Image Segmentation and Regular Partitions

Merit Geldmacher [*]

Abstract

This bachelor thesis contributes to the study of image segmentation and spectral graph theory. It aims at finding a link between image segmentation and equitable partitions, which are a common concept in spectral graph theory. The analysis consists of an algorithmic and a theoretical approach. During our research we could not find a strong connection between those two, however, we propose further areas to be investigated.

1 Introduction

The internet, Facebook and our brain are only a few examples of complex networks we encounter in everyday life and we often observe that such networks keep growing larger. One approach to tackle complex networks is to model them as graphs. Due to complexity reasons it is desirable to find a small number of graph parameters that characterise and capture a graph's features.

One example where it is desirable to reduce complexity are images that often consist of more than several million pixels. Figure 1 below for example is made up of more than 16 million pixels. It may be of interest here to e.g. separate the foreground from the background, one example of image segmentation.

[*] Merit Geldmacher received a bachelor degree in Econometrics & Operations Research at Maastricht University in 2016, where she currently takes the Research Master in the same field.
Contact: m.geldmacher@student.maastrichtuniversity.nl

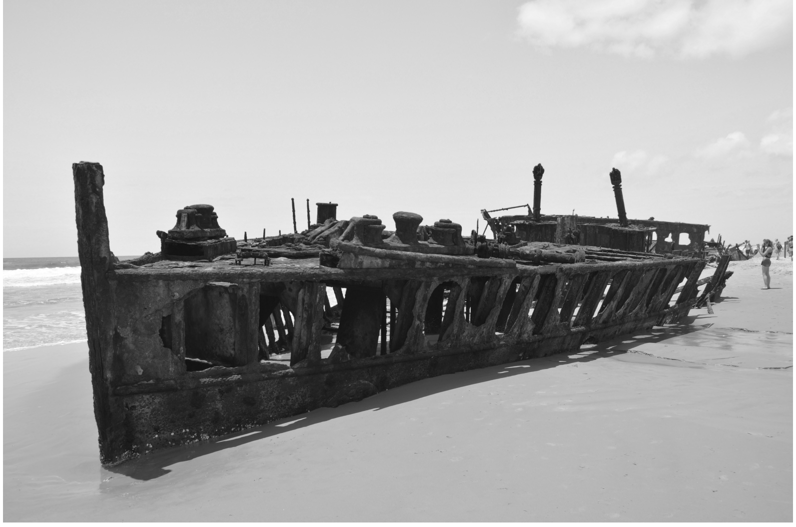


Figure 1: Example Image

The ultimate goal of image segmentation is to partition the image into several segments or clusters that capture the image's characteristics but leave out irrelevant details. Criteria could be colour, brightness and/or texture of different parts of an image.

Image segmentation is being applied in various areas, for example in crime detection through face recognition and in medicine through tumour detection. Furthermore, image segmentation is implemented in smart car algorithms through pedestrian detection.

There are many approaches to image segmentation, one of them is applying spectral graph theory (see Casaca et al. (2013), Cour et al. (2005), Shi and Malik (2000)). The general idea is to model an image as a complete graph where each node corresponds to a pixel. One can then construct a matrix that reveals the relationship between every pair of pixels. Every graph has matrices associated with it, a simple example is the (boolean) adjacency matrix that indicates which vertices are connected and which are not. The set of eigenvalues of a matrix is called the *spectrum* of that matrix and spectral graph theory studies exactly that relationship between the eigenvalues and the graph's structural characteristics. It is known that the spectrum of a graph captures important structural features of the graph.

This thesis addresses whether there is a link between image segmentation and the very well-known topics of spectral graph theory and equitable partitions. To our knowledge such a link has not yet been demonstrated before. However, we have a strong suspicion that those are linked for particular types of images.

Firstly, we give an overview about related work that has been done about image segmentation. Next, we clarify the notation and the concepts that we applied during our research and that are addressed in this thesis. Thirdly,we outline our approach to the research question. Then we present results of our research, split in the outcome of empirical and more theoretical work. Finally, the thesis concludes with summary of the results and further outlooks on what to research on in the future. Refer to the appendix for R code.

2 Literature Review

There is various literature about image segmentation available and giving a comprehensive and detailed overview about this topic is beyond the scope of this thesis. Instead, we present a few selected results that supplement our research.

Shi and Malik (2000) focus on the global features of an image in their research. They translate image segmentation into a graph partitioning problem and employ the normalised cut criterion before they use spectral theory to optimise this criterion. They found that the normalised cut which is proven to be NP complete can serve as a measure of the goodness of a partition. Minimising the normalised cut which measures disassociation between groups (the different parts of a partition) automatically maximises the normalised association which measures association within groups.

It is shown that the smaller the normalised cut is, the better is the graph partition. Thus, it is desirable to minimise the normalised cut which can also be formulated as a generalised eigenvalue problem. In their approach the image is partitioned through the graph's spectrum and this process can be applied several times to refine the results. Furthermore, Shi and Malik (2000) presents a grouping algorithm that constructs an affinity graph and uses the eigenvectors with the smallest eigenvalues of

$$(D - W)x = \lambda Dx \tag{1}$$

where D is an n x n diagonal matrix where d_i, the sum of weights from node i to all other nodes, is on the diagonal. W represents the n x n affinity matrix and x is a vector that has $x_i = 1$ if node i is in the first set and -1 otherwise, that is if i is in the second set. They use the eigenvector corresponding to the second smallest eigenvalue to bipartition the graph and recursively bipartition further if desired.

Shi and Malik (2000) conclude that their approach works well on synthetic and real images and extracts the global features of the images.

Casaca et al. (2013) combine spectral graph theory with image decomposition, inner product-based similarity metric that allows user assistance. The first step of their approach is to decompose the image into texture and cartoon parts. Next, an affinity graph is created and a gradient-based inner production function determines its edge weights. This steps allows the user to intervene and to modify weights if calculation and perception differ significantly. The image is partitioned through the spectral cut of the affinity graph and by deploying an image coarsening process the graph's size and thus the computational effort is decreased significantly. Quantitative and qualitative analyses of the results reveal that their approach outperforms other segmentation techniques regarding accuracy and robustness. They conclude that combining decomposition into cartoon and texture parts and spectral cut gives an efficient approach for image segmentation.

Another technique to partition in image is multiscale image segmentation for which the underlying graph is represented in multiple levels. Furthermore, edges and nodes are varied throughout the process. A benefit of this approach is that both local and global structures can be detected more easily.

Cour et al. (2005) developed a multiscale spectral image segmentation algorithm that is efficient enough to work for larger images. The algorithm segments the different scales parallel and simultaneously to address both coarse and fine level characteristics of the image. Larger images are condensed into multiple scales and by increasing the neighbourhood of a pixel it is ensured that relevant image structures are kept. Furthermore, an inter-scale constraint is employed to avoid inter-scale miscommunication and inconsistency. Their segmentation criterion is based on the normalised cut framework as presented by Shi and Malik (2000). Pixel similarity is summarised in a graph that is partitioned using spectral criteria. Cour et al. (2005) find that especially larger images that failed to be segmented due to complexity reasons can be partitioned through their multiscale spectral image segmentation algorithm. Both running time and quality of the segmentation have significantly improved when compared to other algorithms.

Preliminaries In this section we will present some theorems and corollaries with intuitive explanations and we will clarify notation that is employed throughout the thesis.

2.1 Theoretical Preliminaries

Theorem 1. *(Interlacing, see Abiad (2012)) Let* $\boldsymbol{A}$ *be a symmetric* n *x* n *matrix with eigenvalues* $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n$ *and respective eigenvectors* $\mathbf{u_1}, \mathbf{u_2}, ..., \mathbf{u_n}$*. For some integer* $m < n$*, let* $\boldsymbol{S}$ *be a real* n *x* m *matrix such that* $S^T S = I$ *(its columns are orthonormal), and consider the* m *x* m *matrix* $\mathbf{B} = \mathbf{S^T A S}$ *with eigenvalues* $\mu_1 \geq \mu_2 \geq ... \geq \mu_m$ *and respective eigenvectors* $\nu_{\mathbf{1}}, \nu_{\mathbf{2}}, ..., \nu_{\mathbf{m}}$*. Then the following statements hold.*

(i) The eigenvalues of $\boldsymbol{B}$ *interlace the eigenvalues of* $\boldsymbol{A}$*, that is:*
$\lambda_i \geq \mu_i \geq \lambda_{n-m+i} \quad 1 \leq i \leq m.$

(ii) If $\mu_i = \lambda_i$ *or* $\mu_i = \lambda_{n-m+i}$ *for some* $i \in [1, m]$ *then* $\boldsymbol{B}$ *has a* μ_i*-eigenvector* ν *such that* $\mathbf{S}\nu_{\mathbf{i}}$ *is a* μ_i*-eigenvector of* $\boldsymbol{A}$*.*

(iii) If for some integer l*,* $\mu_i = \lambda_i$ *for* $i = 1, ..., l$ *(or* $\mu_i = \lambda_{n-m+i}$ *for* $i = l, ..., m$*), then* $\mathbf{S}\nu_{\mathbf{i}}$ *is a* μ_i*-eigenvector of* $\boldsymbol{A}$ *for* $i = 1, ..., l$ *(respectively* $i = l, ..., m$*).*

(iv) If the interlacing is tight, that is, for some $0 \leq k \leq m, \lambda_i = \mu_i \quad (1 \leq i \leq k)$ *and* $\mu_i = \lambda_{n-m+i} \quad (k+1 \leq i \leq m)$*, then* $\mathbf{SB} = \mathbf{AS}$*.*

For a formal proof of this theorem please refer to Abiad (2012).

Let $\mathcal{P} = \{V_1, ..., V_m\}$ be a partition of the vertex set V, with each $V_i \neq 0$. Let $\mathbf{A}$ be partitioned according to $\mathcal{P}$:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A_{1,1}} & \cdots & \mathbf{A_{1,m}} \\ \vdots & & \vdots \\ \mathbf{A_{m,1}} & \cdots & \mathbf{A_{m,m}} \end{bmatrix},$$

where $\mathbf{A_{i,j}}$ denotes the submatrix (block) of $\mathbf{A}$ formed by rows in V_i and columns in V_j. The *characteristic matrix* $\tilde{\mathbf{S}} = (\tilde{s}_{uj})$ is the n x m matrix whose j-th column is the *characteristic vector* of V_j for $j = 1, ..., m$, that is

$$(\tilde{s}_{uj}) = \begin{cases} 1 & if\ u \in V_j, \\ 0 & otherwise. \end{cases}$$

The *quotient matrix* is the m x m matrix $\tilde{\mathbf{B}} = (\tilde{b}_{ij})$ whose entries are the average row sums of the blocks of $\mathbf{A}$, more precisely:

$$(\tilde{b}_{ij}) = \frac{1}{|V_i|}\mathbf{j}^T\mathbf{A}_{i,j}\mathbf{j} = \frac{1}{|V_i|}\left(\tilde{\mathbf{S}}^\mathbf{T}\mathbf{A}\tilde{\mathbf{S}}\right)_{i,j}$$

The partition is called *regular* or *equitable* if each block $\mathbf{A_{i,j}}$ of $\mathbf{A}$ has constant row (and column) sum, that is, $\mathbf{A}\tilde{\mathbf{S}} = \tilde{\mathbf{S}}\tilde{\mathbf{B}}$.

Throughout our investigations we only employed bipartitions, so $\tilde{\mathbf{S}}$ is of dimension n x 2 and $\tilde{\mathbf{B}}$ is a square matrix of dimension 2.

Corollary 1. *Suppose* $\tilde{\mathbf{B}}$ *is the quotient matrix of a symmetric partitioned matrix* $\boldsymbol{A}$*.*

(i) The eigenvalues of $\tilde{\mathbf{B}}$ *interlace the eigenvalues of* $\boldsymbol{A}$*.*

(ii) If the interlacing is tight then the partition is regular.

For a formal proof of this corollary please refer to Abiad (2012).

The spectrum of a matrix A is the set of its eigenvalues given by

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v},$$

where $\mathbf{v}$ is an eigenvector and λ is the corresponding eigenvector. Eigenvalues can be obtained solving the characteristic equation following from

$$(\mathbf{A} - \lambda\mathbf{I}) = 0,$$

where $\mathbf{I}$ is the identity matrix.

Example: Tight Interlacing To illustrate the theory about regular partition we will give an intuitive example. Consider a star with four vertices, see figure 2.

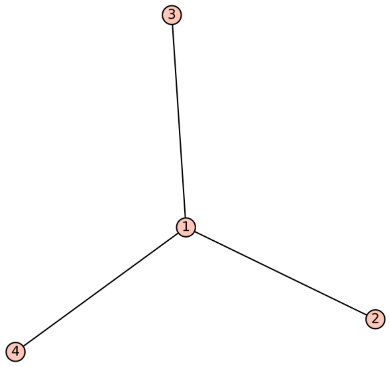


Figure 2: Star with four vertices

The adjacency matrix A of that graph is as follows:

$$A = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

The corresponding eigenvalues to that matrix are $\sqrt{3},\ 0,\ 0,$ and $-\sqrt{3}$.

If we consider the partition of the graph that has vertex 1 in one set and vertices 2, 3, and 4 in the other set, we get a corresponding matrix that indicate which vertices are in which partition, i.e.

$$S = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$$

Note that the columns of S are complements. Eventually, we find that the eigenvalues of the averaged row entries of the matrix $S^TAS = \tilde{B}$, the so-called quotient matrix, coincide with the two extreme eigenvalues of A, i.e. $\sqrt{3}$ and $-\sqrt{3}$. This is called *tight interlacing*. Furthermore, tight interlacing of the extreme eigenvalues of A and $\tilde{B}$ implies a regular partition of matrix A. Potential partitioning criteria are the degree of a vertex and spectral characteristics of the matrix.

2.2 Image segmentation - Preliminaries

Every image consists of pixels and to achieve a good segmentation it is essential to cover the characteristics of those pixels. Our approach is to create a similarity matrix W that summarises the relationship between every pair of pixels. If you model an image as a complete graph, each entry in W can be interpreted as the edge weight. In our case W has values between 0 and 1 and it captures how similar pixels are to each other. W is essential in our theoretical investigations and we work further with it in our algorithms. Note that the dimension of the square matrix W will be xy x xy given that the size of the original image is x x y. This matrix is very sparse and furthermore, we would only need the upper triangle to capture all weights due to symmetry reasons. However, we will stick to that weight matrix for now. Working with sufficiently small images will not create any running time problems.

There are different ways of modelling the relationship and assign weights to the edges. One possibility is to include both a similarity term and a proximity term in the weight of the corresponding edge.

Following Shi and Malik (2000) we decided to use following weight formula that includes the two criteria just mentioned. That is,

$$w_{ij} = e^{\frac{-||F(i)-F(j)||_2^2}{\sigma_I}} * e^{\frac{-||X(i)-X(j)||_2^2}{\sigma_X}} \qquad if\ ||X(i) - X(j)||_2 < r \tag{2}$$

$$w_{ij} = e^{\frac{-||F(i)-F(j)||_2^2}{\sigma_I}} * 0 \qquad otherwise, \tag{3}$$

where the first exponent describes the similarity of two pixels, i.e. colour, and the second exponent describes distance between pixels.

We chose to set σ_I and σ_X equal to one which simplifies the above equations to

$$w_{ij} = e^{-||F(i)-F(j)||_2^2} * e^{-||X(i)-X(j)||_2^2} \qquad if\ ||X(i) - X(j)||_2 < r \tag{4}$$

$$w_{ij} = 0 \qquad otherwise, \tag{5}$$

where r determines how far two pixels can be apart from each other to still be considered as being in the neighbourhood of each other. Note that the weight is zero if two pixels are too far apart. Throughout all our investigations r is set equal to two since we work with comparably small images (up to size 10 x 10). If we chose r to be much larger the entries in the W matrix would be more likely to conclude that two pixels are similar even though they are of opposite colour.

Example: W To clarify the information kept in the weight matrix W consider the following example.

Image 3 is of size 2 x 2 and consist of one black pixel in the top left and three white pixels.

The matrix that is used to construct this image is

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

When R displays a matrix as an image then columns of a matrix correspond to pixel rows of an image. In our example, the first column matches the lower pixel row that is all white, and the second column matches the upper pixel row that has one black and one white pixel. See image 3 below as an illustration.

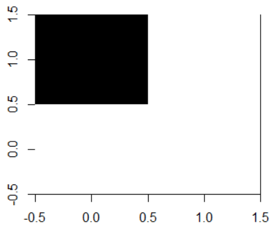


Figure 3: 2x2 image with one black pixel

W of figure 3 is calculated following equations (4) and (3.4) and we obtain

$$W = \begin{bmatrix} 1.0000 & 0.1353 & 0.3679 & 0.1353 \\ 0.1353 & 1.0000 & 0.0498 & 0.1353 \\ 0.3679 & 0.0498 & 1.0000 & 0.3679 \\ 0.1353 & 0.1353 & 0.3679 & 1.0000 \end{bmatrix}.$$

Here, the pixels are enumerated

$$\begin{pmatrix} 2 & 4 \\ 1 & 3 \end{pmatrix}$$

and W presents the similarities between every pair of pixels. One can observe that each entry on the diagonal of W equals one since the distance between the pixel and itself is zero.

Assume we want to segment the image into two parts, such that the black pixel and the white pixels are separated. That is, $\mathcal{P} = \{V_1, V_2\} = \{(2), (1, 3, 4)\}$. The characteristic vector s then is

$$s = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix},$$

and S follows as

$$S = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}.$$

We have seen that tight interlacing of eigenvalues of W and $\tilde{B}$ implies a regular partition of the graph. Furthermore, the goal of image segmentation is to partition an image such that all relevant features are covered by the clusters. We have a strong suspicion that those two approaches are linked, at least for particular types of images. Our hope is to show by means of simulations that a regular partition implies a good image segmentation.

The question arises how to measure whether a partition is close to a regular partition. We will assess a partition in terms of eigenvalues of W and $\tilde{B}$. Our first thought was the tighter the interlacing is, the closer to a regular partition and the better image segmentation we have obtained. However, our notion did not turn out to be supported by our findings. Following our initial idea, the aim is to minimise the differences between the extreme eigenvalues of W and $\tilde{B}$.

Our goal is not to prove that a equitable partition implies a good image segmentation but as a first step to investigate whether or not there is a link between those two concepts.

2.3 Notation

From here on, what is referred to as **A** in Theorem 1 will be W, the edge weight matrix within this thesis. The eigenvalues corresponding to W are denoted as $\lambda_1, ..., \lambda_n$, where n is the total number of pixels of the image. Since we only consider bipartitions, the matrix S is of dimension n x 2. The characteristic vector s is boolean that has entry 1 if the corresponding pixel is in set one of the partition, and 0 otherwise. This s and its complement match the columns of S. The quotient matrix will be referred to as $\tilde{B}$ with corresponding eigenvectors $\mu_1, ..., \mu_m$. Note that we will have μ_1 and μ_2 due to segmenting the image into only two parts.

3 The Approach

In this section we present our approach on how to tackle the investigation. Our aim is to establish a link between image segmentation and equitable partition and we approach this question from two sides simultaneously.

Our analysis has an algorithmic and a theoretical side. We implemented and tested two neighbourhood search algorithms; a local search algorithm and a tabu search algorithm. At the same time we examined theoretical implications and investigated the connection between image segmentation and both tight interlacing and constant row sum.

We coded in RStudio using the programming language R because it offers advantages when working with images. For example, when R reads in an image, it converts it into a matrix where each pixel is represented by an entry of that matrix.

4 Results

In this section we show the findings of our research. As mentioned in the previous section, we split our analysis in algorithmic and theoretical investigations. First, we present our findings to the algorithmic approach, that is the local search and the tabu search algorithm. After that we demonstrate our theoretical results of the eigenvalue criterion and of the constant row sum. It is important to keep in mind that the algorithmic side and the theoretical side were examined simultaneously.

When a matrix is converted to an image in R, the left column of the matrix, from top to bottom, corresponds to the bottom row of pixels in the image, from left to right. The image is filled row by row, bottom to top.

[1,3]	[2,3]	[3,3]
[1,2]	[2,2]	[3,2]
[1,1]	[2,1]	[3,1]

Table 1: Matrix to image enumeration

Now assume our current bipartition is denoted by the characteristic vector s. When s_i equals 1, then pixel i is in set V_1, and in set V_2 otherwise. Each entry in s corresponds to a pixel in the image The enumeration starts with the bottom left pixel of the image, so s_1 shows which set the corner bottom left pixel is in. Enumeration of the pixels is continued column by column, left to right, see below. Hence, if our image consists of 9 pixels, s_9 shows which set the top right corner pixel is in.

3	6	9
2	5	8
1	4	7

Table 2: Enumeration of pixels when compared with s

It is important for our analysis to keep in mind how the pixels are enumerated. The image we investigate later is symmetric so it will lead to less confusion regarding the differences in enumeration.

4.1 Algorithmic Approach: Neighbourhood Search Algorithms

As for the practical approach to our research question we considered two different neighbourhood search algorithms; a local search algorithm and a tabu search algorithm. Both algorithms are similar in that they look for a solution improvement in the neighbourhood of the currently best solution. To give an intuitive explanation of the difference between the two algorithms consider the following graph that displays solution quality. Assume the start solution is at $x = 1$ where x is the horizontal axis in figure 4.

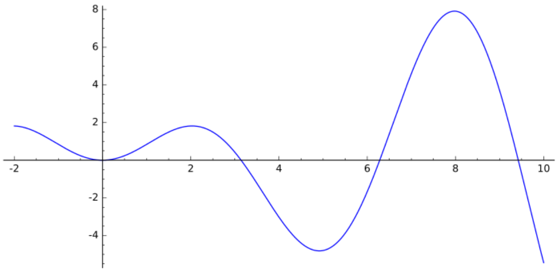


Figure 4: Illustrating neighbourhood search algorithms

The local search algorithm will choose a solution that is on the right of $x = 1$ on the graph and has not come across any downward sloping part of the graph. Thus, local search will terminate at the (local) maximum at $x = 2$.

Tabu search, however, accepts intermediary worse solutions than the current best one and thus will not get stuck in local minima. Taking up the example again, the tabu search algorithm will most likely find the global maximum at $x = 8$. Therefore, tabu search may be our tool of choice after having investigated the results of the local search algorithm. In the following we describe the results obtained with both of the two neighbourhood search algorithms.

Local Search To start off with we used a square version of the Yin and Yang symbol as shown in figure 5.

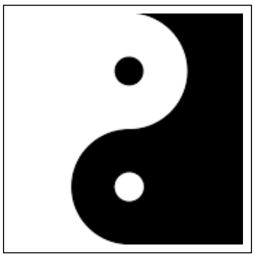


Figure 5: Yin and Yang symbol

To make the analysis less complex we decided to re-size and smoothen the image from 168 x 168 to 100 x 100. Note that the square weight matrix is still of dimension 10,000. Furthermore, averaging the colours of pixels does not only give black and white pixels with corresponding matrix entries 0 and 1 but also grey steps with matrix values between 0 and 1.

After having obtained the weight matrix through pixel similarity and the proximity as shown in equations (4) and (3.4) the aim is to get a bipartition of the pixels through neighbourhood search algorithms such that the image's characteristics are captured well.

First of all, we decided to shrink the image even further to 15 x 15 due to running times of the algorithm, see figure 6.

As mentioned earlier, the local search starts with a start partition, an initial guess, and tries to improve it further by checking neighbourhood partitions. The algorithm

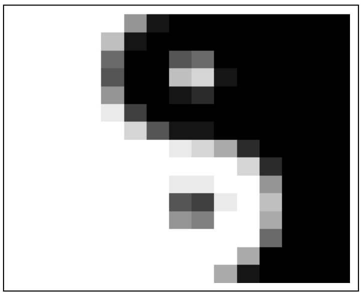


Figure 6: Yin and Yang symbol, 15 x 15

stops once no neighbour candidate yields a better solution than the current one.

As an initial guess we chose to partition the pixels in two halves, i.e. the first 113 are in one and the last 112 pixels are in the other set. This corresponds to one half of the pixels being white and the other half being black. Following our expectations the local search algorithm terminated rather quickly, it only took a few iterations. Figure 7 shows that with this start guess two pixels have swapped the set; the last pixel in the ninth row and the seventh pixel in the ninth row. However, this far away from the original image we started with (see figure 6).

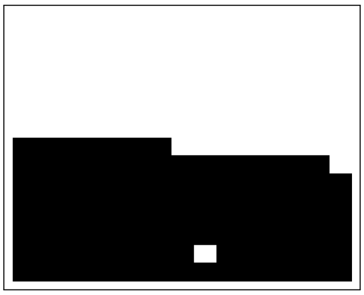


Figure 7: Yin and Yang symbol, 15 x 15, result of local search

Randomised start solution guesses did not yield significantly better results. This lets us conclude that there is lots of room for improvement and that local search is probably not the tool to apply here. We suspect to be stuck in local extrema here, meaning that for the current best partition there is no improvement in its neighbourhood and the algorithm terminates. Thus, we decided to implement an algorithm that can find the best solution and does not get stuck if there is no improvement in the neighbourhood of the current solution.

Tabu Search Since the local search algorithm just described did not give satisfying results we chose to implement and to test the tabu search algorithm. As mentioned earlier, it does not get stuck in local extrema but accepts worse solutions along the way. Thus, we expect the tabu search algorithm to give better results. The way we measure goodness remains as the minimum difference between the extreme eigenvalues of $\tilde{B}$ and W, respectively.

We decided to consider a simpler image that may allow to derive more conclusions from the output due to symmetry reasons. The image we are working with is a centred black square of size 4 x 4 on a white background of size 10 x 10. Thus, the white boundary is of width 3, see figure 8.

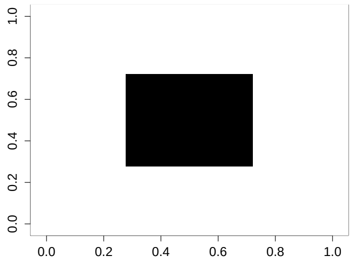
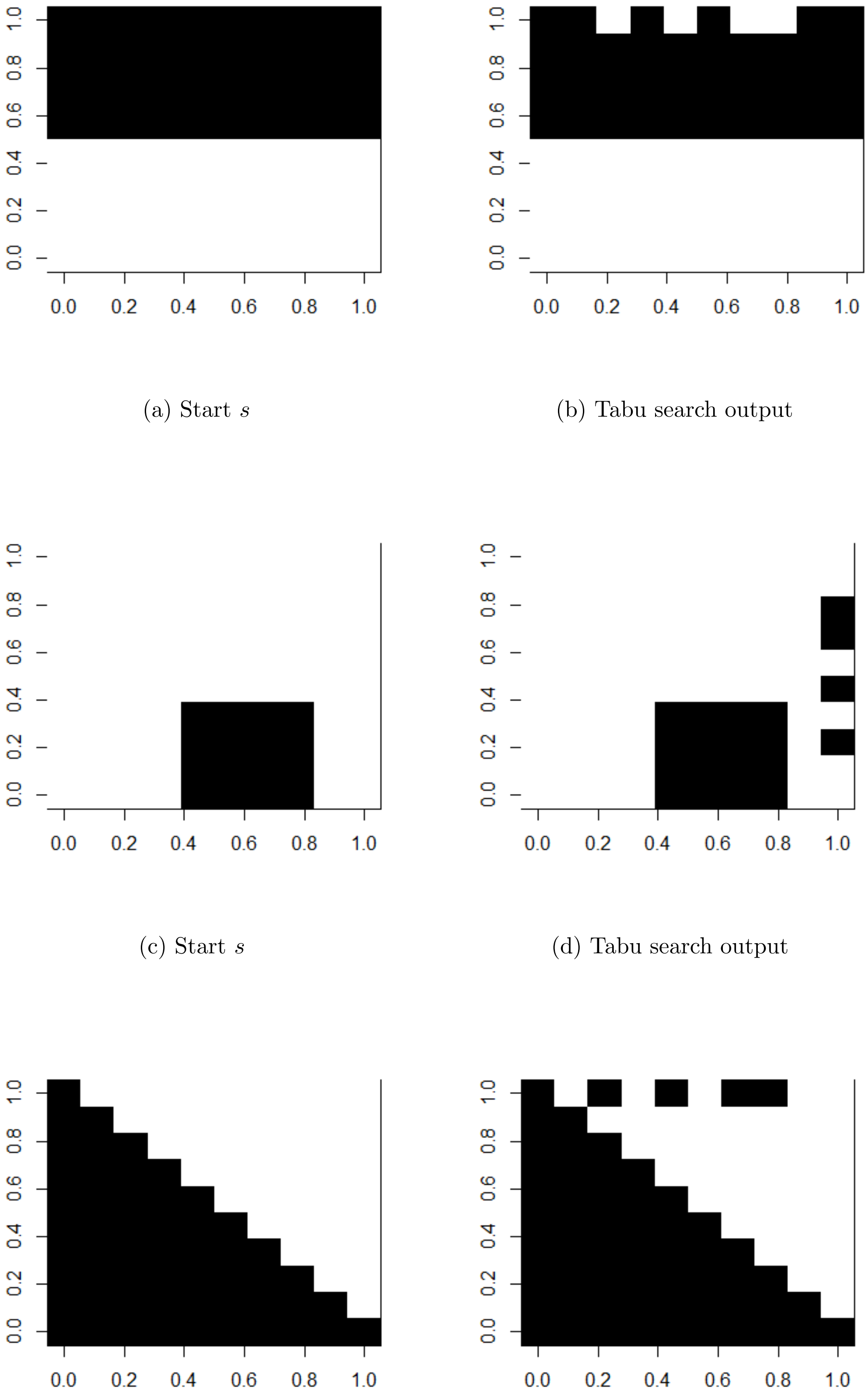


Figure 8: Black Square image

The tabu search algorithm is fairly straightforward. It starts with a start partition, an initial guess, and checks its neighbourhood for a solution improvement. Whichever neighbour is found to be best will be put on the tabu list to prevent it from being chosen again. In our case neighbourhood is defined as swapping the colour of one of the pixels of the partition.

To illustrate, assume we start with a partition that splits the image in half, see figure 9a. Each neighbourhood solution takes the start partition and swaps the colour of one pixel which yields 100 different partitions. The best neighbourhood solution, assuming it's the current best one, is added to the tabu list and will be the new solution of which the neighbours will be investigated. The length of our tabu list was 200 but our number of iterations was 10,000. Thus, as soon as the tabu list was filled, the first entry was dropped and the new candidate was added.

The results of tabu search can be found in figure 9. The start partitions can be found in the left column and the tabu search outputs can be found in the right column.



(a) Start s

(b) Tabu search output

(c) Start s

(d) Tabu search output

(e) Start s

(f) Tabu search output

Figure 9: Output of tabu search

Against our expectations tabu search does not give a visible solution improvement. In fact, the start partitions and the tabu search outputs only differ with respect to a few pixels. Especially the outputs 9d and 9f do not even suggest a visible improvement when compared to their start partitions since those pixels who swapped from white to black are white in the original image, see figure 8.

4.2 Theoretical Approach: In Terms of Eigenvalues

To check empirically in how far tight interlacing and a good partition may coincide we decided to check controlled modifications of the partition in terms of eigenvalues.

We continue working with the Black Square image, see Figure 8. The largest, second largest and smallest eigenvalues of this image's W are $\lambda_1 = 2.7185, \lambda_2 = 2.6564$ and $\lambda_{100} = 0.1225$, respectively.

Remember that the objective is to minimise the distance between λ_1 and μ_1 and between λ_{100} and μ_2. For each modification in figures 10 the colours of two pixels are swapped. Enumeration of pixels in the partition starts in the bottom left corner and goes up column wise, so pixel 100 is in the top right corner and pixel 99 is right underneath pixel 100, compare table 3 below.

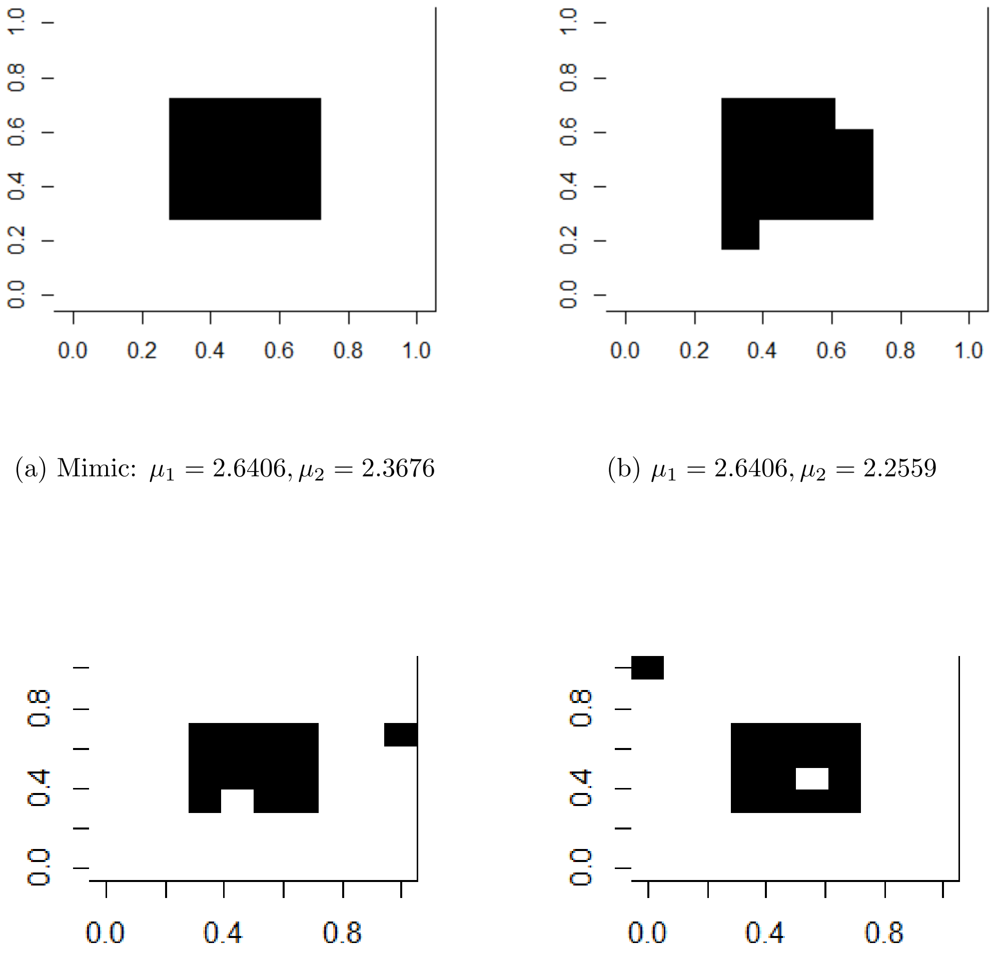
10	20	...	100
...	...	...	...
2	12	...	92
1	11	...	91

Table 3: Enumeration

One can observe that all μ_1 of the modifications are relatively close together when compared with the difference to λ_1. Against our expectation modification 10d is better than the mimic of the original image (10a) in terms of our objective function of tight interlacing. μ_1 barely varies between the different modifications when compared to μ_2. Furthermore, μ_2 is not close to λ_{100} in any of the modifications, and μ_1 turns out to be much closer to $\lambda_2 = 2.6564$ than to λ_1.

To summarise, it seems that tight interlacing does not imply good image segmentation. We suspect the size of the image and the boundary effect when calculating W are the problem here. Furthermore, we may have to reconsider our choice of our objective function in terms of what to minimise and how to compare different partitions. Firstly, we decided to check if we can find constant row sum of the different segments of an image which would also imply a regular partition.

In the next section we will try to show whether minimising the constant row sum of each block of the two partitions implies the best image segmentation.



(a) Mimic: $\mu_1 = 2.6406, \mu_2 = 2.3676$

(b) $\mu_1 = 2.6406, \mu_2 = 2.2559$

(c) $\mu_1 = 2.6407, \mu_2 = 2.1685$

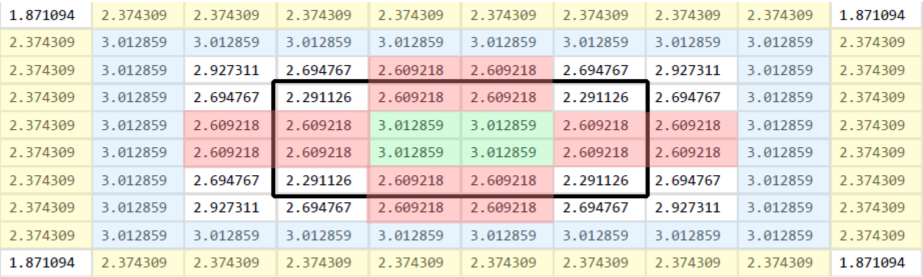
(d) $\mu_1 = 2.6428, \mu_2 = 2.0930$

Figure 10: Modifications of Black Square partitions

4.3 Constant Row Sum

As described in Preliminaries, constant row sums within the blocks that follows from the partition is equivalent to a regular partition. We decided to continue working with the Black Square image, see figure 8.

The row sums that have been obtained from a partition that perfectly mimics the original image can be found in figure 11 below.



1.871094	2.374309	2.374309	2.374309	2.374309	2.374309	2.374309	2.374309	2.374309	1.871094
2.374309	3.012859	3.012859	3.012859	3.012859	3.012859	3.012859	3.012859	3.012859	2.374309
2.374309	3.012859	2.927311	2.694767	2.609218	2.609218	2.694767	2.927311	3.012859	2.374309
2.374309	3.012859	2.694767	2.291126	2.609218	2.609218	2.291126	2.694767	3.012859	2.374309
2.374309	3.012859	2.609218	2.609218	3.012859	3.012859	2.609218	2.609218	3.012859	2.374309
2.374309	3.012859	2.609218	2.609218	3.012859	3.012859	2.609218	2.609218	3.012859	2.374309
2.374309	3.012859	2.694767	2.291126	2.609218	2.609218	2.291126	2.694767	3.012859	2.374309
2.374309	3.012859	2.927311	2.694767	2.609218	2.609218	2.694767	2.927311	3.012859	2.374309
2.374309	3.012859	3.012859	3.012859	3.012859	3.012859	3.012859	3.012859	3.012859	2.374309
1.871094	2.374309	2.374309	2.374309	2.374309	2.374309	2.374309	2.374309	2.374309	1.871094

Figure 11: Row sums of mimic

This matrix consists of the row sums of W of 8. Entry [10,1] of the matrix is the sum of the first row of W, entry [9,1] is the sum of the second row of W, entry [10,2] is the sum of the eleventh row of W, and so on.

The enumeration of pixels in a partition of the original image is as follows:

10	20	...	100
...	...	...	...
2	12	...	92
1	11	...	91

Table 4: Enumeration of s

Each entry in figure 11 of the row sums corresponds to the respective pixel in the original image and the black box indicates where the black square in the original image (see figure 8) is placed.

As we saw earlier, constant row sum within the blocks of a partition is equivalent to an equitable partition. Assume we split the image such that it perfectly mimics the original image, so the black pixels are in first set and the white pixels are in the second set. Considering this partition and its different blocks in

$$W = \begin{pmatrix} W_{1,1} & W_{1,2} \\ W_{2,1} & W_{2,2} \end{pmatrix}$$

then $W_{1,1}$ has dimension 16 x 1 since it consists of the row sums of the black pixels, shown within the black frame in figure 11.

Only looking at this block of the partition we already see that those 16 sums are not constant. Instead, the 2 x 2 green coloured middle square has constant row sum and that the rims suggest constant row sum, too. However, for example the four corner sums that correspond to the four corner pixels, have constant row sum but differ from the rest of the outer rim. This is due to the boundary effect. Depending on the location of the pixel the number of its neighbours differs. For example, pixel 1 has 3 neighbours (2,11,12), pixel 2 has 6 neighbours (1,3,11,12,13), and pixel 12 has 8 neighbours (1,2,3,11,13,21,22,23). The image is not large enough to find more evidence for or against constant row sum in a regular partition.

Another observation is that the minimum eigenvalues of W and $\tilde{B}$ of figure 8 are fairly far apart. Furthermore, the maximum eigenvalue of $\tilde{B}$, $\mu_1 = 2.6406$, lies closer to the second largest eigenvalue of W ($\lambda_2 = 2.6564$) than to the largest one ($\lambda_1 = 2.7185$). Thus, we suspect that the largest and second largest eigenvalue may partly be criteria to achieve tightness.

5 Conclusion and Discussion

Our objective was to find a link between image segmentation and equitable partition. We tackled this idea with two contrary approaches; from a algorithmic and a theoretical side. As for the algorithmic approach, we implemented and tested the local search and the tabu seach algorithm that try to improve a partition by checking its neighbourhood partitions. However, neither the local search algorithm nor the tabu search algorithm gave satisfactory results.

At the same time we checked whether tight interlacing of the eigenvalues of W and $\tilde{B}$ gives us a connection between image segmentation and equitable partition. After having examined the black square modifications we could not conclude that there is a relationship between tight interlacing of eigenvalues and a good image segmentation.

The second theoretical approach we took is looking at the row sums of the blocks of a partition. Constant row sums imply a regular partition. However, the small size of our images and the boundary effect hindered us to show a strong connection between image segmentation and regular partition here.

Shi and Malik (2000) use the eigenvector corresponding to the second smallest eigenvalue to bipartition the graph and recursively bipartition further if desired. This could be one approach for future research to find a good image segmentation algorithm using spectral graph theory. One might also consider to rethink the choice of the weight function since it might not be appropriate for our problem. Furthermore, one could investigate the criterion of constant row sum within segments that also implies regular partition. The images we have been working with were too small to draw conclusions from this criterion due to the boundary effect.

Once a theoretical background has been established that relates regular partitions to image segmentation, the next step would be to find an algorithm that achieves a good image segmentation. Possible extensions could then be to segment an image into more than two parts, or to investigate coloured images.

References

Abiad A. (2012). Some Applications of Linear Algebra in Spectral Graph Theory. *Departament de Matemàtica Aplicada IV*. Master of Science in Advanced Mathematics and Mathematical Engineering. Universitat Politècnica de Catalunya.

Casaca W., Paiva A., Gomez-Nieto E., Joia P, Nonato L.G. (2013). Spectral Image Segmentation using Image Decomposition and Inner Product-based metric. *Journal of Mathematical Imaging and Vision*, 45(3), 227-238.

Cour T., Bénézit F., Shi J. (2005). Spectral Segmentation with Multiscale Graph Decomposition. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2, 1124-1131.

Shi J. and Malik J., Member, IEEE. (2000). Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), 888-905.