Direct Estimation of Appearance Models for Segmentation[*]

Jeova F. S. Rocha Neto[†]   Pedro Felzenszwalb[†]   Marilyn Vazquez[‡]

September 15, 2021

Abstract

Image segmentation algorithms often depend on appearance models that characterize the distribution of pixel values in different image regions. We describe a new approach for estimating appearance models directly from an image, without explicit consideration of the pixels that make up each region. Our approach is based on novel algebraic expressions that relate local image statistics to the appearance of spatially coherent regions. We describe two algorithms that can use the aforementioned algebraic expressions to estimate appearance models directly from an image. The first algorithm solves a system of linear and quadratic equations using a least squares formulation. The second algorithm is a spectral method based on an eigenvector computation. We present experimental results that demonstrate the proposed methods work well in practice and lead to effective image segmentation algorithms.

Keywords: Image segmentation, Mixture models, Markov Random Fields
MSC: 68U10, 62M05, 62H30, 65C20

1 Introduction

Image segmentation is a fundamental problem in computer vision and image processing. The goal of segmentation is to partition an image into regions corresponding to different objects or materials that are visible in a scene. There are numerous applications of automatic segmentation methods including in medical image analysis, remote sensing, industrial inspection, object recognition and interactive image editing. Therefore, advances in image segmentation can have significant impact in many applications.

Segmentation algorithms often rely on some type of appearance modeling to be able to classify each pixel in the image into different regions. Many segmentation algorithms are based on minimization of functionals that encourage spatial coherence of regions and a fit of the pixels in each region to an appearance model.

In some settings such as medical image analysis or remote sensing the appearance of different objects of interest may be known in advance. On the other hand, in many applications the appearance of the objects or materials in a scene need to determined during the segmentation process.

When appearance models are not known in advance, the challenge of image segmentation is that it is essentially a "chicken-and-egg" problem. If we had appearance models we could segment the image. On the other hand, if we had a segmentation we could estimate the appearance of each region. Some methods such as [14] and [16] alternate between estimating appearance models and estimating a segmentation in an iterative fashion. Similarly, variational methods such as [5] and [13] repeatedly evolve a segmentation and the appearance model of each region. Less frequently, methods such as [18] and [17] tackle the segmentation problem by making the dependency on appearance models implicit. Other methods such as the graph-based approaches in [15] and [6] avoid using appearance models and instead use pairwise similarity to cluster pixels.

[†]School of Engineering, Brown University, Providence, RI, USA
[‡]Mathematical Bioscience Institute, Ohio State University, Columbus, OH, USA

In this paper we describe a novel approach for estimating appearance models directly from an image, without explicit consideration of the pixels that make up each region. The approach relies on the spatial coherence of regions but does not require solving for a segmentation to determine the appearance of each region. Instead we derive a set of algebraic expressions that can be used to solve for the appearance models of each region using local statistics that can be easily estimated directly from an image.

We focus specifically on the problem of *binary* segmentation where the goal is to partition an image into two regions (often the foreground and background). Our results allow for the development of new segmentation algorithms that use a two-step process. First we determine appearance models without segmenting the image. We can then segment the image using any available method that requires appearance models to be fixed in advance. In practice we propose to use the estimated appearance models to define a Markov Random Field [2], and use efficient graph cut algorithms to segment the image [8, 3]. This two-step approach leads to efficient methods for unsupervised segmentation.

Many image segmentation algorithms pre-process an image using the response of Gabor filters to define features [9, 11, 19]. Although very useful for capturing texture information, the use of filter banks requires significant processing and memory overhead. Moreover, the use of filter responses can lead to inaccurate localization of region boundaries, due to the non-trivial spatial extent of texture filters.

Recent segmentation methods have also use machine learning techniques to define local features or in complete end-to-end systems (see, e.g. [12]). Machine learning methods typically require labeled datasets for training and often do not generalize across different imaging domains. The resulting systems can also suffer from limited interpretability. In contrast, our model based approach is easily interpretable and can be applied to different types of images without requiring any training data.

The algorithms we describe in this paper do not rely on filter responses or learned features. Instead we work directly with raw pixel values. Our methods estimate non-parametric models that represent the appearance of regions in terms of distributions over pixel values. This makes it possible to accurately localize the boundaries between regions with complex appearances.

The remainder of the paper is organized as follows. In Section 2 we describe how we represent the appearance of regions using non-parametric models that capture the distribution of pixel values within a region. In Section 3 we discuss how local image statistics can be related to the appearance models of images with two regions using a set of algebraic constraints. These algebraic constraints lead to two different methods for estimating appearance models described in Section 4. In Section 5, we show experimental results and evaluate both the accuracy of the estimated appearance models and the segmentations obtained using these models. Finally, Section 6 concludes the paper with a summary and brief discussion.

2   Appearance Models for Segmentation

Let $I : \Omega \to L$ be an image, where Ω is a set of pixel locations and L is a finite set of pixel values. For an n by m graylevel image we have $\Omega = \{1, \ldots, n\} \times \{1, \ldots, m\}$ while $L = \{0, \ldots, 255\}$. In Section 5.6, we will discuss how to generalize our methods to RGB and vector valued images. We use $I(x)$ to denote the value of a pixel $x \in \Omega$. A region $\mathcal{R}$ is a subset of the pixels in Ω.

We define an *appearance model* $\theta \in \mathbb{R}^L$ to be a distribution over L. An appearance model for a region $\mathcal{R}$ specifies the typical values for the pixels in $\mathcal{R}$. Note, however, that an appearance model θ does not fully specify the joint distribution of the pixels in a region, only the probabilities of observing different pixel values. We do not assume the pixel values in each region are independent. Therefore the appearance models considered here only define a coarse representation for the appearance of a region. For example, if we permute the pixels within a region the appearance model would remain the same.

We assume the image domain Ω can be divided into two regions $\mathcal{R}_0$ and $\mathcal{R}_1$ and use $S : \Omega \to \{0, 1\}$ to denote a binary image specifying an assignment of image pixels into regions, or a *segmentation*. Let θ_0 and θ_1 be appearance models for $\mathcal{R}_0$ and $\mathcal{R}_1$ respectively.

As discussed in the introduction, one of the fundamental challenges of unsupervised image segmentation is that it is essentially a "chicken-and-egg" problem. If we have appearance models θ_0 and θ_1 we can partition the image into regions using several existing approaches. On the other hand, if we have a partition of an

image into regions $\mathcal{R}_0$ and $\mathcal{R}_1$ we can estimate appearance models by computing normalized histograms of the pixel values in each region.

The main contribution of our work is a method for estimating the appearance models θ_0 and θ_0 *directly* from an image, without explicit consideration of the regions $\mathcal{R}_0$ and $\mathcal{R}_1$.

Once appearance models are estimated there are many methods that can be used to partition an image. In the simplest setting we can classify each pixel independently using a likelihood ratio defined by θ_0 and θ_1. Alternatively, methods based on Markov Random Fields ([2]) segment an image by minimizing an energy that combines the appearance models with a boundary regularization term,

$$E(S|\lambda, \theta_0, \theta_1) = -\sum_{x \in \Omega} \ln \theta_{S(x)}(I(x)) + \lambda \sum_{x,y \text{ neighbors}} |S(x) - S(y)|. \tag{1}$$

In this case the minimum energy segmentation corresponds to a MAP estimate of S, where the boundary regularization defines a prior over segmentations. Importantly, the energy function E can be efficiently minimized by the computation of a minimum cut in a graph ([8]).

3   Statistical Model

In order to address the appearance model estimation problem, we treat the image $I : \Omega \to L$ as a realization of a random field. Let $\mathcal{R}_0$ and $\mathcal{R}_1$ be a partition of Ω into two regions. Let θ_0 and θ_1 be appearance models associated with $\mathcal{R}_0$ and $\mathcal{R}_1$ respectively.

Our approach relies on two key assumptions. The first assumption is that regions have homogeneous appearance in the following sense.

Assumption 1 (Homogeneity). *The probability that a pixel x takes a particular value depends only on the region the pixel belongs to,*
$$P(I(x) = i \,|\, x \in \mathcal{R}_s) = \theta_s(i).$$

Note that this assumption does not specify a full generative model for the image. We assume the pixels in each region have the same marginal distribution, but their joint distribution could involve dependencies, as we see for example in images with textures. Similar assumptions have been used in other approaches for image segmentation, including several methods for unsupervised texture segmentation [13, 19, 11].

In general we expect the values of nearby pixels to be dependent. For example, neighboring pixels that are in the same region almost always have similar values. The second assumption we make captures the idea that sufficiently far away pixels are independent.

Assumption 2 (Independence at a distance). *If x and y are two pixels at a distance r from each other,*

$$P(I(x) = i, I(y) = j \,|\, x \in \mathcal{R}_s, y \in \mathcal{R}_t) = P(I(x) = i \,|\, x \in \mathcal{R}_s)P(I(y) = j \,|\, y \in \mathcal{R}_t).$$

To estimate θ_0 and θ_1 we consider two distributions α and β that can be directly estimated from an observed image. We will relate θ_0 and θ_1 to α and β using the assumptions above.

Let $x, y \in \Omega$ be a pair of pixels at a distance r from each other, selected uniformly at random. Since the pixels are in a discrete grid we use the L1 norm to measure the distance between them. Let $\alpha \in \mathbb{R}^L$ be a distribution over L where $\alpha(i)$ is the probability that pixel x has value i. Let $\beta \in \mathbb{R}^{L \times L}$ be a distribution over $L \times L$ where $\beta(i, j)$ is the probability that pixel x has value i and pixel y has value j.

$$Q = \{(x, y) \in \Omega^2 \,|\, ||x - y|| = r\}.$$
$$(x, y) \sim \text{Uniform}(Q).$$
$$\alpha(i) = P(I(x) = i).$$
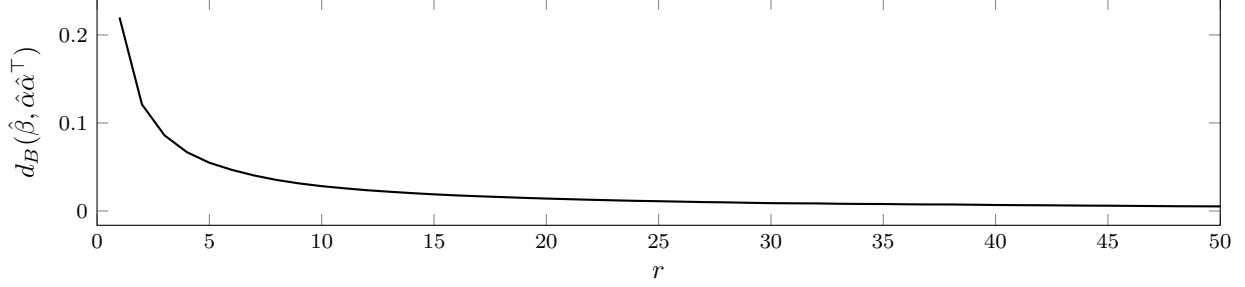$$\beta(i, j) = P(I(x) = i, I(y) = j).$$

3

Figure 1: Evaluating independence at a distance for images with a single Brodatz texture (see Figure 5 for examples of the images in this dataset). We show the average Bhattacharyya distance between $\hat{\beta}$ and $\hat{\alpha}\hat{\alpha}^\top$ as a function of r.

Note that we can easily estimate α and β from an observed image. We simply enumerate all pairs of pixels at distance r from each other and count the number of times we observe pixels with particular values. If the image is large we can also estimate the two distributions using a random sample of pairs of pixels at distance r. We use $\hat{\alpha}$ and $\hat{\beta}$ to denote the estimates of α and β computed from an observed image.

$$\hat{\alpha}(i) = \frac{1}{|Q|} \sum_{(x,y) \in Q} \mathbb{1}(I(x) = i).$$

$$\hat{\beta}(i,j) = \frac{1}{|Q|} \sum_{(x,y) \in Q} \mathbb{1}(I(x) = i)\mathbb{1}(I(y) = j).$$

For images with a single region Assumption 2 leads to $\beta = \alpha\alpha^\top$. Figure 1 evaluates this identity in textured images from the Brodatz dataset [4]. See Figure 5 for examples of the images in the Brodatz dataset. In this case each image has a single textured region and we compare $\hat{\beta}$ to $\hat{\alpha}\hat{\alpha}^\top$ for different values of r using the Bhattacharyya distance (see Section 5.1). When r is small we see that the two distributions, $\hat{\beta}$ and $\hat{\alpha}\hat{\alpha}^\top$, are quite different, because nearby pixels are not independent. As we increase r we see that $\hat{\beta}$ is close to $\hat{\alpha}\hat{\alpha}^\top$, suggesting that pixels that are relatively far from each other are independent.

Let $w_0 = P(x \in \mathcal{R}_0)$ and $w_1 = P(x \in \mathcal{R}_1)$. Now consider the probability that x and y are in different regions. Let $\epsilon = P(x \in \mathcal{R}_0, y \in \mathcal{R}_1) = P(x \in \mathcal{R}_1, y \in R_0)$. The probability that x and y are in different regions is 2ϵ.

The following proposition derives a set of linear and quadratic algebraic constraints between α, β, w_0, w_1, θ_0, θ_1 and ϵ. These constraints will enable us to estimate θ_0 and θ_1 without explicit consideration of $\mathcal{R}_0$ and $\mathcal{R}_1$.

Proposition 1. *Under Assumption 1 and Assumption 2 we have:*

$$\alpha = w_0\theta_0 + w_1\theta_1. \tag{2}$$

$$\beta = (w_0 - \epsilon)\theta_0\theta_0^\top + (w_1 - \epsilon)\theta_1\theta_1^\top + \epsilon\theta_0\theta_1^\top + \epsilon\theta_1\theta_0^\top. \tag{3}$$

Proof. The result follows from the law of total probability. For (2),

$$
\begin{aligned}
\alpha(i) &= P(I(x) = i) \\
&= P(x \in \mathcal{R}_0)P(I(x) = i \mid x \in \mathcal{R}_0) + P(x \in \mathcal{R}_1)P(I(x) = i \mid x \in \mathcal{R}_1) \\
&= w_0\theta_0(i) + w_1\theta_1(i).
\end{aligned}
$$

For (3), first note that

$$P(x \in \mathcal{R}_0) = P(x \in \mathcal{R}_0, y \in \mathcal{R}_0) + P(x \in \mathcal{R}_0, y \in \mathcal{R}_1).$$

4

Therefore $P(x \in \mathcal{R}_0, y \in \mathcal{R}_0) = w_0 - \epsilon$. Similarly $P(x \in \mathcal{R}_1, y \in \mathcal{R}_1) = w_1 - \epsilon$. Now,

$$\begin{aligned}
\beta(i,j) &= P(I(x) = i, I(y) = j) \\
&= P(x \in \mathcal{R}_0, y \in \mathcal{R}_0) P(I(x) = i, I(y) = j \mid x \in \mathcal{R}_0, y \in \mathcal{R}_0) + \\
&\quad P(x \in \mathcal{R}_1, y \in \mathcal{R}_1) P(I(x) = i, I(y) = j \mid x \in \mathcal{R}_1, y \in \mathcal{R}_1) + \\
&\quad P(x \in \mathcal{R}_0, y \in \mathcal{R}_1) P(I(x) = i, I(y) = j \mid x \in \mathcal{R}_0, y \in \mathcal{R}_1) + \\
&\quad P(x \in \mathcal{R}_1, y \in \mathcal{R}_0) P(I(x) = i, I(y) = j \mid x \in \mathcal{R}_1, y \in \mathcal{R}_0) \\
&= (w_0 - \epsilon)\theta_0(i)\theta_0(j) + (w_1 - \epsilon)\theta_1(i)\theta_1(j) + \epsilon\theta_0(i)\theta_1(j) + \epsilon\theta_1(i)\theta_0(j).
\end{aligned}$$

$\square$

3.1 Extension to images with m regions

It is possible to generalize the algebraic constraints in Proposition 1 to the case of images with m regions. Let $\mathcal{R}_0, \ldots, \mathcal{R}_{m-1}$ be a partition of Ω into m regions. In this case we have parameters θ_i, w_i and ϵ_{ij} for $0 \le i, j \le m - 1$ and $j \neq i$. Here θ_i is the appearance model associated with $\mathcal{R}_i$ while $w_i = P(x \in \mathcal{R}_i)$ and $\epsilon_{ij} = P(x \in \mathcal{R}_i, y \in \mathcal{R}_j)$. Since x and y are exchangeable $\epsilon_{ij} = \epsilon_{ji}$.

Using the law of total probability we can derive the following algebraic expressions,

$$\alpha = \sum_{i=0}^{m-1} w_i \theta_i, \tag{4}$$

$$\beta = \sum_{i=0}^{m-1} \left((w_i - \sum_{j \neq i} \epsilon_{ij}) \theta_i \theta_i^\top + \sum_{j \neq i} \epsilon_{ij} \theta_i \theta_j^\top \right). \tag{5}$$

We note, however, that the algorithms we develop to estimate appearance models when $m = 2$ do not readily generalize to the case when $m > 2$. For the remainder of the paper we consider only the case of images with 2 regions.

4 Appearance Estimation

As discussed in the previous section we can estimate α and β directly from an image. In this section we show how we can recover θ_0 and θ_1 from α and β or their estimates $\hat{\alpha}$ and $\hat{\beta}$. Initially we assume that w_0, w_1 and ϵ are known, and present two different methods (Sections 4.1 and 4.2) for estimating θ_0 and θ_1. We consider the case where w_0, w_1 and ϵ are unknown in Section 4.3.

Our methods require that $w_0 w_1 \neq \epsilon$. Proposition 2 in Section 4.2 shows that if $w_0 w_1 = \epsilon$ then $\beta = \alpha\alpha^\top$. In this case we do not have enough information to recover the appearance models. In general we expect that $w_0 w_1 > \epsilon$ as long r is not too large. This condition imposes certain constraints of the shape and size of the regions $\mathcal{R}_0$ and $\mathcal{R}_1$ relative to the value of r.

4.1 Algebraic Method

Suppose α, β, w_0, w_1 and ϵ are known, and consider the problem of recovering θ_0 and θ_1.

Let $L = \{1, \ldots, k\}$. We have $2k$ unknowns $\theta_0(1), \ldots, \theta_0(k)$ and $\theta_1(1), \ldots, \theta_1(k)$. Proposition 1 defines k linear and k^2 quadratic constraints,

$$\alpha(i) = w_0\theta_0(i) + w_1\theta_1(i), \tag{6}$$

$$\beta(i,j) = (w_0 - \epsilon)\theta_0(i)\theta_0(j) + (w_1 - \epsilon)\theta_1(i)\theta_1(j) + \epsilon\theta_0(i)\theta_1(j) + \epsilon\theta_1(i)\theta_0(j). \tag{7}$$

Since θ_0 and θ_1 define probability distributions we also have the additional constraints that both vectors should sum to one and $\theta_0(i) \ge 0$, $\theta_1(i) \ge 0$ for $1 \le i \le k$.

4.1.1 Minimal constraints

We first consider a simple method that uses a subset of the constraints to solve for all of the unknowns in the appearance models. The approach uses the k linear constraints defined by α and k quadratic constraints defined by a single row of β. For the derivation below we treat α and β as known although in practice we only have empirical estimates of the two distributions.

1. Let $i \in L$. Using the quadratic constraint defined by $\beta(i, i)$ and the linear constraint defined by $\alpha(i)$ we can solve for $\theta_0(i)$ and $\theta_1(i)$,

$$\theta_0(i) = \alpha(i) \pm \frac{\sqrt{\frac{w_0 w_1 - \epsilon}{\omega_1^2}(\beta(i, i) - \alpha^2(i))}}{\frac{w_0 w_1 - \epsilon}{w_1^2}}, \ \theta_1(i) = \frac{\alpha(i) - w_0 \theta_0(i)}{w_1}. \tag{8}$$

2. Now consider each $j \in L$ with $j \neq i$. Since we solved for $\theta_0(i)$ and $\theta_1(i)$ in Step 1, now $\beta(i, j)$ defines a linear constraint on $\theta_0(j)$ and $\theta_1(j)$. Together with the linear constraint defined by $\alpha(j)$ we can solve for $\theta_0(j)$ and $\theta_1(j)$,

$$\theta_0(j) = \frac{w_1 \beta(i, j) - \alpha(j)(w_1 \theta_1(i) + \epsilon(\theta_0(i) - \theta_1(i)))}{(w_0 w_1 - \epsilon)(\theta_0(i) - \theta_1(i))}, \ \ \theta_1(j) = \frac{\alpha(j) - w_0 \theta_0(j)}{w_1}. \tag{9}$$

When $\beta(i, i) = \alpha^2(i)$ we have $\theta_0(i) = \theta_1(i)$. To avoid dividing by zero when solving for $\theta_0(j)$ in Step 2 and to increase the robustness of the method, we can select i maximizing $\beta(i, i) - \alpha^2(i)$ in Step 1. Note that if $\beta(i, i) - \alpha^2(i) = 0$ for all i then $\theta_0 = \theta_1 = \alpha$.

To solve for $\theta_0(i)$ in Step 1 we require that $(w_0 w_1 - \epsilon)(\beta(i, i) - \alpha^2(i)) \geq 0$. Proposition 2 below shows that under the assumptions we have made

$$\beta(i, i) - \alpha^2(i) = (w_0 w_1 - \epsilon)(\theta_0(i) - \theta_1(i))^2. \tag{10}$$

Therefore $(w_0 w_1 - \epsilon)(\beta(i, i) - \alpha^2(i)) \geq 0$.

4.1.2 Least squares solution

The approach described above uses a small number of the constraints defined by α and β to exactly recover θ_0 and θ_1. However, in practice we only have empirical estimates of α and β. We also don't expect real data to perfectly fit our assumptions. We now describe an alternative method that uses all of the constraints defined by α and β in a least squares formulation.

Let $i_1, \ldots, i_k$ be an ordering of $L = \{1, \ldots, k\}$. Our empirical results show that ordering the indices in decreasing value of $\hat{\beta}(i, i) - \hat{\alpha}^2(i)$ works well and is better than a random order.

1. We start by solving for $\theta_0(i_1)$ and $\theta_1(i_1)$ using the quadratic constraint defined by $\hat{\beta}(i_1, i_1)$ and the linear constraint defined by $\hat{\alpha}(i_1)$.

$$\theta_0(i_1) = \hat{\alpha}(i) \pm \frac{\sqrt{\frac{w_0 w_1 - \epsilon}{\omega_1^2}(\hat{\beta}(i_1, i_1) - \hat{\alpha}^2(i_1))}}{\frac{w_0 w_1 - \epsilon}{w_1^2}}, \ \theta_1(i_1) = \frac{\hat{\alpha}(i_1) - w_0 \theta_0(i_1)}{w_1}. \tag{11}$$

2. We iterate ℓ from 2 to k and solve for $\theta_0(i_\ell)$ and $\theta_1(i_\ell)$ in each step. When solving for $\theta_0(i_\ell)$ and $\theta_1(i_\ell)$ we already have values for $\theta_0(i_1), \ldots, \theta_0(i_{\ell-1})$ and $\theta_1(i_1), \ldots, \theta_1(i_{\ell-1})$. Therefore $\hat{\beta}(i_1, i_\ell), \ldots, \hat{\beta}(i_{\ell-1}, i_\ell)$ define $\ell - 1$ linear constraints on $\theta_0(i_\ell)$ and $\theta_1(i_\ell)$. Together with the constraint defined by $\hat{\alpha}(i_\ell)$ we form a system with ℓ linear equations and 2 unknowns that can be solved using linear least squares:

$$\forall j \in \{i_1, \ldots, i_{\ell-1}\} \ ((w_0 - \epsilon)\theta_0(j) + \epsilon\theta_1(j))\theta_0(i_\ell) + ((w_1 - \epsilon)\theta_1(j) + \epsilon\theta_0(j))\theta_1(i_\ell) = \hat{\beta}(j, i_\ell),$$

$$w_0 \theta_0(i_\ell) + w_1 \theta_1(i_\ell) = \hat{\alpha}(i_\ell).$$

3. We improve our estimates by iterating ℓ from 1 to k and re-estimate $\theta_0(i_\ell)$ and $\theta_1(i_\ell)$ in each step. To re-estimate $\theta_0(i_\ell)$ and $\theta_1(i_\ell)$ we use $\hat{\beta}(j, i_\ell)$ and the current values for $\theta_0(j)$ and $\theta_1(j)$ for $j \neq i_\ell$ to define $k-1$ linear constraints. Together with the constraint defined by $\hat{\alpha}(i_\ell)$ we form a system with k linear equations and 2 unknowns that can be solved using linear least squares:

$$\forall j \neq i_\ell \;\; ((w_0 - \epsilon)\theta_0(j) + \epsilon\theta_1(j))\theta_0(i_\ell) + ((w_1 - \epsilon)\theta_1(j) + \epsilon\theta_0(j))\theta_1(i_\ell) = \hat{\beta}(j, i_\ell)$$

$$w_0\theta_0(i_\ell) + w_1\theta_1(i_\ell) = \hat{\alpha}(i_\ell)$$

Empirically we found that iterating over the entries one time using this method is enough to obtain improved results.

4. We set $\theta_s(i) = \max(\theta_s(i), 0)$ and normalize θ_0 and θ_1 to add up to one. This ensures θ_0 and θ_1 define valid probability distributions.

Note that there are two choices for the value of $\theta_0(i_1)$ when solving the quadratic equation in Step 1. We consider both choices to estimate full appearance models. We then compare β defined by the estimated models and Equation (3) to $\hat{\beta}$ using the Bhattacharyya distance. We select the appearance models leading to the smaller Bhattacharyya distance.

4.2 Spectral Method

Now we describe a spectral method for estimating the appearance models. As in the previous section we assume the values of w_0, w_1 and ϵ are known. The following proposition provides the basis for the approach.

Proposition 2.
$$\beta - \alpha\alpha^\top = (w_0 w_1 - \epsilon)\,(\theta_0 - \theta_1)\,(\theta_0 - \theta_1)^\top \tag{12}$$

Proof. First note that $w_0 + w_1 = 1$ implies $w_0 w_1 = w_0 - w_0^2$ and $w_0 w_1 = w_1 - w_1^2$.

$$\begin{aligned}
\beta - \alpha\alpha^\top &= (w_0 - \epsilon)\theta_0\theta_0^\top + (w_1 - \epsilon)\theta_1\theta_1^\top + \epsilon\theta_0\theta_1^\top + \epsilon\theta_1\theta_0^\top - (w_0\theta_0 + w_1\theta_1)(w_0\theta_0 + w_1\theta_1)^\top \\
&= (w_0 - \epsilon - w_0^2)\theta_0\theta_0^\top + (w_1 - \epsilon - w_1^2)\theta_1\theta_1^\top + (\epsilon - w_0 w_1)\theta_0\theta_1^\top + (\epsilon - w_1 w_0)\theta_1\theta_0^\top \\
&= (w_0 w_1 - \epsilon)\theta_0\theta_0^\top + (w_0 w_1 - \epsilon)\theta_1\theta_1^\top - (w_0 w_1 - \epsilon)\theta_0\theta_1^\top - (w_1 w_0 - \epsilon)\theta_1\theta_0^\top \\
&= (w_0 w_1 - \epsilon)(\theta_0 - \theta_1)(\theta_0 - \theta_1)^\top
\end{aligned}$$

$\square$

Let $u = \theta_0 - \theta_1$. The above result shows that that matrix $\beta - \alpha\alpha^\top$ is of rank one and its only eigenvector with non-zero eigenvalue is proportional to u. Moreover, the corresponding eigenvalue is $(w_0 w_1 - \epsilon)\|u\|^2$.

The matrix $\hat{\beta} - \hat{\alpha}\hat{\alpha}^\top$ defines an approximation to $\beta - \alpha\alpha^\top$ that can be computed from an image. Let v be the dominant eigenvector of $\hat{\beta} - \hat{\alpha}\hat{\alpha}^\top$ normalized so that $\|v\| = 1$. The vector v gives us an estimate of $u/\|u\|$ up to a sign ambiguity. The corresponding eigenvalue λ can be used to approximate $(w_0 w_1 - \epsilon)\|u\|^2$. We can estimate u as,

$$\hat{u} = \pm\sqrt{\frac{\lambda}{w_0 w_1 - \epsilon}}\, v \tag{13}$$

We can then use $\hat{\alpha}$ and Equation (2) to estimate θ_0 and θ_1,

$$\begin{aligned}
\theta_0 &= \hat{\alpha} + w_1\hat{u}, \\
\theta_1 &= \hat{\alpha} - w_0\hat{u}.
\end{aligned}$$

Finally, we set $\theta_s(i) = \max(\theta_s(i), 0)$ and normalize θ_0 and θ_1 to add up to one. This ensures θ_0 and θ_1 define valid probability distributions.
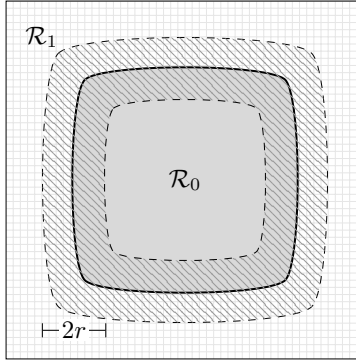
Figure 2: The area where pairs of pixels with $||x - y|| = r$ can be in different regions.

To handle the sign ambiguity in Equation (13) we consider both choices to estimate θ_0 and θ_1. We then compare β defined by the estimated appearance models and Equation (3) to $\hat{\beta}$ using the Bhattacharyya distance. We select the choice of sign in Equation (13) leading to the smaller Bhattacharyya distance.

Notice that we can use simple power iteration methods to compute v and λ. The algorithm complexity depends on $|L|$, which is usually much smaller that the number of pixels in the image. The rate of convergence of power iteration depends on the spectral gap, which is proportional to $w_0 w_1 - \epsilon$.

4.3   Estimating w_0, w_1 and ϵ

The methods described above assume w_0, w_1 and ϵ are known. We have experimented with two different approaches for estimating the appearance models when w_0, w_1 and ϵ are unknown. The first approach simply selects a typical, or average, value for each of the unknown parameters. The second approach involves an explicit search over a discretized set of choices for the unknown parameters.

4.3.1   Typical values

A simple approximation for w_0 and w_1 involves setting $w_0 = w_1 = 0.5$. Although this is a crude approximation we have found that it leads to good appearance models in a wide variety of images.

In order to approximate ϵ, we start from the assumption that the ground truth segmentation S is spatially coherent and that the boundary between regions ∂S is short, i.e., $|\partial S| \approx \sqrt{|\Omega|}$. In this case ϵ is proportional to the area within distance r from ∂S divided by $|\Omega|$, see Figure 2. In our experiments we set $r = \rho \sqrt{|\Omega|}$, where ρ is a parameter set by the user. This makes the selection of the distance r be adaptive to the image resolution. Our estimate of ϵ then becomes,

$$\epsilon = \kappa \frac{r \sqrt{|\Omega|}}{|\Omega|} = \kappa \rho. \tag{14}$$

We have found that setting $\kappa = 0.5$ often leads to good results in practice.

4.3.2   Searching over w_0, w_1 and ϵ

To search over w_0 and w_1 we use the fact that $w_0 + w_1 = 1$ and search over possible values for w_0. We assume without loss of generality that $w_0 \le w_1$ and $w_0 \in (0, 0.5)$. In practice we discretize the interval $(0, 0.5)$ using a step of size of 0.05, leading to 11 choices for w_0. To estimate ϵ we search over the interval $(0, 0.1)$ using a step size of 0.01, leading to 11 choices for ϵ. Together this leads to 121 combined choices for w_0 and ϵ.

For each choice of parameters w_0, w_1 and ϵ we estimate θ_0 and θ_1 using either the algebraic or spectral method above. We then compare β defined by the estimated appearance models and Equation (3) to the

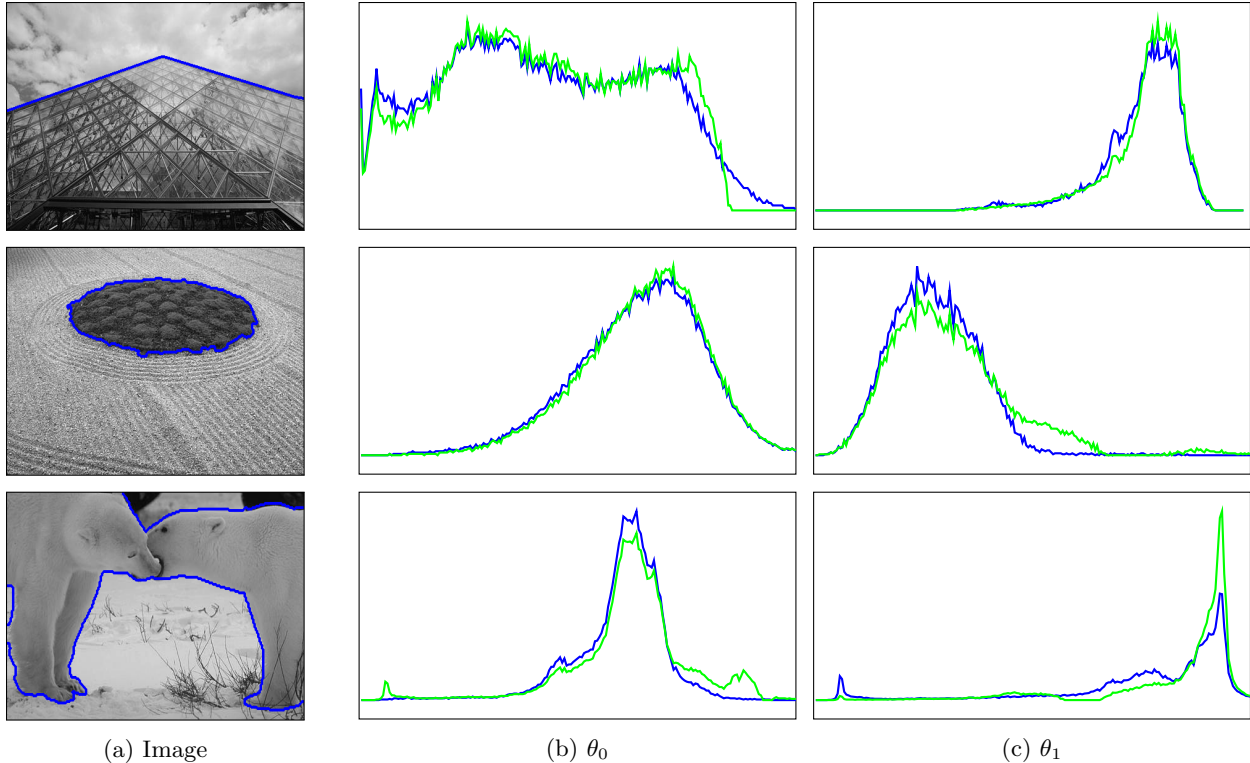(a) Image   (b) θ_0 (c) θ_1

Figure 3: Estimation of appearance models with the algebraic method. In (a) we show the input images and their ground truth segmentation. In (b) and (c) we show the appearance models computed using the ground truth segmentation in blue (——) and the algebraic method in green (——). The images are from the Berkeley Segmentation [10] dataset.

empirical $\hat{\beta}$ computed from the image. We select the model parameters minimizing the Bhattacharyya distance between β and $\hat{\beta}$.

Searching over w_0, w_1 and ϵ with the spectral method is fairly efficient because the bottleneck in the spectral method is computing the dominant eigenvector of $\hat{\beta} - \hat{\alpha}\hat{\alpha}^\top$. Since this matrix does not depend on the unknown parameters the eigenvector only has to be computed once. In this case searching for the parameters leads to limited overhead. Searching for the parameters with the algebraic method is much less efficient. The experiments in the Section 5 evaluate the running time of the different approaches.

4.4 Examples

Figures 3 and 4 illustrate the estimation of appearance models on real images with the algebraic and spectral methods respectively. In these examples we used $\rho = 0.06$ to select r. The values of w_0, w_1 and ϵ were estimated separately for each image by searching over discrete choices as described in the last section. For comparison we also show the appearance models computed using ground truth segmentations. For the case of a ground truth segmentation the appearance models are normalized histograms of the pixel values within each region. We see that both the algebraic and spectral methods give good results, leading to appearance models that are close to the ground truth.
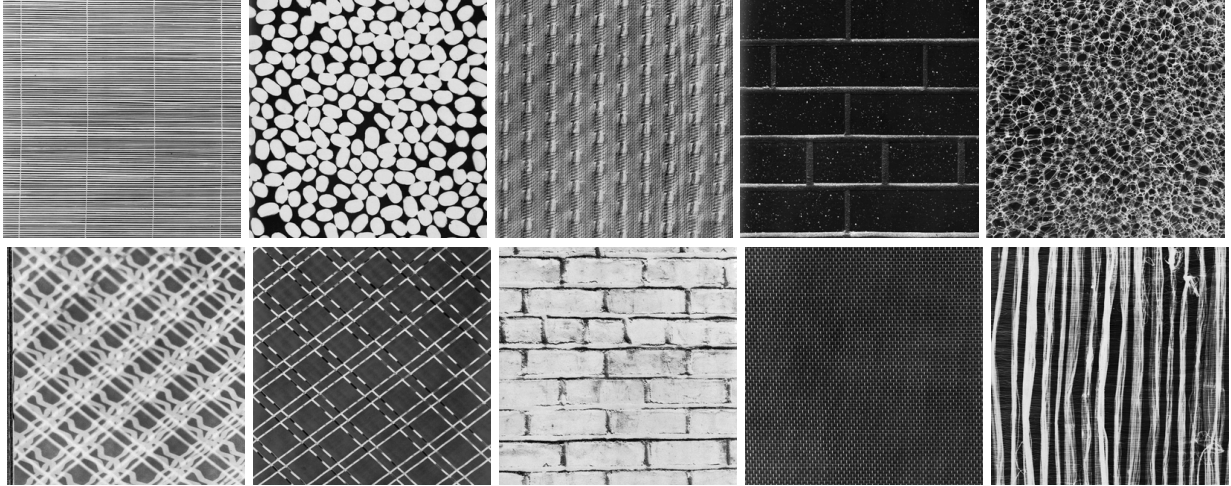
9

(a) Image	(b) θ_0	(c) θ_1

Figure 4: Estimation of appearance models with the spectral method. In (a) we show the input images and their ground truth segmentation. In (b) and (c) we show the appearance models computed using the ground truth segmentation in blue (———) and the spectral method in red (———). The images are from the Berkeley Segmentation [10] dataset.

5   Numerical Experiments

All of our algorithms were implemented in Matlab and the experiments presented here were run on a Laptop computer with an Intel Core i5-6200U CPU 2.30GHz with 8 Gb of RAM.

5.1   Evaluation Measures

We use the Bhattacharyya distance to measure similarity between two probability distributions. Let p and q be two distributions over a finite set Z. The Bhattacharyya distance beetween p and q is,

$$d_B(p,q) = -\ln\left(\sum_{x \in Z} \sqrt{p(x)q(x)}\right). \tag{15}$$

To evaluate the quality of the appearance models we estimate we compare them to the appearance models defined by a ground truth segmentation.

Let I be an image with a ground truth segmentation defined by $\mathcal{R}_0$ and $\mathcal{R}_1$. Let θ_0 and θ_1 be the normalized histograms of the pixel values within each region. Let $\hat{\theta}_0$ and $\hat{\theta}_1$ be the appearance models estimated from I using one of our algorithms. We assess the quality of the estimated appearance models using a sum of two Bhattacharyya distances, allowing for a permutation of the region labels,

$$D_B = \min\left(\frac{d_B(\theta_0, \hat{\theta}_0) + d_B(\theta_1, \hat{\theta}_1)}{2}, \frac{d_B(\theta_0, \hat{\theta}_1) + d_B(\theta_1, \hat{\theta}_0)}{2}\right). \tag{16}$$

Figure 5: Selected Brodatz patterns

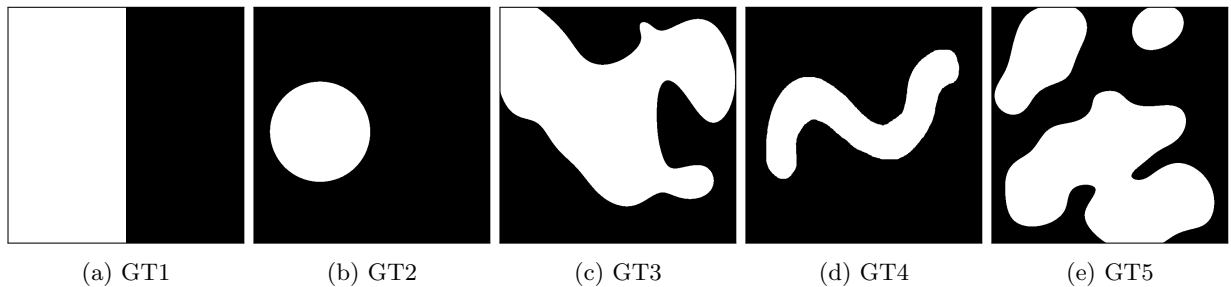(a) GT1   (b) GT2   (c) GT3   (d) GT4   (e) GT5

Figure 6: Ground truth segmentations used to generate synthetic data.

We also evaluate the accuracy of segmentations obtained using different methods by comparing them to the ground truth segmentation.

We assess the overlap between two regions $J, Q \subseteq \Omega$ in different segmentations using the Jaccard index,

$$J(S, Q) = |S \cap Q| / |S \cup Q|.$$

Let $\mathcal{Q}_0$ and $\mathcal{Q}_1$ be two regions obtained by segmenting I. We compare the segmentation defined by $\mathcal{Q}_0$ and $\mathcal{Q}_1$ to the ground truth segmentation defined by $\mathcal{R}_0$ and $\mathcal{R}_1$ using a sum of two Jaccard indices, again allowing for a permutation of the region labels,

$$\text{Jac} = \max \left(\frac{J(\mathcal{R}_0, \mathcal{Q}_0) + J(\mathcal{R}_1, \mathcal{Q}_1)}{2}, \frac{J(\mathcal{R}_0, \mathcal{Q}_1) + J(\mathcal{R}_1, \mathcal{Q}_0)}{2} \right). \tag{17}$$

5.2 Synthetic Data

We first illustrate the results of a series of experiments with synthetic data. To generate the synthetic data we used the segmentations masks in Figure 6 together with pairs of images defined as follows:

- IID: we used 50 pairs of random appearance models to generate pairs of images. For each appearance model we generate a 320×320 image where the pixel values are independent samples from the corresponding distribution.

11
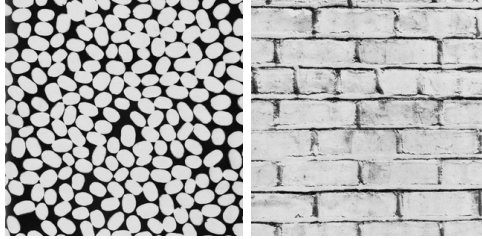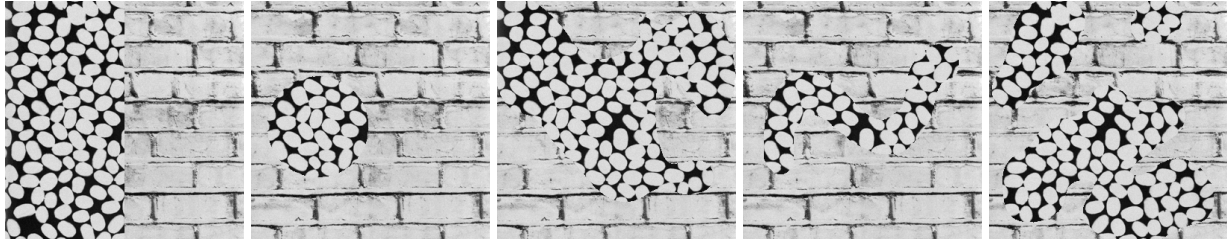
(a) Textures

(b) Synthetic images

Figure 7: Examples of synthesized images with textures.

- Brodatz: we selected all possible pairings of images from the Brodatz textures [4] shown in Figure 5. We resized the images to be 320×320 pixels and added uniform IID noise to the pixels to to remove quantization artifacts.

For each pair of images defined above we use the segmentation masks in Figure 6 to generate graylevel images with two regions.

Given a pair of images (A, B) and a binary segmentation mask M we generate a graylevel image I with $I(x) = A(x)M(x) + B(x)(1 - M(x))$. Figure 7 shows the images generated using two Brodatz patterns.

5.3 Evaluating the effect of r

In our experiments we set $r = \rho\sqrt{\Omega}$ where ρ is a parameter set by the user. This makes the distance r adaptive to the image resolution.

In Figure 8 we evaluate the quality of the appearance models estimated by our methods using different values of ρ on the synthetic data defined by the ground truth segmentations GT1 and GT2 (Figure 6). For these experiments we set w_0, w_1 and ϵ using the ground truth values defined by the corresponding segmentation masks.

Both of our algorithms almost perfectly recover the underlying appearance models for images where the pixels in each region are IID. In this case the methods work well over the whole range of values of ρ tested. This is expected since these images strictly follow both Assumption 1 and Assumption 2 and, therefore, provide the optimal setting for our algorithms.

For images with Brodatz textures Assumption 2 is violated for small values of ρ. As ρ increases the assumption is satisfied and the quality of our estimation improves.

5.4 Appearance Model Evaluation on Synthetic Images

We compare the performance of our methods for estimating appearance models to a variation of the iterative scheme described in [16], here called ALT.

In ALT, we start with an initial segmentation of the image and alternate between computing new appearance models using the current segmentation and computing a new segmentation using the current appearance
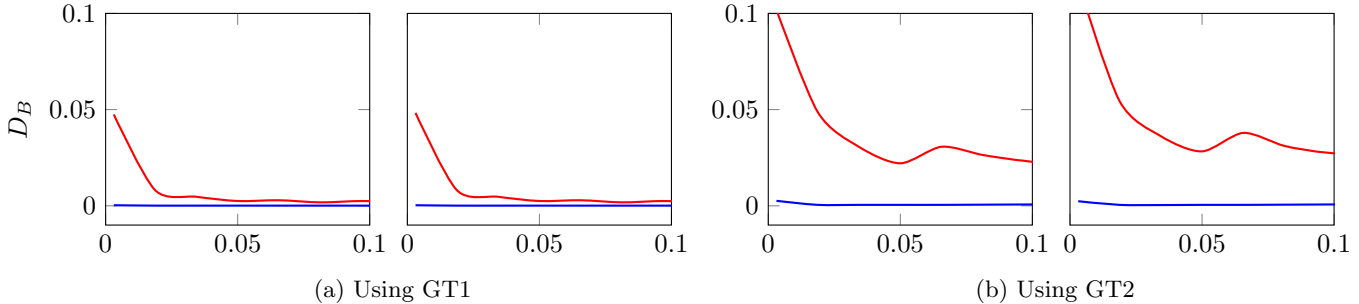
(a) Using GT1   (b) Using GT2

Figure 8: Average appearance model estimation error (D_B) as a function of ρ on images composed of IID (——) and Brodatz (——) patterns disposed as in GT1 and GT2. For both (a) and (b) the results on the left are from the algebraic method, whereas the results on the right are from the spectral method.

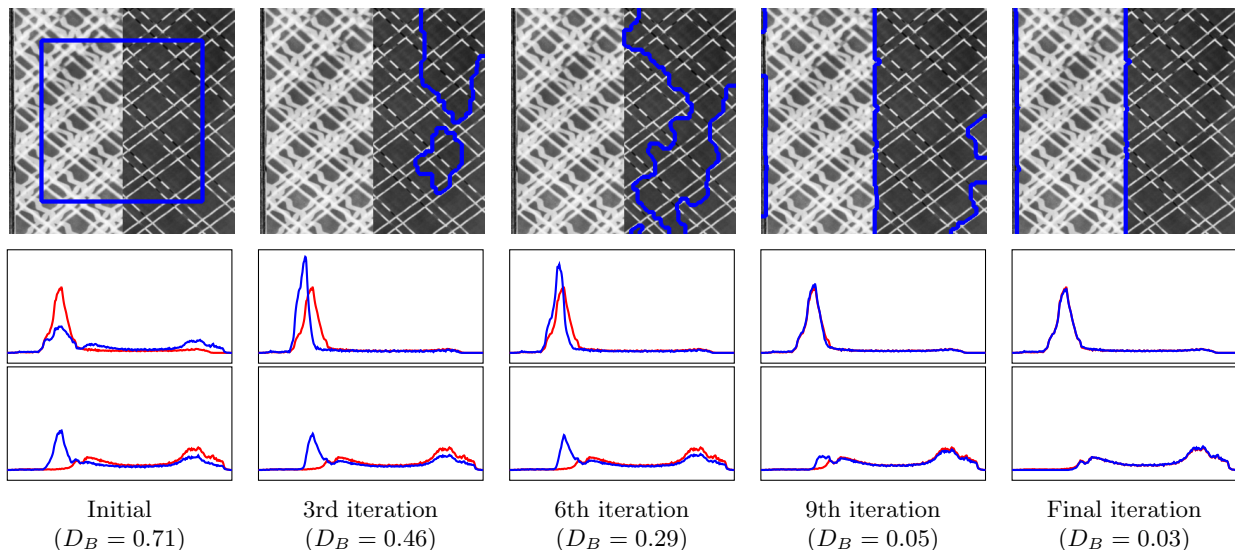| Initial | 3rd iteration | 6th iteration | 9th iteration | Final iteration |
| $(D_B = 0.71)$ | $(D_B = 0.46)$ | $(D_B = 0.29)$ | $(D_B = 0.05)$ | $(D_B = 0.03)$ |

Figure 9: Model estimation and segmentation using ALT. On the bottom, we see how both foreground and background color distributions estimated by ALT (——) evolve compared to the ground truth appearance models (——). The evolution of the segmentations given the models is shown on top.

models. This procedure is iterated until convergence. To update the appearance models using the current segmentation we histogram the pixel values in each region. We "smooth" the histograms by adding a constant $K = 1$ to their bins before normalizing them. To update the segmentation using the current appearance models we minimize the energy in Equation (1) using a max-flow/min-cut algorithm ([8, 3]).

For the experiments described here the initial segmentation used for ALT is defined by a square region in the middle of the image. Figure 9 shows an example of how the segmentation and appearance models evolve over time. Empirically, we found that ALT works well in many examples but a typical failure mode leads to assigning the whole image to single segment.

Table 1 compares the results of our methods to the result of ALT using several values of λ for the segmentation step. We used $\rho = 0.06$ (which corresponds to $r \approx 20$ pixels for the 320×320 synthetic images) for both the algebraic and spectral methods. We evaluated our algorithms using three different approaches for selecting w_0, w_1 and ϵ. In the first approach we set the parameters to the their ground truth values defined by the corresponding segmentation mask. In the second approach we fix the parameters to typical values that work well for many images. In the third approach we search over the parameters explicitly (see

13

Table 1: Average D_B distance (lower is better) between estimated and ground truth appearance models on the synthetic data generated using different segmentation masks. We evaluate our algorithms using different methods for selecting w_0, w_1 and ϵ (see text).

	Method	Image Setting										Time (s)
		GT1		GT2		GT3		GT4		GT5		
		IID	Brodatz	IID	Brodatz	IID	Brodatz	IID	Brodatz	IID	Brodatz	
Algebraic	GT w_0,w_1,ϵ	0.000	0.003	0.001	0.028	0.000	0.006	0.001	0.030	0.000	0.007	0.23
	$w_0,w_1=0.5,\epsilon=0.5\rho$	0.000	0.003	0.019	0.064	0.001	0.007	0.016	0.058	0.002	0.011	0.14
	Search w_0,w_1,ϵ	0.000	0.010	0.002	0.040	0.002	0.017	0.003	0.032	0.003	0.021	7.81
Spectral	GT w_0,w_1,ϵ	0.000	0.002	0.001	0.034	0.000	0.006	0.001	0.032	0.000	0.007	0.06
	$w_0,w_1=0.5,\epsilon=0.5\rho$	0.000	0.003	0.019	0.067	0.001	0.007	0.016	0.059	0.002	0.011	0.06
	Search w_0,w_1,ϵ	0.000	0.008	0.001	0.034	0.002	0.016	0.003	0.040	0.003	0.020	0.16
ALT	$\lambda=1$	0.044	0.091	0.000	0.112	0.000	0.079	0.000	0.098	0.000	0.083	3.70
	$\lambda=3$	0.043	0.019	0.000	0.035	0.000	0.021	0.000	0.033	0.000	0.026	3.26
	$\lambda=5$	0.043	0.020	0.005	0.014	0.000	0.004	0.032	0.019	0.042	0.016	3.29
	$\lambda=10$	0.043	0.073	0.031	0.033	0.027	0.008	0.032	0.046	0.042	0.055	3.04

Section 4.3). All of the approaches lead to good results but searching for the optimal parameters leads to a significant increase in runtime for the algebraic method.

We see that our algorithms perform extremely well on images where the pixel values in each region are IID. The results on images with textures are also good and compare favorably to ALT. This result is compelling in particular because the proposed methods do not rely on an iterative model re-estimation scheme such as in ALT, which makes them faster and independent of initialization. The average runtime of the different methods are shown in the last column of Table 1.

5.5   Segmentation Evaluation on Synthetic Images

After estimating appearance models using either the algebraic or spectral methods we compute segmentations by minimizing Equation (1) using a max-flow/min-cut algorithm ([8, 3]). We compared this approach to several texture segmentation methods.

The methods we compare to include Level Set Segmentation using Wasserstein Distances (LSWD) [13], Images as Occlusions of Textures (ORTSEG) [11] and Factorization Based Segmentation (FBS) [19]. For each of these methods, we used the Matlab implementations provided by the authors. We tuned the parameters of each method to improve their performance in our dataset. We also evaluate the segmentation results obtained with the iterative scheme ALT described above.

All of the methods we have used for comparison assume either explicitly or implicitly that regions have homogeneous appearance. FBS uses a filter bank to define local features, while LSWD and ORTSEG work with raw pixel values. LSWD, ORTSEG and FBS require the selection of a window size parameter that has a function similar to r in our methods.

We used $\rho = 0.06$ to estimate appearance models with our methods. We set w_0, w_1 and ϵ by searching over the parameters explicitly (see Section 4.3). We compute segmentations using several choices for λ in Equation (1) and evaluate each choice separately.

Figure 10 illustrates some of the segmentations obtained using the different methods for both types of images (IID and Brodatz) used for evaluation. In these examples we used $\lambda = 5$ to compute segmentations with our methods and in ALT.

Table 2 provides a quantitative evaluation on the full set of synthetic images generated using the procedure described in Section 5.2. This is the same data used to generate the results in Table 1. Notice that the runtime of our methods is increased for the segmentation experiments (Table 2) when compared to the model estimation experiments (Table 1) due to the addition of the graph cut computation to obtain a segmentation after estimating appearance models.

Table 2 demonstrates a clear advantage of our methods under the IID case. For the Brodatz setting, the results demonstrate that our methods provide high quality segmentations without relying on iterative

Table 2: Average Jac index (higher is better) of different segmentation methods on the synthetic data generated using different segmentation masks.

Method	λ	GT1 IID	GT1 Brodatz	GT2 IID	GT2 Brodatz	GT3 IID	GT3 Brodatz	GT4 IID	GT4 Brodatz	GT5 IID	GT5 Brodatz	Time (s)
Algebraic	3	1.000	0.896	0.991	0.789	0.977	0.869	0.980	0.780	0.956	0.850	8.13
	5	1.000	0.919	0.989	0.814	0.955	0.873	0.966	0.787	0.884	0.841	8.15
	7	1.000	0.937	0.986	0.838	0.919	0.856	0.886	0.782	0.708	0.811	8.22
	10	1.000	0.936	0.976	0.860	0.793	0.837	0.826	0.777	0.570	0.767	8.31
Spectral	3	1.000	0.894	0.991	0.777	0.977	0.863	0.980	0.780	0.958	0.846	0.53
	5	1.000	0.918	0.989	0.805	0.957	0.862	0.965	0.795	0.898	0.841	0.57
	7	1.000	0.934	0.986	0.819	0.918	0.845	0.884	0.783	0.759	0.808	0.64
	10	1.000	0.930	0.974	0.850	0.804	0.824	0.823	0.783	0.590	0.771	0.72
ALT	1	0.467	0.683	0.990	0.591	0.986	0.700	0.984	0.620	0.982	0.694	3.70
	3	0.500	0.874	0.991	0.761	0.981	0.854	0.977	0.757	0.974	0.820	3.26
	5	0.500	0.874	0.858	0.843	0.966	0.904	0.187	0.785	0.500	0.835	3.29
	10	0.500	0.673	0.141	0.784	0.551	0.842	0.177	0.631	0.500	0.668	3.04
LSWD	–	0.936	0.959	0.602	0.737	0.805	0.844	0.576	0.669	0.718	0.776	89.35
ORTSEG	–	0.804	0.935	0.785	0.773	0.761	0.883	0.766	0.762	0.719	0.851	1.53
FBS	–	0.585	0.908	0.582	0.734	0.549	0.842	0.581	0.700	0.547	0.810	0.08

approaches and filter banks. This makes our methods faster than most of the other approaches, while still leading to accurate results.

These results confirm the soundness of the assumptions presented in Section 3. They also confirm the efficacy of our methods for segmenting images with complex textures, despite the fact that we work directly with raw pixel values. As can be seen in Figure 10 this leads to segmentations that are accurate near region boundaries, where methods that rely on filter responses often suffer. Finally, although not presented here for the sake of simplicity, our methods could have their segmentation performance further improved when using the estimated appearance as an initial guess for an iterative scheme such as ALT.
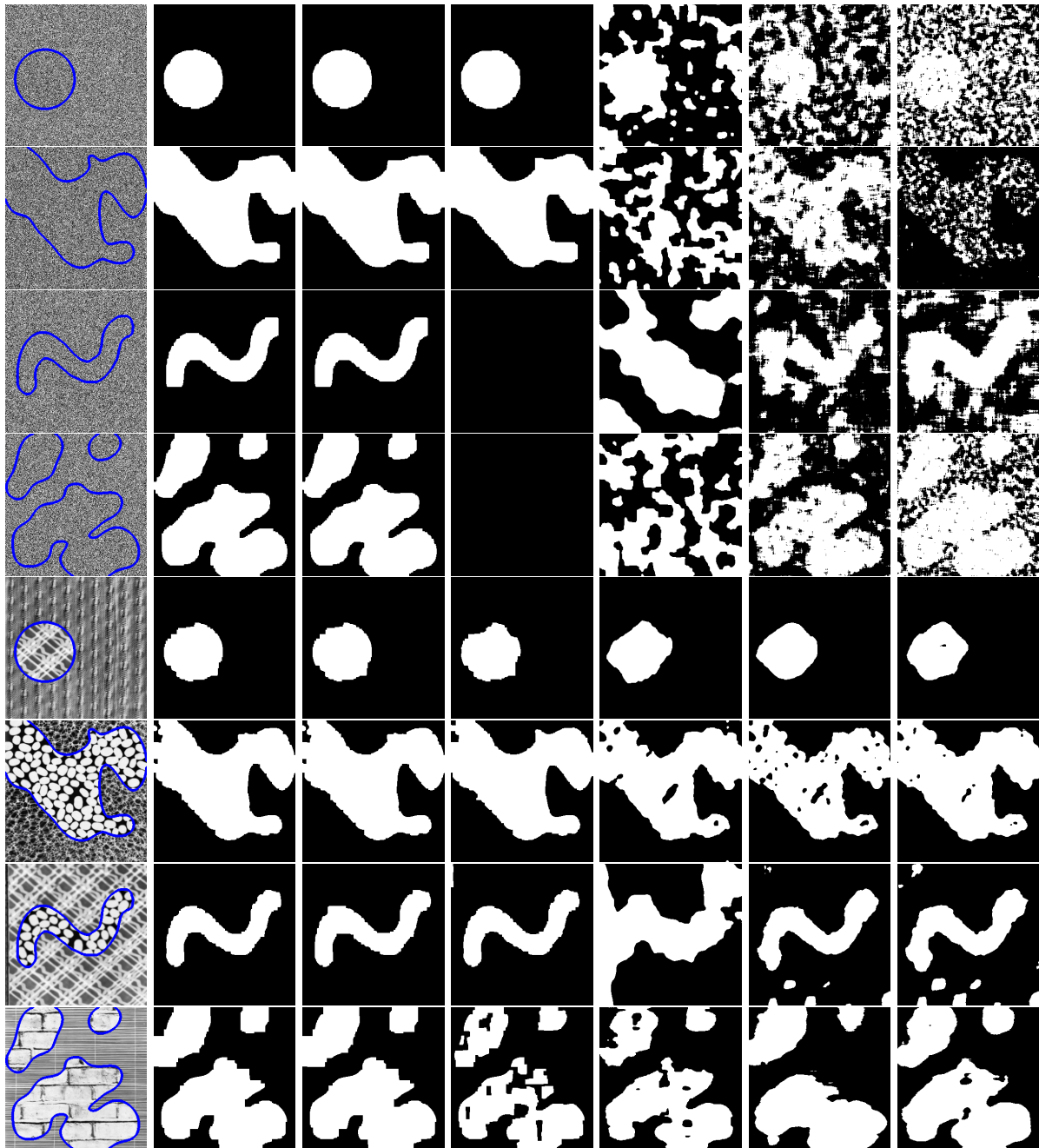
5.6 Real Images

We also tested the proposed algorithms on real images from a variety of datasets, including the Berkeley Segmentation Dataset [10], the Plant Seedlings Dataset [7] and a Scanning Electron Microscope (SEM) dataset [1]. The images were chosen such that Assumption 1 approximately holds.

Figure 11 shows some of the results obtained using our methods for estimating appearance models followed by segmentation using graph cuts. For each image, we used $\rho = 0.03$ and $\lambda = 5$. These results illustrate how the proposed algorithms work well on a variety of different types of images.

For these experiments we added a pre-processing step to our algorithms to reduce the total number of colors in RGB images to a smaller number of quantized values. This is necessary in order to obtain good estimates for α and β on images with limited resolution.

To quantize the colors in an RGB image we repeatedly partition the color space until each partition has at most 1000 pixels. Starting from the whole set of pixels, we partition the set into two using a random hyperplane in RGB space going through the center of mass of the set. We recurse this procedure until the stopping criteria is met. The same approach could be used for vector valued images such as hyperspectral images that arise in remote sensing applications.
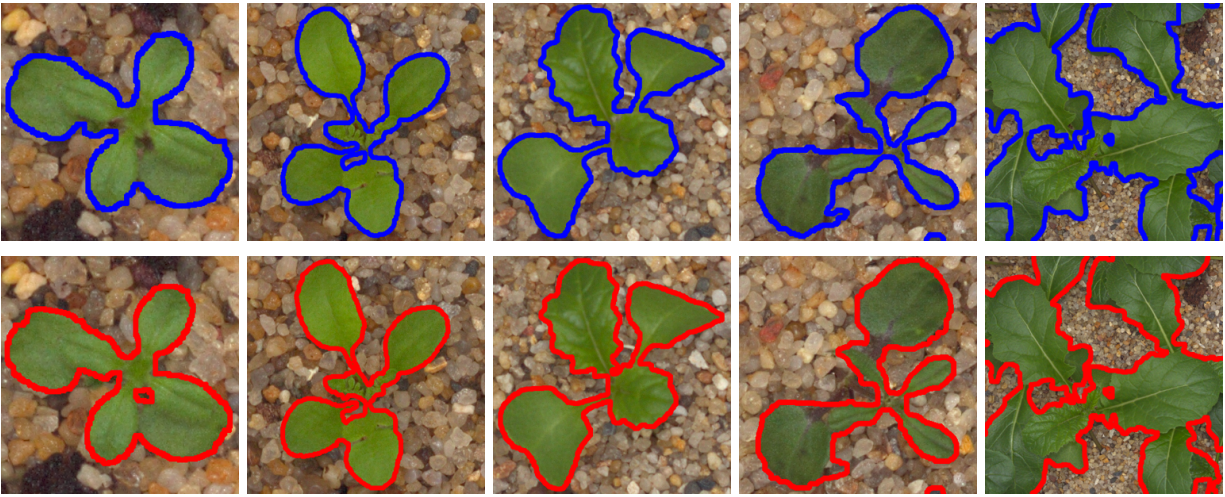
(a) Original   (b) Algebraic   (c) Spectral   (d) ALT   (e) LSWD   (f) ORTSEG   (g) FBS
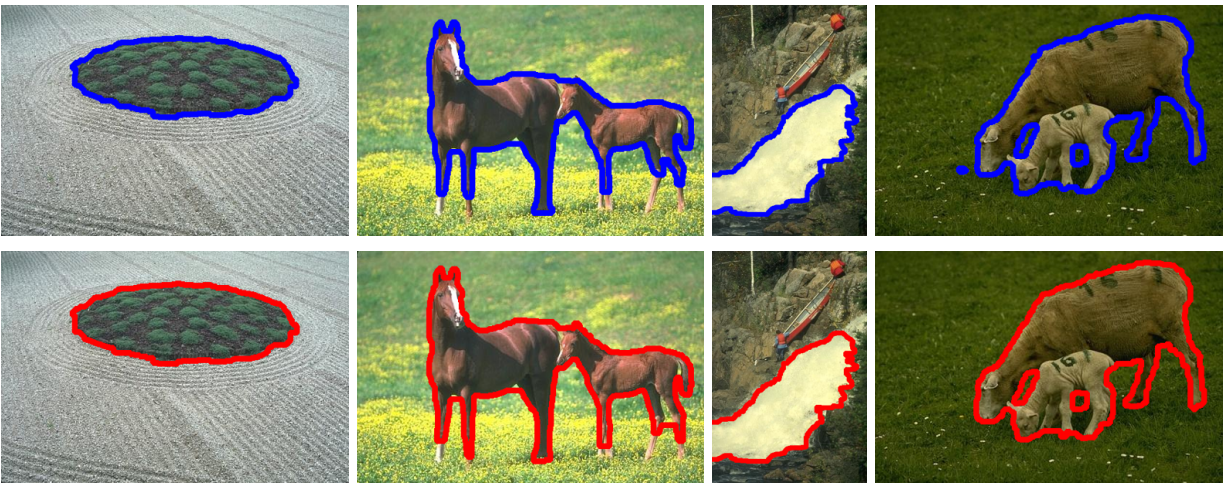
Figure 10: Selected segmentation results comparing our algorithms in (b) and (c) to other methods.
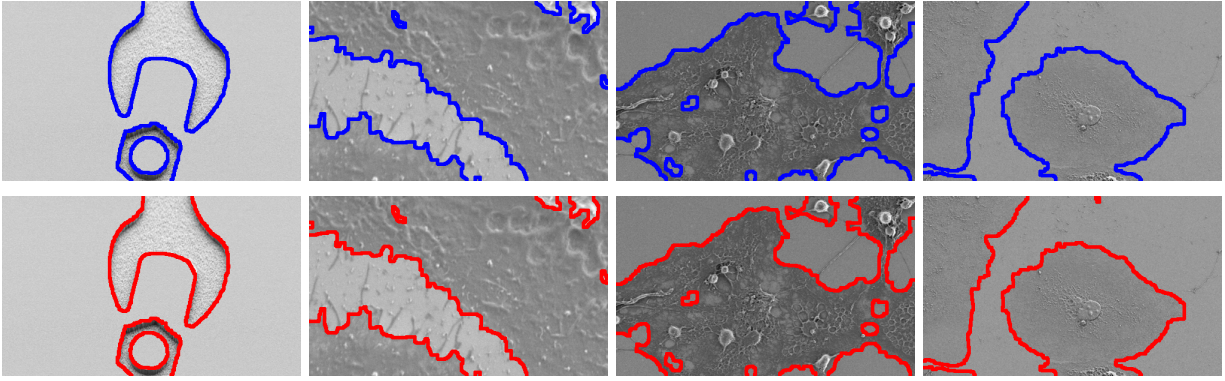
(a) [1]

(b) [7]

(c) [10]

Figure 11: Application our methods on real images from a variety of different datasets. The blue and red contours are the results of segmentation using appearance models estimated using the algebraic and spectral methods, respectively.

6   Conclusion

Many image segmentation algorithms rely on appearance models to classify pixels into different regions. We have shown that appearance models can be estimated directly from an unsegmented image. Our approach is based on novel algebraic expressions that relate local image statistics to the appearance models of different image regions. Our experiments demonstrate the algorithms we introduce in this paper work well in a variety of settings. The resulting appearance models can be used to segment images of different types, including textured images and other images where regions have complex appearances. These results also suggest that segmentation algorithms can be improved by making use of second order pixel statistics.

Thus far we have focused on the problem of binary segmentation. An interesting direction for future work would be to extend our approach to images with more than two regions. As discussed in Section 3.1 it is possible to derive algebraic constraints similar to the ones we have used to the case of images with (at most) m regions. However, solving for appearance models using these constraints remains a challenging computational problem.

References

[1] Rossella Aversa, Mohammad Hadi Modarres, Stefano Cozzini, and Regina Ciancio. NFFA-EUROPE - SEM dataset, 2018.

[2] Julian Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society: Series B (Methodological)*, 48(3):259–279, 1986.

[3] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. In *IEEE International Conference on Computer Vision*, volume 1, pages 377–384, 1999.

[4] Phil Brodatz. *Textures: a photographic album for artists and designers*. Dover Pubns, 1966.

[5] Tony F Chan and Luminita A Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001.

[6] Pedro Felzenszwalb and Daniel Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[7] Thomas Mosgaard Giselsson, Rasmus Nyholm Jørgensen, Peter Kryger Jensen, Mads Dyrmann, and Henrik Skov Midtiby. A public image database for benchmark of plant seedling classification algorithms. *arXiv preprint arXiv:1711.05458*, 2017.

[8] Dorothy M Greig, Bruce T Porteous, and Allan H Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society: Series B (Methodological)*, 51(2):271–279, 1989.

[9] Anil K Jain and Farshid Farrokhnia. Unsupervised texture segmentation using gabor filters. *Pattern Recognition*, 24(12):1167–1186, 1991.

[10] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *IEEE International Conference on Computer Vision*, volume 2, pages 416–423, July 2001.

[11] Michael T McCann, Dustin G Mixon, Matthew C Fickus, Carlos A Castro, John A Ozolek, and Jelena Kovacevic. Images as occlusions of textures: A framework for segmentation. *IEEE transactions on Image Processing*, 23(5):2033–2046, 2014.

[12] Shervin Minaee, Yuri Y Boykov, Fatih Porikli, Antonio J Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[13] Kangyu Ni, Xavier Bresson, Tony Chan, and Selim Esedoglu. Local histogram based segmentation using the Wasserstein distance. *International Journal of Computer Vision*, 84(1):97–111, 2009.

[14] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23:309–314, 2004.

[15] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[16] Meng Tang, Ismail Ben Ayed, and Yuri Boykov. Pseudo-bound optimization for binary energies. In *European Conference on Computer Vision*, pages 691–707, 2014.

[17] Meng Tang, Lena Gorelick, Olga Veksler, and Yuri Boykov. Grabcut in one cut. In *IEEE International Conference on Computer Vision*, pages 1769–1776, 2013.

[18] Sara Vicente, Vladimir Kolmogorov, and Carsten Rother. Joint optimization of segmentation and appearance models. In *IEEE International Conference on Computer Vision*, pages 755–762, 2009.

[19] Jiangye Yuan, Deliang Wang, and Anil M Cheriyadat. Factorization-based texture segmentation. *IEEE Transactions on Image Processing*, 24(11):3488–3497, 2015.