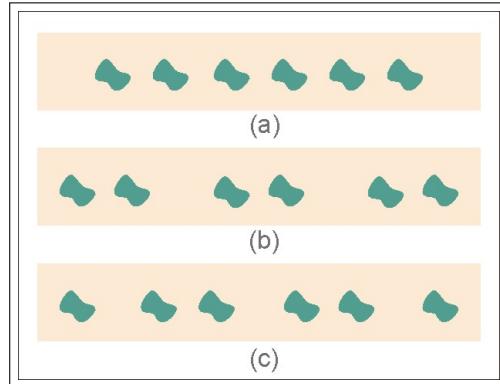


Figure 1: Proximity principle.

2. **Law of similarity:** Elements that are similar to each other are perceptually grouped together. This similarity can be in shape, color, or other qualities.

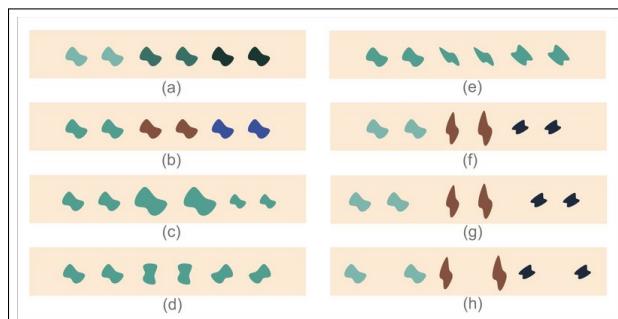


Figure 2: Similarity principle.

3. **Law of closure:** Humans tend to perceive objects as complete even when parts are missing. The mind fills in the gaps to form complete shapes or figures.

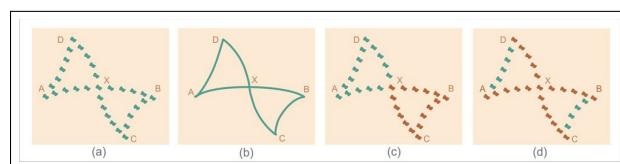


Figure 3: Closure principle.

4. **Law of symmetry:** Objects are perceived as symmetrical and forming around a center point. Symmetrical elements are grouped together to form coherent shapes.
5. **Law of common fate:** Objects are perceived as moving along the smoothest path. Elements with the same trend of motion are grouped together

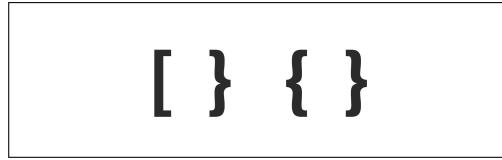


Figure 4: Symmetry principle.

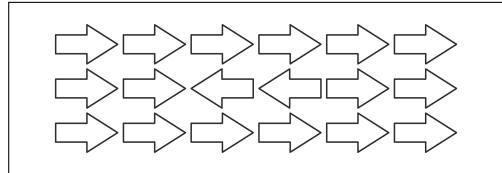


Figure 5: Common fate principle.

6. **Law of continuity:** Elements tend to be grouped together if they are aligned within an object, even if there is overlap or intersection.

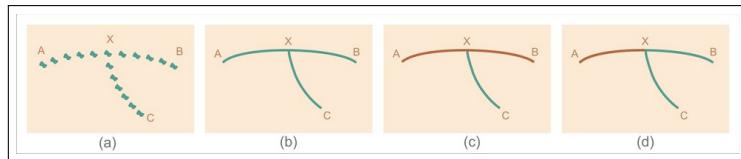


Figure 6: Continuity principle

7. **Law of past experience:** Visual stimuli are categorized according to past experience. Objects observed within close proximity or small temporal intervals are more likely to be perceived together.

For any given stimulus, one or more of these rules may be involved in determining the perceived grouping. If multiple rules are in play, they might either cooperate or compete. A key question to address is how to resolve conflicts that may arise as a result of applying these different rules. Researchers in computational vision have proposed explanations and mechanisms detailing the workings of the grouping processes and reasons for their existence. Our environment shapes the structure of images. Therefore, the criteria for grouping are closely linked to environmental properties. Some examples of environmental properties that can significantly influence the nature of image structure were discussed by Marr:

- The visible world comprises smooth surfaces with certain reflectances.
- Surface reflectance functions are often generated by processes operating at different scales.
- Tokens generated at a given scale by the same process tend to be similar.
- Markings on a surface, generated by the same process, often exhibit coherent spatial arrangements.
- Discontinuities in depth or surface orientation are typically smooth almost everywhere.

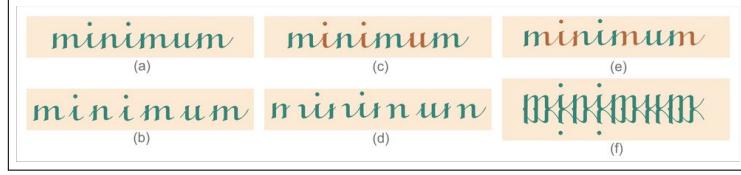


Figure 7: Past experience principle.

The human visual system generally disregards possible interpretations of an image that depend on an accidental viewpoint and favors interpretations that are viewpoint-stable. Moreover, interpretations requiring a larger number of simultaneous accidental alignments to support them are less likely to be adopted. Some suggest that the grouping criteria identified by Gestalt psychologists stem from assumptions that eliminate improbable interpretations.

1.3 Philosophical Debates to Computational Approaches

The issue of perceptual grouping in vision has a lengthy and intricate history spanning several centuries. Beginning with early philosophical debates on perception's nature and extending to modern neuroscientific investigations, understanding how we group visual elements to form coherent objects and scenes has remained a major focus of scholarly inquiry. Recent research has particularly emphasized the role of attention in perceptual grouping. It has been proposed that attention plays a crucial role by enabling the visual system to selectively process and integrate information from various sources. In summary, research on perceptual grouping has persisted for over a century and remains an active area of investigation in both psychology and neuroscience. Our primary goal is to computationally capture perceptual grouping with reasonable precision. Here, we approach image segmentation as a graph partitioning problem and introduce a novel global criterion, the normalized cut, for segmenting the graph.

2 Graph Partitions

2.1 Preliminaries

Consider an undirected weighted graph $G = (V, E)$ with weights given by the function w .

Definition 1. *Let V be partitioned into two disjoint sets A and B ($A \cup B = V$ and $A \cap B = \emptyset$). Then,*

$$\text{cut}(A, B) = \sum_{u \in A, v \in B} w(u, v)$$

Definition 2. *Let $A \subseteq V$. Then the association,*

$$\text{assoc}(A, V) = \sum_{u \in A, v \in V} w(u, v)$$

The task of identifying a partition of V that minimizes the cut value has been extensively explored, and a clustering approach utilizing the minimum-cut criteria has been established

by Wu and Leahy. However, the minimum-cut criteria tend to prioritize the partitioning of small sets of isolated vertices within the graph. This bias arises because, typically, the cut value rises as more edges traverse the partition. Therefore, to mitigate the inclination towards isolating small sets of vertices, the authors introduce the following disassociation criteria:

Definition 3. Let V be partitioned into two disjoint sets A and B ($A \cup B = V$ and $A \cap B = \emptyset$). Then,

$$NCut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(B, A)}{assoc(B, V)}.$$

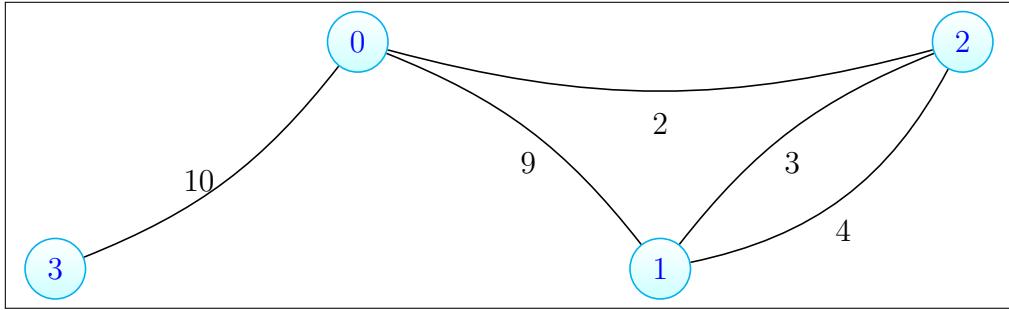


Figure 8: Graph, $G(V, E)$

For example, for the above graph, $G(V, E)$, we observe the following:

- Let $A = \{0,3\}$ and $B = \{1,2\}$. Also let $A^* = \{3\}$ and $B^* = \{0,1,2\}$. Then $Cut(A, B) = 11 = Cut(B, A)$ and $Cut(A^*, B^*) = 10 = Cut(B^*, A^*)$
- $Assoc(A, V) = 10 + 10 + 9 + 2 = 31$ and $Assoc(B, V) = 4 + 3 + 2 + 9 + 3 + 4 = 25$.
- $Assoc(A^*, V) = 10$ and $Assoc(B^*, V) = 4 + 3 + 2 + 9 = 18$
- $Ncut(A, B) = \frac{11}{31} + \frac{11}{25} \approx 0.79$ and
- $Ncut(A^*, B^*) = \frac{10}{10} + \frac{10}{18} \approx 1.55 > Ncut(A, B)$

Hence, NCut circumvents the issue of severing small sets of isolated vertices (for instance, in this case, vertex 3) even though A ($=\{3\}$) is strongly connected to G with a weight of 10, the above computations reveal that A and B are weakly connected to G compared to A^* and B^* . Nevertheless, determining the partition of a graph that minimizes NCut is NP-complete. Nonetheless, the paper presents an efficient method to obtain an approximate solution.

3 NCut Optimisation

3.1 Formulating the Minimisation Problem

Our goal is to find a partition of V into two disjoint sets A and B such that $A \cup B = V$ and $A \cap B = \emptyset$, which minimizes $NCut(A, B)$. Consider a partition of V into two disjoint

sets A and B . Let x be an $N = |V|$ dimensional indicator vector for membership in A , such that $x_i = 1$ if $i \in A$, else $x_i = -1$. Let $d(i) = \sum_j w(i, j)$.

Then,

$$\begin{aligned} \text{NCut}(A, B) &= \frac{\text{cut}(A, B)}{\text{assoc}(A, V)} + \frac{\text{cut}(B, A)}{\text{assoc}(B, V)} \\ &= \frac{\sum_{(x_i > 0, x_j < 0)} -w_{ij}x_i x_j}{\sum_{x_i > 0} d_i} + \frac{\sum_{(x_i < 0, x_j > 0)} -w_{ij}x_i x_j}{\sum_{x_i < 0} d_i}. \end{aligned}$$

Let D be an $N \times N$ diagonal matrix with d on its diagonal, W be an $N \times N$ symmetrical matrix with $W(i, j) = w_{ij}$, $k = \frac{\sum_{x_i > 0} d_i}{\sum_i d_i}$, and 1 be an $N \times 1$ matrix of all ones. Using the fact that $\frac{1+x}{2}$ and $\frac{1-x}{2}$ are indicator vectors for $x_i > 0$ and $x_i < 0$, respectively, we can write as:

$$\begin{aligned} 4. \text{NCut}(A, B) &= \frac{(1 - 2k + 2k^2)(\alpha(x) + \gamma) + 2(1 - 2k)\beta(x)}{k(1 - k)M} + \frac{2\alpha(x)}{M} \\ &= \frac{\frac{1-2k+2k^2}{(1-k)^2}(\alpha(x) + \gamma) + \frac{2(1-2k)}{(1-k)^2}\beta(x)}{\frac{k}{1-k}M} + \frac{2\alpha(x)}{M}, \end{aligned}$$

where $\alpha(x) = x^T(D - W)x$, $\beta(x) = 1^T(D - W)x$, $\gamma = 1^T(D - W)1$, and $M = 1^T D 1$.

Letting $b = \frac{k}{1-k}$ and since $\gamma = 0$, it becomes

$$\begin{aligned} &= \frac{(1 + b^2)(\alpha(x) + \gamma) + 2(1 - b^2)\beta(x)}{bM} + \frac{2b\alpha(x)}{bM} \\ &= \frac{(1 + b^2)(x^T(D - W)x + 1^T(D - W)1)}{b1^T D 1} + \frac{2(1 - b^2)1^T(D - W)x}{b1^T D 1} \\ &\quad + \frac{2bx^T(D - W)x}{b1^T D 1} - \frac{2b1^T(D - W)1}{b1^T D 1} \\ &= \frac{(1 + x - b(1 - x))^T(D - W)(1 + x - b(1 - x))}{b1^T D 1}. \end{aligned}$$

Setting $y = \frac{(1+x)}{2} - \frac{b(1-x)}{2}$, it is easy to see that $y^T D 1 = \sum_{x_i > 0} d_i - b \sum_{x_i < 0} d_i = 0$ (since $b = \frac{k}{1-k} = \frac{\sum_{x_i > 0} d_i}{\sum_{x_i < 0} d_i}$) and

$y^T D y = \sum_{x_i > 0} d_i + b^2 \sum_{x_i < 0} d_i = b \sum_{x_i < 0} d_i + b^2 \sum_{x_i < 0} d_i = b(\sum_{x_i < 0} d_i + \sum_{x_i > 0} d_i) = b1^T D 1$. Combining these, we get:

$$\text{NCut}(x) = \frac{y^T(D - W)y}{y^T D y},$$

with $y(i) \in \{1, -b\}$ and $y^T D 1 = 0$.

3.2 Relaxing the Formulated Minimisation problem

Since solving the actual minimization is NP-complete, we'll instead relax the problem. We'll consider the relaxed problem of minimizing $\frac{y^T(D - W)y}{y^T D y}$ where $y \in \mathbb{R}^{|V|}$ and $y^T D \mathbf{1} = 0$. We can consider $z = D^{\frac{1}{2}}y$. Then the above minimization is the same as minimizing

$$\frac{z^T D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}z}{z^T z}$$

where $z^T z_0 = 0$, $z_0 = D^{\frac{1}{2}} \mathbf{1}$. Now $D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}z_0 = D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}D^{\frac{1}{2}}\mathbf{1} = D^{-\frac{1}{2}}(D - W)\mathbf{1} = D^{-\frac{1}{2}}\mathbf{0} = \mathbf{0}$. Hence z_0 is an eigenvector of $D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}$ for eigenvalue 0. Now, for a graph, $D - W$ is known as the Laplacian, and is known to be positive semidefinite. Hence $D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}$ is also positive semidefinite. Thus all its eigenvalues are non-zero. As 0 is an eigenvalue, it is the smallest eigenvalue. We will now use the Rayleigh quotient property:

Theorem 1 (Rayleigh quotient property). *Let A be a real symmetric matrix. Under the constraint that x is orthogonal to the $j - 1$ smallest eigenvectors x_1, \dots, x_{j-1} , the quotient $\frac{x^T A x}{x^T x}$ is minimized by the next smallest eigenvector x_j and its minimum value is the corresponding eigenvalue λ_j .*

Using the Rayleigh property, $\frac{z^T D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}z}{z^T z}$ for z satisfying $z^T z_0 = 0$ is minimized by the next smallest eigenvector z_1 of $D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}$. Hence our minimization problem achieves the minimum at $z = z_1$, or $y = D^{-\frac{1}{2}}z_1$. One should Note that $D^{-\frac{1}{2}}z_1$ need not correspond to any valid x (due to our relaxation). We can still use this vector to determine a decent partition of the graph.

4 Algorithms

Lets have a look at the different algorithms for graph partitioning based image segmentation as discussed in the paper.

4.1 Recursive Two-Way NCut

4.1.1 Constructing the graph

We need to construct a graph from an image. The paper constructs a graph from images by taking the pixels as vertices and setting the edge weight w_{ij} between i and j as

$$w_{ij} = e^{-\|F(i) - F(j)\|_2^2} * \begin{cases} e^{-\|X(i) - X(j)\|_2^2} & \text{if } \|X(i) - X(j)\|_2 < r \\ 0 & \text{otherwise} \end{cases}$$

where $X(i)$ is the position of i , and $F(i)$ is a feature vector based on intensity, color or texture information at i , such as:

- $F(i) = I(i)$, the intensity value, for segmenting brightness images,
- $F(i) = [v, v \sin(h), v \cos(h)]$, where h, s, v are HSV values, for colored images.

4.1.2 Solve for eigenvectors

Using the construction in section 2, we want eigenvectors $D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}$. The standard eigenvalue problem for all eigenvectors takes $O(n^3)$ operations where n is the number of vertices in the graph. This is impractical for images with lots of pixels.

The paper instead uses the fact that for our problem:

1. Our graphs are locally connected, making the resulting eigensystems sparse
2. We only need the top few eigenvectors

3. The required precision for eigenvectors is low (since we only need to partition the graph using them).

Thus, we can instead use an eigensolver called the Lanczos method. The running time for this method is $O(mn) + O(mM(n))$, where m is the maximum number of matrix-vector computations required and $M(n)$ is the cost of a matrix-vector computation of Ax , where $A = D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}$.

Using the fact that W is sparse (locally connected graph), A is also sparse, and so a matrix-vector computation of Ax takes $O(n)$ time.

The constant factor is determined by the size of the spatial neighborhood of a vertex. The authors empirically found that they could remove up to 90% of the total connected with each of the neighborhoods when the neighborhoods are large without affecting the eigenvector solution of the system.

Combining these, each matrix-vector computation costs $O(n)$ operations with a small constant factor. In their experiments, the authors found that m is typically less than $O(n^{1/2})$.

4.1.3 Partition the graph

Now we can take the second smallest eigenvector and get our y . Ideally, by section 2.2, y should take on two discrete values, and using the signs of the values we could partition the graph. However as we relaxed the problem in section 2.3, y need not take on two discrete values. Thus we need to choose a splitting point to divide the graph.

We could choose 0 or the median value as the splitting point. The paper instead decides to check l evenly spaced splitting points, compute the best Ncut among them, and use the best point.

4.1.4 Recursion

We then recursively run the algorithm on the two partitioned parts, stopping when the NCut value exceeds a certain threshold.

4.2 Simultaneous K-Way Cut

Instead of just using the second eigenvector, we can use the top l eigenvectors as a l dimensional indicator vector for each pixel.

We first apply a clustering algorithm (like k-means) to obtain an over-segmentation of the image into k' groups. Then we can perform:

1. **Greedy pruning:** Iteratively merge two segments at a time until the number of segments is small enough. At each step we merge two segments which minimize the k -way Ncut, $\sum_{i=1}^k \frac{\text{cut}(A_i, V - A_i)}{\text{assoc}(A_i, V)}$, where A_i are the segments.
2. **Global recursive cut:** From the initial k' segments, we can build a condensed graph with the segments corresponding to vertices, and the weight of the edge between A_i and A_j is $\text{assoc}(A_i, A_j)$. We then recursively bipartition this graph according to the NCut criteria, either using the eigenvalue system discussed before or by exhaustive search (this is possible as k' is generally small).

5 Comparison with related Eigenvector based Models

The normalized cut formulation has a certain resemblance to the average cut, as well as the average association formulation. All three of these algorithms can be reduced to solving certain eigenvalue systems.

Finding clumps			Finding splits		
Discrete formulation		Normalized Cut	Average cut		
$\frac{\text{asso}(A,A)}{ A } + \frac{\text{asso}(B,B)}{ B }$		$\frac{\text{cut}(A,B)}{\text{asso}(A,V)} + \frac{\text{cut}(A,B)}{\text{asso}(B,V)}$ or $2 - \left(\frac{\text{asso}(A,A)}{\text{asso}(A,V)} + \frac{\text{asso}(B,B)}{\text{asso}(B,V)} \right)$	$\frac{\text{cut}(A,B)}{ A } + \frac{\text{cut}(A,B)}{ B }$		
$Wx = \bar{\lambda}x$		$(D-W)x = \bar{\lambda}Dx$ or $Wx = (1 - \bar{\lambda})Dx$	$(D-W)x = \bar{\lambda}x$		

Figure 9: Relationship between Normalized Cut and other eigenvector-based partitioning techniques

- Average association tends to favor "tight" clusters, which can result in the identification of small but densely connected clusters in the graph. While effective for Gaussian-distributed data, real-world datasets often contain a mixture of different distribution types, leading to undesirable outcomes.
- Normalized cut, on the other hand, aims to strike a balance between identifying cohesive clusters and avoiding excessive fragmentation. It seeks to achieve a balance between clustering and segmentation objectives.
- Average cut, unlike normalized cut, does not guarantee that the two partitions computed will exhibit strong within-group similarity.

To illustrate these above said points, we consider a set of randomly distributed data in 1 Dimension. The one-dimensional dataset consists of two subsets of points:

- The first 20 points are randomly distributed between 0 and 0.5.
- The remaining 12 points are randomly distributed between 0.65 and 1.

Constructing a Graph: Each data point corresponds to a node in the graph. The weighted edge connecting two points is defined to be inversely proportional to the distance between the nodes. We express this relationship using a weight function:

$$w(x) = f(d(x)),$$

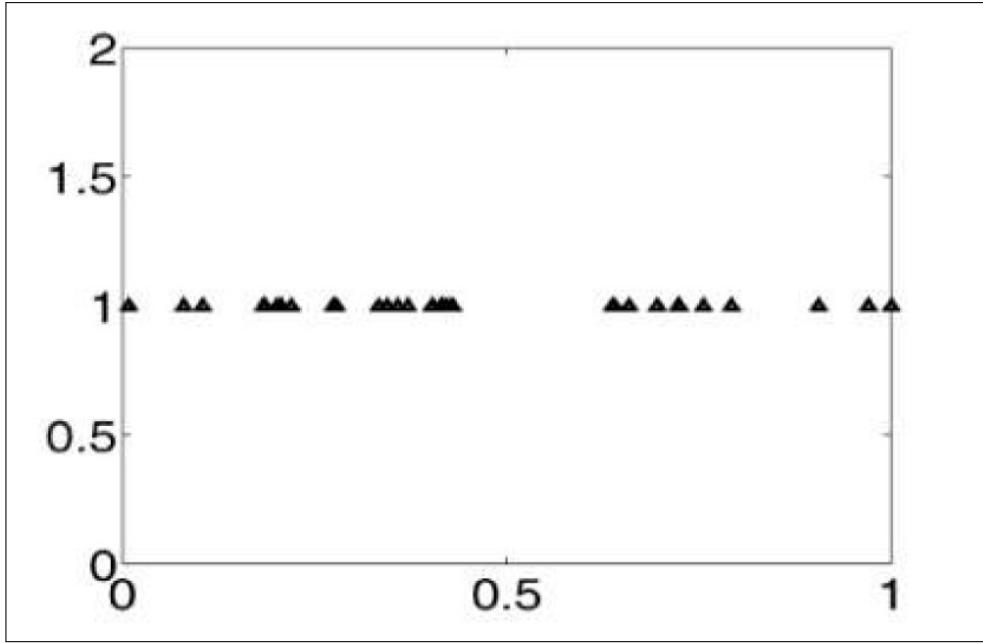


Figure 10: A set of randomly distributed data in one dimension

where $d(x)$ represents a distance function. We employ three monotonically decreasing weight functions, each with a different rate of fall-off.

5.1 First function

Consider the function,

$$w(x) = e^{\left(\frac{-d(x)}{-0.1}\right)^2}$$

This function has the fastest decreasing rate among the three considered functions. In Figure 11 below, the left panel illustrates the rate of fall-off of the first function:

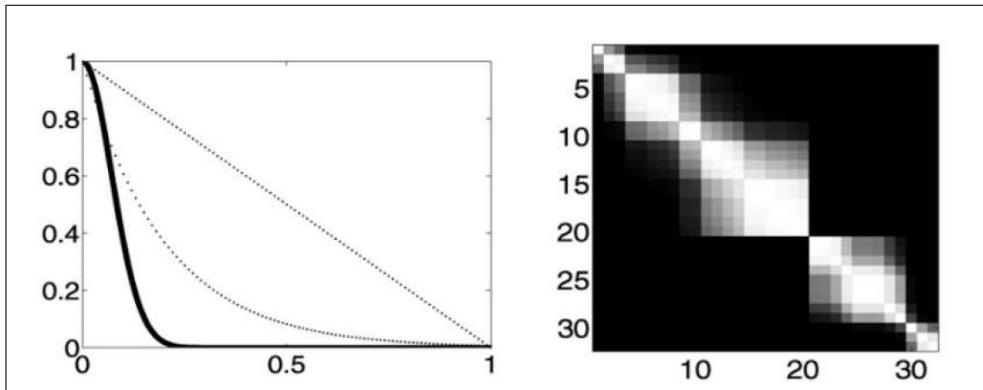


Figure 11:

$$w(x) = e^{\left(\frac{-d(x)}{-0.1}\right)^2}$$

represented by a solid line.

Meanwhile, the right panel displays the corresponding graph weight matrix W , where weight is depicted by brightness, indicating that higher weight values correspond to brighter regions and vice versa. With this weight function, only nearby points are connected as depicted below in Figure 12. The key observations are written below:

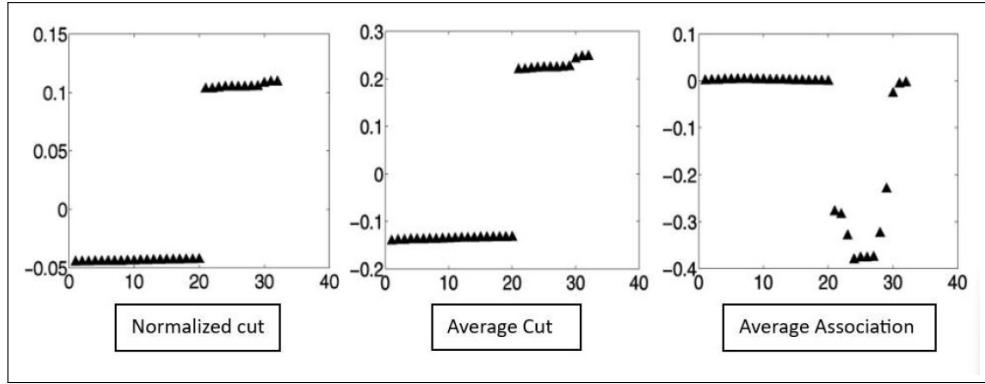


Figure 12:

- Utilizing the second extreme eigenvector, both Normalized and Average Cut methods effectively partitioned the data points into two clusters, reflecting the true underlying structure.
- However, the Average Association approach failed to produce accurate clustering results. It resulted in partitioning the data points into isolated small clusters, highlighting its tendency to bias towards identifying "tight" clusters.

5.2 Second Function

The second function is defined as:

$$w(x) = 1 - d(x)$$

This function exhibits the slowest decreasing rate among the three as depicted below in Figure 13. With this weight function, most points have non-trivial connections to the rest as depicted below in Figure 14. The key observations are also laid down.

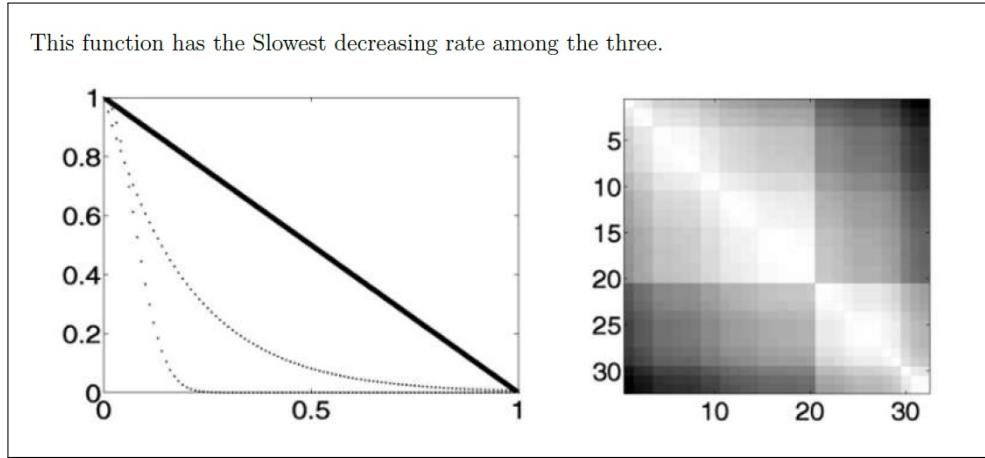


Figure 13:

- Normalized Cut produces the correct partition.
- Average Association also yields the correct partition by easily identifying the two tight clusters and eliminating a few edges with heavy weights across them.

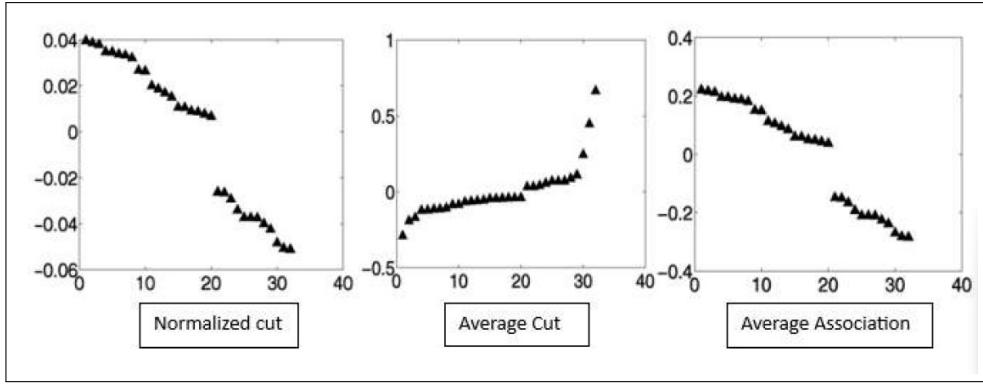


Figure 14:

- However, Average Cut fails because the cluster on the right exhibits less within-group similarity compared to the cluster on the left. Therefore, the Average Cut algorithm struggles to decide where to make the cut.

5.3 Third Function

The third function is defined as:

$$w(x) = e^{\left(\frac{-d(x)}{-0.2}\right)}$$

This function exhibits a moderate decreasing rate as depicted below in Figure 15. With this weight function, the connections between nearby points are balanced against connections to faraway points as depicted below in Figure 16. Based on these, the following observations can be made:

- All three algorithms perform satisfactorily in partitioning the data.
- However, Normalized Cut produces a clearer solution compared to the other two algorithms. This may be attributed to its ability to balance the goals of clustering and segmentation.

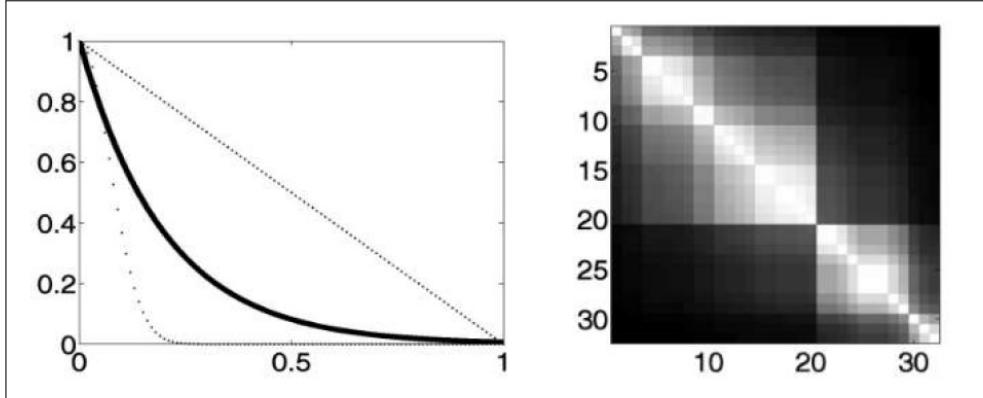


Figure 15:

In summary, based on the observations across the three scenarios presented, it can be concluded that Normalized Cut consistently performs well.

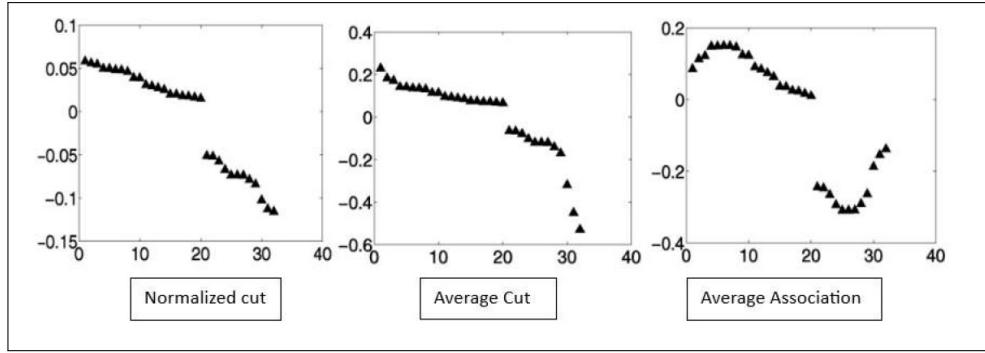


Figure 16:

6 Implementation

6.1 Python Script

Here is the Python script used for image segmentation and here is a link to the Colab notebook.

```
from skimage import io
from skimage import segmentation, color
from skimage.future import graph
import matplotlib.pyplot as plt

img = io.imread('/content/Neumann.jpg')

labels1 = segmentation.slic(img, compactness=30, n_segments=1000,
    start_label=1)
out1 = color.label2rgb(labels1, img, kind='avg', bg_label=0)

g = graph.rag_mean_color(img, labels1, mode='similarity')
labels2 = graph.cut_normalized(labels1, g)
out2 = color.label2rgb(labels2, img, kind='avg', bg_label=0)

fig, ax = plt.subplots(nrows=2, sharex=True, sharey=True, figsize=(6, 8))

ax[0].imshow(out1)
ax[1].imshow(out2)

for a in ax:
    a.axis('off')

plt.tight_layout()
```

6.2 Description of the Segmentation Results

Here is a brief description of the two types of images output by the code, Figure 18, focusing on the image of John von Neumann's childhood picture, Figure 17:

Superpixel Segmentation (First Image of Fig 17): The first image shows the result of applying the Simple Linear Iterative Clustering (SLIC) algorithm to segment the image into superpixels. Superpixels are regions of the image that share similar color and texture characteristics. This segmentation reduces the complexity of the image by grouping pixels

into larger, meaningful segments, which can simplify subsequent image processing tasks. In this visualization, each superpixel is represented by the average color within that segment, creating an abstracted version of the original image.

Graph-Based Segmentation (Second Image of Fig 18): The second image displays the result of a graph-based segmentation technique using the Region Adjacency Graph (RAG) and Normalized Cut algorithm. This method involves constructing a graph where each node represents a superpixel, and edges between nodes represent the similarity between adjacent superpixels. The Normalized Cut algorithm then partitions this graph to minimize the similarity between different segments while maximizing the similarity within each segment. This results in a more refined segmentation, highlighting the distinct regions in the image based on color and texture similarity.



Figure 17: John von Neumann at age 7 (1910)

7 Conclusion

In conclusion, we have developed a top-down grouping algorithm that prioritizes perceptual grouping to extract the global impression of a picture. By minimizing Normalized Cut, which is an unbiased measure of disassociation between subgroups of a graph, we indirectly maximize Normalized Association, an unbiased measure of the total association within the subgroups. The relationship between Normalized Cut and Normalized Association is given by the equation:

$$NCut(A, B) = 2 - N \cdot NAssoc(A, B)$$



Figure 18:

This criterion remains unbiased. By transforming the problem of computing the minimum Normalized Cut into solving a generalized eigenvalue system, the algorithm becomes more efficient.

8 Bibliography

1. K. Koffka, *Principles of Gestalt Psychology*, Harcourt Brace, New York, 1935.
2. M. Wertheimer, "Laws of Organization in Perceptual Forms," in *A Source Book of Gestalt Psychology*, W. D. Ellis (ed), pp. 71-88, Harcourt Brace, 1938.
3. D. Marr, *Vision*, W.H. Freeman and Company, San Francisco, CA, 1982.
4. D. G. Lowe, *Perceptual Organization and Visual Recognition*, Kluwer Academic, Boston, 1985.
5. <https://www.researchgate.net>