Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multi-band Image Segmentation.

Song Chun Zhu

Harvard Robotics Laboratory.

Technical Report no. 94-10

Abstract

We present a novel statistical and variational approach to image segmentation based on a new algorithm named *region competition*. This algorithm is derived by minimizing a generalized Bayes/MDL(Minimum Description Length) criterion using the variational principle, and it combines the best features of snake/balloon model and region growing. Thus we show that existing techniques in early vision such as, Snake/balloon models, region growing, and Bayes/MDL are addressing different views of the same problem and they can be unified within a common statistical framework to gain the advantage of all of them. We further analyze how to optimize the precision of the resulting boundary location by studying the statistical properties of the region competition algorithm, and discuss what are the "good" initial conditions for the algorithm. We will briefly discuss the relations between region competition and local edge filters. We also present a novel color model, and apply our method to color and texture segmentation. Our algorithm is demonstrated on grey level images, color images and texture images.

** Version III March.1995

Content:

1. Introduction

2. Snake,Region Growing and Energy/Bayes/MDL

    2.1 Sankes and Balloons

    2.2 Region Growing and Merging

    2.3 Energy, Bayes and MDL

3. Region Competition For Grey Level Images

    3.1 From MDL to A Unified Framework.

    3.2 Generalizing the MDL Criterion for Region Competition.

    3.3 Simulations on Grey Level Images.

4. Analyzing the Boundary and windows

    4.1 Precision of the Boundary

    4.2 The Size and Shape of the Windows

5. The Analysis of Seeds

    5.1 What is a "good" seed?

    5.2 The Domino Effect of Seed Transferring

6. Multivariate Models: Color

    6.1 The Color Model

    6.2 Color Segmentation

    6.3 Color Images Experiments

7. Texture Segmentation.

8. Discussion

Acknowledgement

Appendix I, II

References

1 Introduction

Image segmentation is a critical problem of early vision and it is also one of the most intensively studied area in computer vision. In the past two decades, researchers had developed various approaches to image segmentation from different perspectives. These methods can be roughly classified into four classes: (i) Local filtering approaches such as the Canny edge detector [4], (ii) Snake [16] and Balloon methods [6], [7], [37], (iii) Region growing and merging techniques [2] [23], and (iv) Global optimization approaches based on energy functions [26] or Bayesian [10] [3] and MDL criteria [19] [15]. A common property of these approaches is that they all make *hypotheses* about the image, *test* features and make *decisions* by applying thresholds explicitly or implicitly. As shown by the shadowed areas in figure (1), a major difference between the four approaches lies in the domains on which the hypotheses, tests, and decisions are based . These approaches all have certain drawbacks. The filtering approach only makes use of local information and cannot guarantee continuous closed edge contours. Snake/balloon models make use only of information along the boundary and requires good initial estimates to yield correct convergence. An advantage of region growing is that it does test the statistics inside the region, however it often generates irregular boundaries and small holes. In addition, all these three methods lack a global criterion for segmenting the entire image. By contrast, energy/Bayes/MDL have global criteria but it is often very difficult to find their minima.

We have developed a common statistical framework for image segmentation. The algorithm, which we call *region competition*, is derived by minimizing a generalized Bayes/MDL(Minimum Description Length) criterion using the variational principle, and it combines the most attractive features of snake/balloon model and region growing. Thus we show that three of the standard approaches: (i) snakes and balloons, (ii) region growing by using statistical tests such as Fisher's test, (iii) Energy, Bayesian and MDL methods are addressing different views of the same

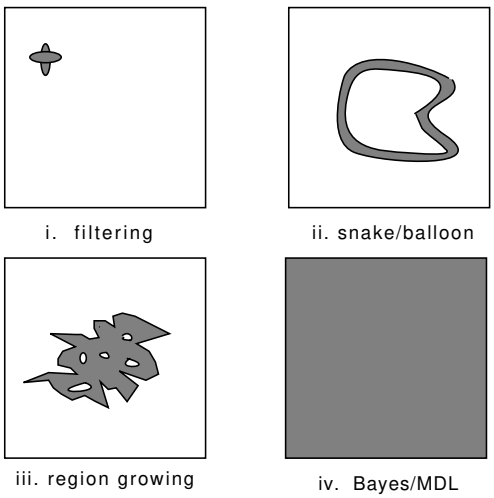


Figure 1 The shadowed areas show the image domains based on which the hypotheses, tests, and decisions are made by i. the filtering, ii. the snake/balloon, iii. the region growing, and iv. the Bayes/MDL approach respectively for edge detection and image segmentation.

problem and they can be unified within a common statistical framework to gain the advantage of all of them.

We further analyze the precision of the resulting boundary by studying the statistics behaviors of the region competition algorithm. It turns out that there exists an uncertainty interval for the boundary location. Then we show how to minimize such uncertainty interval by choosing the optimal windows. Two heuristics are provided to improve the region competition algorithm. Then the relation between region competition and local filters is briefly discussed,

Like all greedy algorithms, the performance of region competition will depends on the initial conditions–more precisely the choice of the seeds. This paper discusses the criteria for defining a "good" initial conditions, and uncovers an interesting Domino effect of transferring "bad" seeds into "good" ones. Some empirical results are addressed.

This approach is applied both for obtaining global segmentation of images and for finding individual regions. We illustrate how it works by applying it to two typical grey level images.

We then applied our approach to deal with color by using a novel color model. A key drawback of most work on image segmentation is that it is based on image properties rather than on properties of the object itself. Consider, for example, an image of a person. The skin color is (approximately) constant over the body but the intensity image will vary because of the person's shape. Similarly, clothes often have regions of homogeneous albedo or texture, but will often have sharp shadows due to creases. As we will show, provided that color information is available, we can generalize our framework and produce segmentations based on material properties. Finally we run our algorithm on texture images. Results are demonstrated for both color and texture segmentation.

This paper has some similarities to work described by [27] and [23]. Both approaches use statistical tests to grow multiple seed regions independently, and then use the MDL criterion to compress the overlapping between regions. Our method use the region growing as a way of minimizing the MDL criterion for the entire image directly. Our work also has some similarities to ideas discussed in [26] and [18] [21]. These works differ from ours by not using a statistical criterion.

This paper is organized as follows. Section (2) sets the scene by briefly describing Snakes and Balloons, Region Growing and Energy/Bayes/MDL. In Section (3) we introduce our approach for grey level images and describe implementation results. In section (4), we analyze the precision of the boundary location and discuss how to choose the windows. Then section (5) analyzes the influence of initial conditions. Section (6) describes a novel color model, and we argue that our color model has advantages over the standard ones. Then we extend our approach to color images, thereby allowing us to filter out intensity gradients, and gives results for these cases. In section (7), we apply our method to texture images. Finally section (8) discusses the limits and several possible extensions of our works.

2 Snakes, Region Growing and Energy/Bayes/MDL

This section will briefly review the properties of snakes, region growing and Energy/Bayes/MDL.

2.1 Snakes and Balloons

A snake [16] is an active contour defined by $\Gamma(s) = (x(s), y(s))$, where s can be the arc length of the contour[1]. Here we assume that $\Gamma(s)$ is the closed boundary of a region R (i.e. $\Gamma(s) = \partial R$).

A typical energy for a snake is:

$$E[\Gamma(s)] = \oint_{\Gamma(s)} \{\frac{1}{2}(\alpha \left|\Gamma_s\right|^2 + \beta \left|\Gamma_{ss}\right|^2) - \lambda \left|\vec{\nabla} I \cdot \vec{\nabla} I\right|\} ds \tag{1}$$

which we minimize by steepest descent:

$$\frac{d\Gamma_{(s)}}{dt} = -\frac{\delta E}{\delta \Gamma} = -\alpha \Gamma_{ss} + \beta \Gamma_{ssss} + \lambda \vec{\nabla} \left|\vec{\nabla} I \cdot \vec{\nabla} I\right|, \tag{2}$$

where $\Gamma_{ss} = (\ddot{x}(s), \ddot{y}(s))$, etc.

The balloon models[6], [7], [37] are motivated by the desire to drive the snake automatically to a good initial position. They introduce an additional force term $\nu \vec{\mathbf{n}}_{(s)}$ to Equation (2) which pushes the contour out (or in) along its normal $\vec{\mathbf{n}}_{(s)} = (\dot{y}, -\dot{x})$.

We observe that this additional force term can be derived from an energy function. It corresponds, see Appendix I, to an additional energy term $E_{add}[\Gamma(s)] = -\nu \int \int_R dxdy$. Thus the energy for the balloon is:

$$E[\Gamma_{(s)}] = \oint_{\Gamma_{(s)}} \{\frac{1}{2}(\alpha \left|\Gamma_s\right|^2 + \beta \left|\Gamma_{(ss)}\right|^2) - \lambda \left|\vec{\nabla} I \cdot \vec{\nabla} I\right|\} ds - \nu \int \int_R dxdy \tag{3}$$

[1]The contour can also be represented parametrically, for example by B-splines or Fourier coefficients, but we will not consider such variants in this paper.

Thus the balloon tries to maximize its area while smoothing its bounding contour and maximizing the intensity gradient along the contour.

2.2 Region Growing and Merging

The goal of region merging and region growing [13], [23] is to divide the domain R of the image I into regions $\{R_i : i = 1, ..., M\}$ so that $R = \cup_{i=1}^{M} R_i$, $R_i \cap R_j = \phi$ if $i \neq j$, and I satisfies a homogeneity criterion on each R_i.

Region merging works by building up complicated regions by combining smaller regions based on a statistical similarity test. A popular choice is Fisher's test[33], for example, suppose there are two adjacent regions R_1 and R_2, where $n_1, n_2, \hat{\mu}_1, \hat{\mu}_2, \hat{\sigma}_1^2, \hat{\sigma}_2^2$ are the sizes, sample means and sample variances of R_1, R_2 respectively. Then in order to decide whether or not to merge them, we can look at the squared *Fisher distance*:

$$\frac{(n_1 + n_2)(\hat{\mu}_1 - \hat{\mu}_2)^2}{n_1\hat{\sigma}_1^2 + n_2\hat{\sigma}_2^2} = \frac{n\hat{\sigma}^2}{n_1\hat{\sigma}_1^2 + n_2\hat{\sigma}_2^2} - 1, \tag{4}$$

where $n = n_1 + n_2$ and $\hat{\sigma}^2$ is the sample variance of the mixture region. A generalization to the multi-dimensional case is called Hotelling's test [14]. If this statistic is below a certain threshold then the regions are merged.

Region growing can be considered as a special case of region merging, where R_1 is the growing region, while R_2 is a single pixel at the boundary of R_1, i.e. $n_2 = 1$ and n_1 is very large (say $n_1 > 100$). In this case we can treat $\mu = \hat{\mu}_1, \sigma^2 = \hat{\sigma}_1^2$, and $\mu_2 = I_{(x,y)}$ (the intensity at point (x, y)), and approximate the squared Fisher distance by: $\frac{(I-\mu)^2}{\sigma^2}$.

A variant of region growing [23] is to fit the intensity within each region to a parameterized model, such as a plane or a quadratic form. Then tests like Equation (4) can be applied to the residuals of the boundary pixels after fitting.

Although region growing algorithms are very intuitive they can rarely be proven to converge to the minimum of some global cost function, and the resulting regions

may have noisy boundaries. Another drawback results from the Fisher's test, see Equation (4), which cannot distinguish between two distributions with the same means but different variances. This problem will be discussed in a later section.

2.3 Energy, Bayes and MDL

Both Bayes and MDL specify ways for segmenting an entire image using an energy function criteria. These two approaches are motivated by different considerations but it is straightforward to transfer an MDL criterion into a Bayesian one and vice versa.

Bayesian estimation is a common approach to image segmentation [10]. The observed image is modeled as being a degraded version of an ideal image which is assumed to be piecewise smooth. For example, the energy function used in Mumford and Shah [26], and in Blake and Zisserman [3] (i.e. the *weak membrane* model) is:

$$E[f,\Gamma] = \mu \int\int_R (f-I)^2 dxdy + \lambda \int\int_{R-\Gamma} ||\nabla f||^2 dxdy + \nu|\Gamma|, \tag{5}$$

where I is the input image, f is the output image, and Γ labels the discontinuities. It is easy to see that when $\lambda \mapsto \infty$ this reduces to a cartoon model:

$$E[\{f_i\},\Gamma] = \sum_i \int\int_{R_i} \frac{(f_i - I)^2}{\sigma^2} dxdy + \nu|\Gamma|, \tag{6}$$

where f_i is constant within each region R_i and $\sigma^2 = \frac{1}{\mu}$ is constant over the entire image.

It is usually very difficult to minimize the energy functions resulting from Bayesian models. Algorithms such as simulated annealing [10], graduated non-convexity [3] and deterministic annealing [9] are perhaps the most successful.

On the other hand, Leclerc [19] suggest that segmenting image according to the above Bayesian model should be equivalent to obtaining the minimum description length (MDL) of it in terms of a previously specified description language. Equation (6) is a special case where the image description is in terms of piecewise

constant values, the standard deviation is assumed to be globally constant, and we pay a penalty for encoding the boundary proportional to its length $|\Gamma|$. A more typical MDL criterion occurs in Equation (7) (see next section). It differs from Equation (6) by letting the σ's be unknown variables which are assumed to be constant within each region. The energy function is difficult to minimize and Leclerc [19] proposes a variant of graduated non-convexity[3].

3 Region Competition for Grey Level Images

In this section, we first derive our theory from the MDL optimization criterion, then we show how snake/balloon and region growing can be unified with a generalized MDL criterion under a common statistical framework. Finally we show simulations of our region competition algorithm on two typical grey level images.

3.1 From MDL to A Unified Framework

The goal of image segmentation is to partition the image into subregions with homogeneous intensity (color or texture) properties, such regions will typically correspond to objects or object parts in the scene. Before describing the algorithm, we first need a definition of homogeneity.

In this section a region R is considered to be homogeneous if its intensity values are consistent with having been generated by one of a family of pre-specified probability distributions $P(I : \alpha)$, where α are the parameters of the distribution. We assume that the probability models may be different in different regions. Such an assumption will be necessary if the application domain is human skin and clothes, where different patches will typically have different properties - and hence will need to be described by different distributions. In later sections, we will describe how the probability distributions $P(I : \alpha)$ can be generalized to residues of fitting certain physical models or multi-band features extracted from the images, in which cases α may includes the parameters specifying those physical models. Now suppose that

the entire image domain R has been initially segmented into M piecewise "homogeneous" underlying regions $\{R_i\}, i = 1, 2, ..., M$, i.e. $R = \cup_{i=1}^{M} R_i$, $R_i \cap R_j = \emptyset$, if $i \neq j$. Let ∂R_i be the boundary of region R_i where we define the direction of ∂R_i to be counter-clockwise, i.e. when we travel along the boundary, R_i is on the left hand side [2]. Let $\Gamma = \cup_{i=1}^{M} \Gamma_i$ be the edges or segmentation boundaries of the entire image with $\Gamma_i = \partial R_i$.

Our MDL criterion corresponds to a global energy functional which is the continuum limit of Leclerc's [19] for an appropriate choice of the family of probability distributions. This gives:

$$E[\Gamma, \{\alpha_i\}] = \sum_{i=1}^{M} \{\frac{\mu}{2} \oint_{\partial R_i} ds - \int\int_{R_i} \log P(I_{(x,y)} : \alpha_i) dxdy + \lambda\}, \tag{7}$$

where the first term within the braces is the length of the boundary curve ∂R_i for region R_i. Without prior knowledge about the shape of the region, we simply assume that the code length is proportional to the curve length where μ is the code length for unit arc length. Since all edge segments are shared by two adjacent regions, we divide the first term by a factor of 2. The second term is the sum of the cost for coding the intensity of every pixel (x, y) inside region R_i according to a distribution $P(I_{(x,y)} : \alpha_i)$. λ is the code length needed to describe the distribution and code system for region R_i, we simply assume λ is common to all regions.

Because the energy E in equation (7) depends on two group of variables –the segmentation Γ and the parameters α_i's, we propose a greedy algorithm which minimizes the energy functional by iterating two stages.

In the first stage, we fix Γ, in other words, we fix R_i and $I_{(x,y)}$ $\forall (x, y) \in R_i$, we solve for the α_i, $i = 1, 2, ..., M$ to minimize the description cost for each region. This corresponds to setting[3]:

$$\alpha_i^* = \arg\min_{\alpha_i} \{-\int\int_{R_i} \log P(\alpha_i \mid I_{(x,y)}) dxdy\}, \quad \forall\ i \tag{8}$$

[2] In case of holes, it will be clock-wise.

[3] We assume α_i has uniform *a priori* distribution.

in discrete case:

$$\alpha_i^* = \arg\max_{\alpha_i} \prod_{(x,y)\in R_i} P(\alpha_i \mid I_{(x,y)})\}, \quad \forall\ i \tag{9}$$

$$= \arg\max_{\alpha_i} P(\alpha_i \mid \{I_{(x,y)}, \forall (x,y) \in R_i\}) \quad \forall\ i \tag{10}$$

In other words, the α_i 's are estimated by maximizing the conditional probabilities. For many distributions this can be done analytically. For example, the α_i^*'s are simply the sample mean and sample variance over pixels inside R_i if $P(I : \alpha_i)$ is Gaussian distribution.

In the second stage, we fix the $\{\alpha_i\}$, thus $P(I : \alpha)$ becomes conditional probability distribution $P(I \mid \alpha)$ and then do steepest descent with respect to Γ. For any point $\vec{v} = (x, y)$ on the edge Γ:

$$\frac{d\vec{v}}{dt} = -\frac{\delta E[\Gamma, \{\alpha_i\}]}{\delta \vec{v}} \tag{11}$$

where the right hand side is (minus) the functional derivative of the energy E.

Taking the functional derivative, see Appendix I, yields the motion equation for point $\vec{v}$:

$$\frac{d\vec{v}}{dt} = \sum_{k\in Q_{(\vec{v})}} \{-\frac{\mu}{2}\kappa_{k(\vec{v})}\vec{\mathbf{n}}_{k(\vec{v})} \quad + \quad \log P(I_{(\vec{v})}|\alpha_k)\vec{\mathbf{n}}_{k(\vec{v})}\} \tag{12}$$

where $Q_{(\vec{v})} = \{k \mid \vec{v} \text{ lies on } \Gamma_k\}$, i.e. the summation is done over all regions R_k if $\vec{v}$ is on Γ_k. $\kappa_{k(\vec{v})}$ is the curvature of Γ_k at point $\vec{v}$ and $\vec{\mathbf{n}}_{k(\vec{v})}$ is the unit normal to Γ_k at point $\vec{v}$. Because we define the direction of each Γ_k as counter-clockwise, $\vec{\mathbf{n}}_k$ will point outward R_k. [4]

Equation (12) has a simple intuitive interpretation. There are two kinds of "forces" acting on the contour, both pointing along the normal. The first term, the *smoothing force*, is strongest at points of high curvature. Figure (2.a) shows the smoothing force at points along the region boundary. This force is independent of

[4] Recently the author noticed that a similar form of equation (12) was derived in [35]. But it is not derived in terms of MDL, and the statistics meanings are not analyzed there.

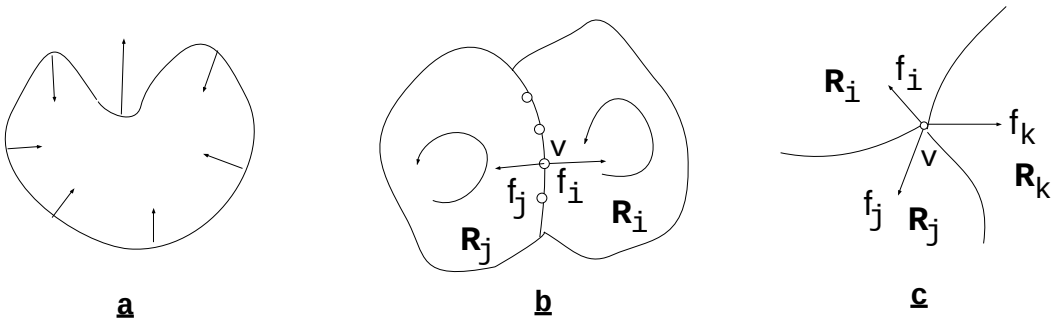


Figure 2 The forces acting on the contour: a. the smoothing forces, b, the statistics force at an boundary point. c, the statistics forces at a junction point

---

the direction of the curve and it tries to make the curve as straight as possible. The second term is the *statistics force*, $\vec{f} = logP(I \mid \alpha)\vec{\mathbf{n}}$. Since $logP \leq 0$, the statistics forces always compress the region. The better the point $\vec{v}$ satisfies the homogeneity requirement, the larger the $P(I \mid \alpha)$, and the weeker the statistics force.

For example, as shown in Figure (2.b), $\vec{v}$ is a point on the common boundary of region R_i and R_j. Since curves Γ_i and Γ_j have inverse normal vectors at $\vec{v}$, $\vec{\mathbf{n}}_i = -\vec{\mathbf{n}}_j$ and $\kappa_i\vec{\mathbf{n}}_i = \kappa_j\vec{\mathbf{n}}_j$ The motion equation for $\vec{v}$ is:

$$\frac{d\vec{v}}{dt} = -\mu\kappa_{i(\vec{v})}\vec{\mathbf{n}}_{i(\vec{v})} \; + \; (logP(I_{(\vec{v})}|\alpha_i) - logP(I_{(\vec{v})}|\alpha_j))\vec{\mathbf{n}}_{i(\vec{v})} \tag{13}$$

$$= -\mu\kappa_{i(\vec{v})}\vec{\mathbf{n}}_{i(\vec{v})} \; + \; log(\frac{P(I_{(\vec{v})}|\alpha_i)}{P(I_{(\vec{v})}|\alpha_j)})\vec{\mathbf{n}}_{i(\vec{v})} \tag{14}$$

It is easy to see that the smoothing term by itself is the *Euclidean geometric heet flow* equation used for curve smoothing and evolution, and it is equal to the following heat diffusion equation.

$$\frac{d\Gamma}{dt} = \frac{\partial^2\Gamma}{\partial s^2} \tag{15}$$

where s is the arc length of the curve Γ. Detailed discussion about the properties of this equation is referred to [12] and [29].

Besides the smoothing term, the motion of $\vec{v}$ is determined by the likelihood ratio test. If $P(I_{(\vec{v})}|\alpha_i) > P(I_{(\vec{v})}|\alpha_j)$, i.e. the intensity at $\vec{v}$ fits better to the distribution

of region R_i than to that of region R_j, then the boundary will move along $\vec{\mathbf{n}}_i$. This is the same for points which are shared by several region boundaries, Figure (2.c) show the statistics forces at a junction point. Intuitively, adjacent regions compete for ownership of pixels along their boundaries, subject to the smoothness constraint, and motion of the edges are mainly determined by the probability distribution of each regions and the properties of the edge. This is why we call our algorithm *region competition.*

Now we find it easy to unify region growing, the balloon model and MDL in a common statistics framework.

Firstly, the region growing algorithm can be considered to be a degenerate case of region competition where we treat the growing region as R_1, with $P(I : \alpha_1)$ chosen according to our desired homogeneity criterion, and the background region as R_0 (so that $R_1 \cup R_0 = R$) which has uniform probability distribution P_0. Then the motion equation for each point $\vec{v}$ along the boundary is:

$$\frac{d\vec{v}}{dt} = (logP(I_{(\vec{v})}|\alpha_1) - logP_0)\vec{\mathbf{n}}_{(\vec{v})} \tag{16}$$

where $\vec{\mathbf{n}}$ is the normal of the region contour. This directly corresponds to region growing where people test the probability $P(I_{(\vec{v})}|\alpha_1)$, and compare it with the absolute threshold P_0. Fisher's test [33] and the χ^2 test [23] correspond to particular choices of $P(I_{(\vec{v})}|\alpha_1)$.

Secondly, the underlying statistical assumption for the balloon model is even simpler, it treats both the current region R_1 and the background R_0 as uniform distributions P_1, P_0 with $\nu = logP_1 - logP_0$. According to the balloon model, taking the derivatives of equation (3), we have the motion equation for $\vec{v}$ on the boundary:

$$\frac{d\vec{\Gamma}_{(s)}}{dt} = -\alpha\Gamma_{ss} + \beta\Gamma_{ssss} + (logP_1 - logP_0)\vec{\mathbf{n}}_{(s)} + \lambda\vec{\nabla}\left|\vec{\nabla}I \cdot \vec{\nabla}I\right| \tag{17}$$

In this equation, the smoothing force is only slightly different from that in Equation (14), observe that $\kappa = \vec{\Gamma}_{ss} \cdot \vec{\mathbf{n}}$ (i.e. the curvature is the projection of Γ_{ss} along

the normal). In equation (17), the threshold for deciding where to stop moving the contour is provided by the local edge measurement–the last term. Since the balloon model does no statistics inside the regions it often fails to segment images into homogeneous regions.

In summary, we can compare the region growing, balloon models and Bayes/MDL (as well as the filter approaches) in a common three-stage statistics framework: hypotheses, tests, and decisions. The differences between these approaches can be easily found by studying what kind of hypotheses they have made for the input images, how they test these hypotheses, and how they make the decision. The MDL criterion, which we adopt, assumes that images consist of regions with smooth boundaries and homogeneous intensity properties defined by a family of probability distribution. For each region it tests the statistics $P(I : \alpha)$ which can be specified by any test like Fisher's test, χ^2 test etc. It decides the position of the boundary by the likelihood ratio instead of an absolute thresholding. Moreover, region competition provides a global criterion for segmenting the entire image instead of for each region, and it is faster than the annealing approaches.

3.2 Generalizing the MDL Criterion for Region Competition

To proceed further we need to specify a family of probability distributions $P(I : \alpha)$. In this paper we will consider Gaussian distributions. Thus we set $\alpha = (\mu, \sigma)$, and $P(I_{(x,y)} : (\mu, \sigma)) = \frac{1}{\sqrt{2\pi}\sigma} \exp\{-\frac{(I_{(x,y)}-\mu)^2}{2\sigma^2}\}$.

However, an analysis of the motion Equation (14) reveals some underlying disadvantages for the greedy algorithm that we proposed .

First, the statistics force at each boundary point (x, y) will depend on $P(I_{(x,y)}|\mu, \sigma)$, in other words on the probability that $I_{(x,y)}$ on the boundary can be generated by a Gaussian distribution $N(\mu, \sigma^2)$. This force seems plausible but since only a single sample is taken from this distribution there is a reasonable chance that $I_{(x,y)}$ will lie on one of the tails of the distribution. The statistics force therefore may be overly

sensitive to fluctuations in the image. This can be seen intuitively in Figure (3.a), where the solid curves shows two overlapping Gaussian distributions and the distribution of the misclassification is indicated by the shadowed area. In the next section, a quantitative analysis will be derived to show the amount of fluctuation of the statistics force.

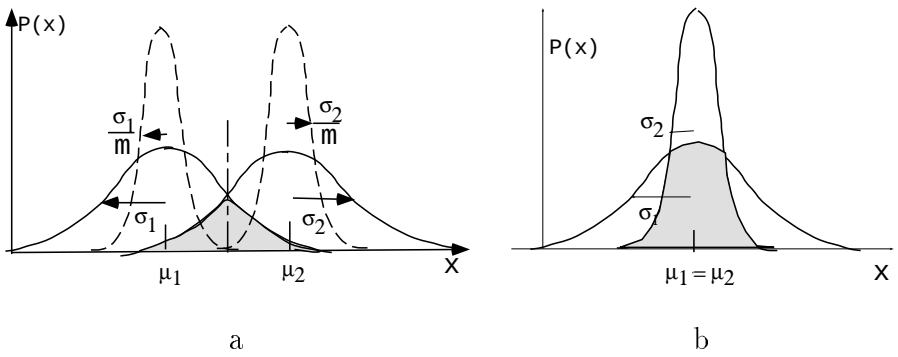


Figure 3 Two possible distribution for adjacent regions : a. Solid lines show two Gaussian distribution with large overlap due to noise fluctuation, the shadowed area displays the classification errors. Dash lines show the distribution after using window sampling. b. Two Gaussian distributions with same mean but different variances.

Second, in the extreme case where two distributions have the same mean but different variances, the classification error will be intolerable, see Figure (3.b). One such image is studied in Figure (5). In such cases, we need to measure the second order statistics (variance) near each boundary point (x, y) in order to tell whether the point (x, y) should be classified as belonging to one distribution or to the other. As we mentioned earlier, this cannot be done by Fisher's test. Furthermore, if the distribution $P(I|\alpha)$ is more sophisticated than the Gaussian distribution then more complex statistics must be computed in the neighborhood of (x, y).

At this point we should make it precise that the problems of misclassification are not the fault of the original energy functional (equation (7)), in fact, we can prove even in cases where intensities in two adjacent regions have same mean but different variances, the global minimum of equation (7) still gives the correct segmentation.

The problems of misclassification are due to the greed of the algorithm, and the hardship in reaching the global minimum of equation (7).

Our method for dealing with the above problems is to use a circular window of m pixels around each boundary point (x, y). We call these pixels the neighbor set of (x, y) denoted by $\mathcal{W}_{(x,y)}$. The effect of this is to replace $P(I_{(x,y)} : \alpha)$ by the joint probability: $\prod_{(u,v)\in\mathcal{W}_{(x,y)}} P(I_{(u,v)} : \alpha)$. divided by the size of window m.

Correspondingly, the energy function becomes:

$$E[\Gamma, \{\alpha_i\}] = \sum_{i=1}^{M} \{\frac{\mu}{2} \oint_{\partial R_i} ds - \int\int_{R_i} \frac{1}{m} \int\int_{\mathcal{W}_{(x,y)}} \log P(I_{(u,v)} : \alpha_i) du dv dx dy + \lambda\}, \quad (18)$$

The window $\mathcal{W}$ solves our problems. First we observe that the larger m is, then the bigger the chance that it is representative of the distribution, and the smaller the risk of misclassification. Second with such samples, we can do higher order statistics over $\{I_{(u,v)} \mid (u, v) \in \mathcal{W}_{(x,y)}\}$ to answer the second problem raised earlier. But the window size m cannot be too big, otherwise the boundary will not be located precisely. Later in section (4) we will further analyze the relationship between the precision of the boundary and the choice of optimal windows.

To interpret equation (18) in the sense of MDL, we observe that each pixel is coded m times, then the final code length is divided by m. Therefore by exchanging the order of integral in the second term, we can rewrite equation (18) as:

$$E[\Gamma, \{\alpha_i\}] = \mu \oint_{\Gamma} ds - \int\int_{R} \sum_{i=1}^{M} \pi_{i(x,y)} \log P(I_{(x,y)} : \alpha_i) dx dy + \lambda M \quad (19)$$

$\Gamma = \cup_{i=1}^{M} \partial R_i$, and $\pi_{i(x,y)} = \frac{\|R_i \cap \mathcal{W}_{(x,y)}\|}{m}$ such that $\sum_{i=1}^{M} \pi_{i(x,y)} = 1$ for all (x, y). In other words, each pixel (x, y) in the image uses a mixture code. The properties of such mixture code are available in [31]. As seen in figure 4, point A is totally inside region R_i, so $\pi_i = 1, \pi_j = 0, \forall j \neq i$. Point B is near the boundary, so the corresponding π_i, π_j are shown by the shadow areas. Similarly for the point near the junction.

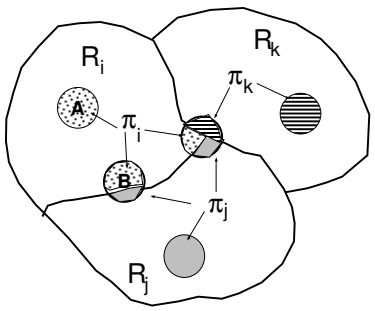


Figure 4 Each pixel is coded by the mixture distribution with π_i being the shaded proportions inside the window.

In the rest of this section, we will explore equation (18) by assuming Gaussian distributions.

Suppose we have window sample $\{I_{(u,v)} \mid (u,v) \in \mathcal{W}_{(x,y)}\}$, with each $I_{(u,v)} \sim N(\mu, \sigma^2)$ shown by the solid curves in Figure (3.a), then we can calculate the quantities $\bar{I}_{(x,y)}$ and $S^2_{(x,y)}$ as the sample mean, and sample variance for $\{I_{(u,v)}\}$, then $\bar{I}_{(x,y)} \sim N(\mu, \sigma^2/m)$, the distribution shown by the dashed lines in Figure (3.a). Because the variances have been divided by m, the misclassification risk is greatly reduced.

We can rewrite the statistics force generated by a single region R_i at (x,y) in the direction of $\vec{\mathbf{n}}_i$ as the following:

$$\begin{aligned}
\frac{1}{m}\int\int_{\mathcal{W}_{(x,y)}} \log P(I_{(u,v)} \mid \alpha_i) du dv &= \frac{1}{m} log \prod_{(u,v)\in\mathcal{W}_{(x,y)}} P(I_{(u,v)}|\mu_i, \sigma_i^2) \\
&= \frac{-1}{m} log(2\pi\sigma_i^2)^{\frac{m}{2}} \sum_{(u,v)\in\mathcal{W}_{(x,y)}} \frac{(I_{(u,v)} - \mu_i)^2}{2\sigma_i^2} \\
&= -\frac{1}{2}(log(2\pi\sigma_i^2) + \frac{(\bar{I} - \mu_i)^2}{\sigma_i^2} + \frac{S^2}{\sigma_i^2}) \qquad (20)
\end{aligned}$$

In equation (20), the second term tests the mean, and the last term tests the variance. With the addition of this "variance" force, we can detect two regions with the same means but different variances, see Figure (3.b). Equation (20) will be generalized to higher dimension in the next section (see Equation (39)).

Thus from equation (18), we can derive the motion equation for point $\vec{v}$ at the boundary $\Gamma_i \cap \Gamma_j$ by plugging equation (20) into the motion equation (14).

$$\frac{d\vec{v}}{dt} = -\mu\kappa_{i(\vec{v})}\vec{\mathbf{n}}_{i(\vec{v})} - \frac{1}{2}\{\log\frac{\sigma_i^2}{\sigma_j^2} + \big(\frac{(\bar{I}-\mu_i)^2}{\sigma_i^2} - \frac{(\bar{I}-\mu_j)^2}{\sigma_j^2}\big) + \big(\frac{s^2}{\sigma_i^2} - \frac{s^2}{\sigma_j^2}\big)\}\vec{\mathbf{n}}_i, \quad (21)$$

Before describing the complete region competition algorithm at the end of this section, we first explain how to decrease the energy by merging two adjacent regions. Suppose that $n_i, n_j, \mu_i, \mu_j, \sigma_i, \sigma_j$ are the pixel number, intensity mean, and intensity variance of two adjacent regions R_i, R_j respectively. Let $R_{ij} = R_i \cup R_j$ be the region after merging, let $\ell_{ij} = \partial R_i \cap \partial R_j$ be the common boundary between R_i and R_j, and let $\alpha_{ij} = (\mu_{ij}, \sigma_{ij})$ be the mean and variance of R_{ij}. It is easy to see that :

$$\mu_{ij} = \frac{1}{n_i + n_j}(n_i\mu_i + n_j\mu_j)$$
$$\sigma_{ij}^2 = \frac{1}{n_i + n_j - 1}\{(n_i - 1)\sigma_i^2 + (n_j - 1)\sigma_j^2 + \frac{n_i n_j}{n_i + n_j}(\mu_i - \mu_j)^2\}$$

Then the change of energy for merging R_i, R_j is:

$$\begin{aligned}\Delta E &= -\mu\int_{\ell_{ij}} ds - \lambda + \int\int_{R_i} logP(I|\alpha_i)dxdy + \int\int_{R_j} P(I|\alpha_j)dxdy - \int\int_{R_{ij}} logP(I|\alpha_{ij})dxdy \\ &= -\mu \mid \partial R_i \cap \partial R_j \mid -\lambda + \frac{1}{2}(n_i log\frac{\sigma_{ij}^2}{\sigma_i^2} + n_j log\frac{\sigma_{ij}^2}{\sigma_j^2} + 1) \quad (22)\end{aligned}$$

Equation (22) is a generalization of equation (4). If $\Delta E \leq 0$, then merging R_i and R_j will decrease the energy. A brief algorithm is given below:

the region competition algorithm:

1. initialize the segmentation , we can put N seeds randomly across the image, and all background (area not occupied by any seed regions) is treated as a single region with uniform probability distribution.
2. Fix the boundary Γ, compute the parameters $\{\alpha_i\}$ by maximizing P(I:α_i). in this step, we may need to fit some surface or physical model (see color model).
3. Fix $\{\alpha_i\}$, move the boundary Γ by minimizing the energy function. When two seed regions meet, edge is formed at their common boundary, then these two regions compete along this boundary.
4. execute step 2, 3 iteratively until the motion of boundary Γ converges. Then goto step 5.
5. If there is background region not occupied by any seed regions, then put a new seed in the background, and goto step 2; else goto step 6.
6. Merge two adjacent regions so that the merging causes the largest energy decrease, goto step 2. if no merge can decrease the energy, then goto step 7.
7. stop.

3.3 Simulations on Grey Level Images

In this subsection, we show how to implement the region competition algorithm on two typical grey level images for the purpose of illustration.

Figure (5.a) is a 100×100 image partitioned into two regions by a S-shaped curve. The intensities in these two regions are generated randomly from two Gaussian distributions with identical means: $N(128, 10^2)$ and $N(128, 35^2)$. Suppose the image is initially segmented by a slanted straight line, see Figure (5.b). Then the motion of the regions at iteration steps $t = 0, 20, 30, 50$ are shown in Figure (5.b.c.d.e) respectively. Finally Figure (5.f) shows how the boundary moves if we drop the last term in Equation (21). Thus if we do not test the variance then the edge just moves to the low-right corner and disappears. This results in an incorrect segmentation with the image being perceived as a single region. In this image, $\mathcal{W}$ is a circular window which contains 32 pixels.

Figure (6.a) is a 150×150 image consists of 4 regions whose intensities are gen-

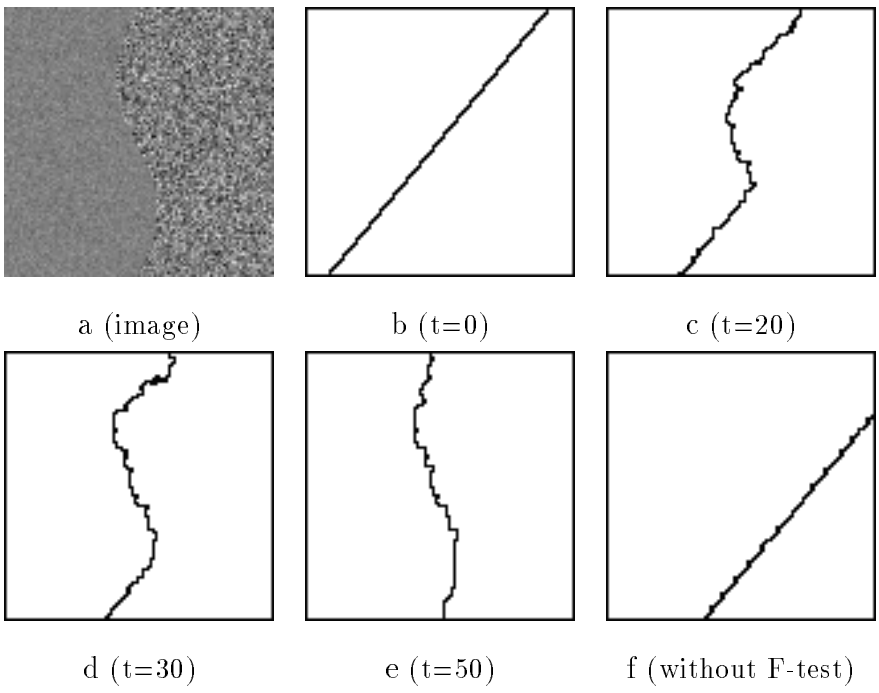


Figure 5 Two regions with identical means but different variances. See text for interpretation

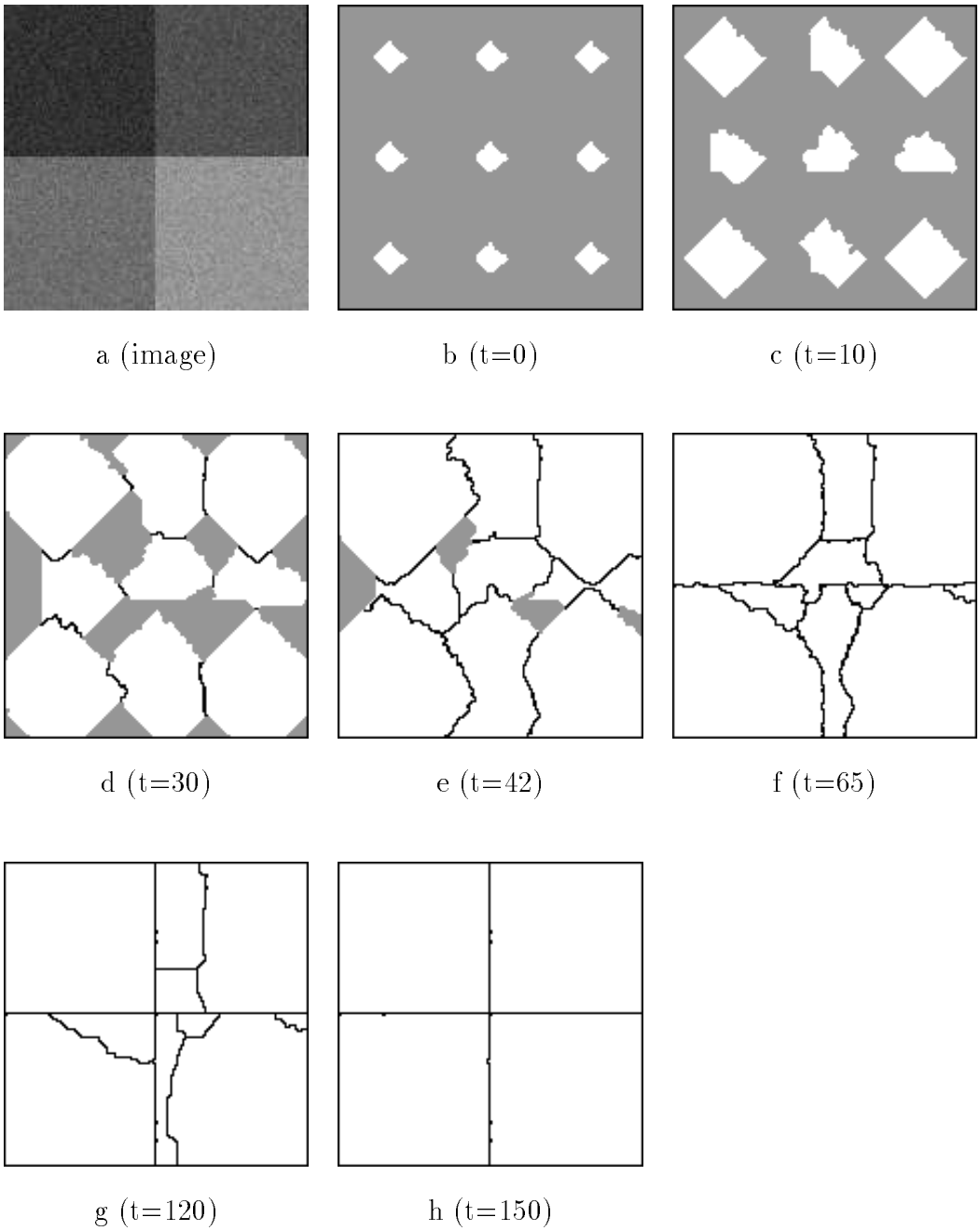


Figure 6 The competition of *good* and *bad* seed regions. See text for interpretation

erated randomly from 4 Gaussian distributions: $N(50, 10^2), N(80, 10^2), N(110, 11^2)$, and $N(150, 15^2)$. The segmentation is started by putting nine seed regions (each has 80 pixels) on a grid, with four seeds inside the four regions, four seeds straddling the boundaries, and one seed lies on the intersection point of the four regions. As we will define in the next section, the first four regions are called the *good seeds*, while the later five ones the *bad seeds*. The background (shown by the shadow) is treated as a single region which has uniform probability distribution P_0. In fact P_0 provides the forces causing the seed regions to grow.

The motion of regions at iteration steps $t = 0, 10, 30, 42, 65, 120, 150$ are shown in Figures (6.b.c.d.e.f.g.h) respectively. At $t = 10$ we observe that the good seeds grow faster than the bad seeds. Where these regions meet edges are formed, see Figure (6.d). Then the four good seeds keep compressing the bad regions and taking over the background, see Figure (6.e.f.). In this competition process, some regions may be squeezed out or split by their neighbors (see step t=42). Finally the competition converges at $t = 120$ in Figure (6.g), where all the bad seed regions are driven into one of the four real regions. Then we merge the pairs of adjacent regions which cause the energy function to decrease the most, and restart region competition. After 30 more iteratives, the algorithm converges at $t = 150$, see Figure (6.h).

4 Analyzing the Boundary and Windows

In section (3.2), we argue that a window $\mathcal{W}_{(x,y)}$ is necessary to reduce the error of classification (see figure(3)). In this section, we further analyze the precision of the boundary Γ calculated by the region competition algorithm. It turns out that there exists an uncertainty interval $\mho$ for the boundary, and $\mho$ cannot be minimized unlimitedly through the choice of windows. Then we demonstrate how the size and shape of the window $\mathcal{W}(x, y)$ are selected to optimize this uncertainty interval. Finally we briefly discuss the connection between region competition and

edge detection.

4.1 Precision of the Boundary

To understand how precise the boundary Γ is, we need to analyze the behaviors of the statistics force[5] around the true boundary of two adjacent regions. Suppose that $\Gamma_{12} = \partial R_1 \cap \partial R_2$ is the common boundary between region R_1 and R_2, and within each region the $I_{(x,y)}$'s are subject to probability distribution $P(I_{(x,y)} : \alpha_1)$ and $P(I_{(x,y)} : \alpha_2)$ respectively. Then we can write the statistics force as $f_{(x,y)}\vec{\mathbf{n}}_1$, with

$$f_{(x,y)} = \frac{1}{m} \sum_{(u,v)\in\mathcal{W}(x,y)} \log^{\frac{P(I_{(u,v)}:\alpha_1)}{P(I_{(u,v)}:\alpha_2)}}$$

where m is the size of window $\mathcal{W}_{(x,y)}$. Obviously, $f_{(x,y)}$ is a random variable depending on the location (x, y). Since the boundary Γ_{12} is located when $f_{(x,y)} = 0$, the distribution of $f_{(x,y)}$ will clearly indicate the precision of the boundary location. As an example, in the rest of this section, we will derive the distribution of $f_{(x,y)}$ in case that the intensities in both region R_1 and R_2 are subject to Gaussian distributions.

Figure (7.a) shows two ideal regions lying in the left and right half-plane, where $x = 0$ is the "true" boundary, and the dashed line is the current moving boundary Γ_{12}. The intensity distributions are $N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$ respectively, as shown in Figure (7.b). If R_1 and R_2 are big enough, then we assume that the true distributions $(\mu_1, \sigma_1), (\mu_2, \sigma_2)$ are known to the algorithm by minimizing the energy with respect to the μ's and σ's. Due to symmetry, without loss of generality, we can reduce $f_{(x,y)}$ to $f_{(x)}$, i.e. analyzing it along the x-axis. Now the force is:

$$f_{(x)} = \frac{a}{2} \ln^{\frac{\sigma_2^2}{\sigma_1^2}} + \frac{a}{2m} \sum_{(u,v)\in\mathcal{W}_{(x)}} \left(\frac{(I_{(u,v)} - \mu_2)^2}{\sigma_2^2} - \frac{(I_{(u,v)} - \mu_1)^2}{\sigma_1^2}\right) \tag{23}$$

where $a = log_2 e$ is a constant. It is easy to calculate the expectation of $f_{(x)}$ as

[5] Here we exclude the smoothing force, since it represents the geometry prior, and will bring bias to the boundary location.

follows.

$$E[f_{(x)}] = \frac{a}{2}[ln^{\frac{\sigma_2^2}{\sigma_1^2}} + \Pi_{(x)}(\frac{\Delta\mu^2}{\sigma_2^2} + \frac{\sigma_1^2}{\sigma_2^2}) - (1 - \Pi_{(x)})(\frac{\Delta\mu^2}{\sigma_1^2} + \frac{\sigma_2^2}{\sigma_1^2}) + (1 - 2\Pi_{(x)})] \quad (24)$$

where $\Delta\mu = \mu_2 - \mu_1$, and $\Pi_{(x)} = \frac{\|\mathcal{W}_{(x)} \cap R_1\|}{\|\mathcal{W}_{(x)}\|}$ is the proportion of the window inside R_1. Let ω be the width of the window, then $\Pi_{(x)} = 1$ for $x < -\frac{\omega}{2}$, $\Pi_{(x)} = 0$ if $x > \frac{\omega}{2}$, and $f_{(0)} = \frac{1}{2}$ if $\mathcal{W}_{(x)}$ is symmetric.

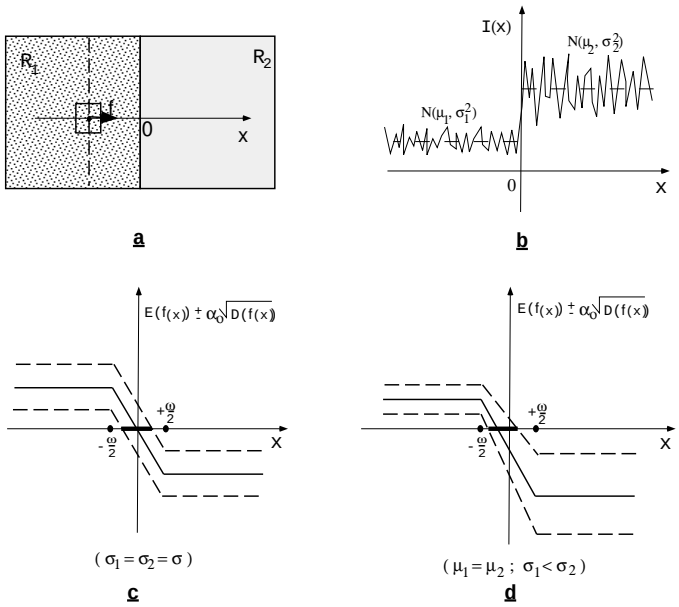


Figure 7 The expectations and variances of the statistics force $f_{(x)}$. See text for interpretation.

To further analyze the properties of the statistics force, we consider the following two typical cases.

Case 1: $\sigma_1 = \sigma_2 = \sigma$. In this case, $f_{(x)}$ is subject to a normal distribution with mean $E[f_{(x)}] = (2\Pi_{(x)} - 1)\frac{a\Delta\mu^2}{2\sigma^2}$ and variance $D[f_{(x)}] = \frac{a^2\Delta\mu^2}{m\sigma^2}$. In figure (7.c), the solid lines show $E[f_{(x)}]$, and the dashed lines show the confidence interval $E[f_{(x)}] \pm \alpha_o\sqrt{D[f_{(x)}]}$, in other words, the up and low bound of the fluctuation of force. For example, let $\alpha_o = 1.95$, then we can claim that the probability that the $f_{(x)}$ falls between the dashed lines is 0.95.

In figure (7.c), $f_{(x)} > 0$ means the boundary moves to the right, while $f_{(x)} < 0$ means the boundary moves to the left. The final boundary will be located at x where $f_{(x)} = 0$. From figure (7.c) we observe the following properties of the boundary Γ_{12} calculated by the region competition algorithm when $\sigma_1 = \sigma_2$.

- *Fact 1:* Since $E[f_{(0)}] = 0$, then Γ_{12} is an unbiased estimate for the true boundary.

- *Fact 2:* The confidence interval of Γ_{12} is defined by the intersections of the two dashed lines with the x-axis, shown by the thick line segment. The length of this thick line segment is:

$$\mho = 2\alpha_o \frac{\sigma}{\Delta\mu}\omega \tag{25}$$

Case 2: $\mu_1 = \mu_2, \sigma_1 = \sigma, \sigma_2 = k\sigma, k > 1$. In this case, $f_{(x)}$ is subject to a χ^2-distribution. with

$$E[f_{(x)}] = \frac{a}{2}[\ln k^2 + \Pi_{(x)}(k^2 + \frac{1}{k^2} - 2) + 1 - k^2]$$

Then we observe:

$$E[f_{(x)}] = \begin{cases} \frac{a}{2}(\ln k^2 + \frac{1}{k^2} - 1) > 0 & x < -\frac{\omega}{2} \\ \frac{a}{2}(\ln k^2 + 1 - k^2) < 0 & x > \frac{\omega}{2} \end{cases}$$

$$D[f_{(x)}] = \begin{cases} \frac{a^2}{2m}(1 - \frac{1}{k^2})^2 & x < -\frac{\omega}{2} \\ \frac{a^2}{2m}(k^2 - 1)^2 & x > \frac{\omega}{2} \end{cases}$$

This is shown in figure (7.d). Since $E[f_{(0)}] \neq 0$, the resulting boundary will be a biased estimate of the true boundary. For example, if $k = 2$, i.e. $\sigma_2^2 = 4\sigma_1^2$, then $E[f_{(x)}]$ will intersect the x-axis at -0.38ω. Thus the boundary will be moved to the region which has less variance. As we will discuss below, the window width ω should increase with the intensity variance σ^2, so in cases where σ^2 is really big, such as in the texture case discussed in the next section, the bias of the estimation will be noticeable. Again, as in case 1, the confidence interval of the boundary is shown by the thick line segment in figure (7.d).

4.2 The Size and Shape of the Window

Now let's analyze how to choose the size and shape of the window $\mathcal{W}_{(x)}$ to minimize the uncertainty (or confidence) interval $\mho$.

First of all, since $D[f_{(x)}]$ is a monotonic decreasing function of the window size m, the larger the window, the less the force fluctuates, and the closer the dashed lines are to the solid line. From figure (7.c. and d) we see that window size m has to satisfy the following constraints:

$$E[f_{(x)}] - \alpha_o\sqrt{D[f_{(x)}]} \geq 0 \qquad \forall x < -\frac{\omega}{2} \tag{26}$$

$$E[f_{(x)}] + \alpha_o\sqrt{D[f_{(x)}]} \leq 0 \qquad \forall x > \frac{\omega}{2} \tag{27}$$

Otherwise, the uncertainty interval of the edge (i.e. the thick line segment) will be infinitely long, in other words, the boundary may be located at any point with a certain probability. For instance, in the case of figure (7.c), solving the above inequalities gives $m \geq \frac{4\alpha_o^2\sigma^2}{\Delta\mu^2}$. Notice that $\frac{\Delta\mu^2}{\sigma^2}$ is the signal-to-noise ratio (SNR), clearly in an image with high contast regions (i.e. high SNR) where $\Delta\mu >> \sigma$, we can simply choose $m = 1$. This explains why the region growing algorithms works for certain images

Secondly, further increasing the size of the window will not improve the boundary uncertainty. As seen in equation (25), the uncertainty interval $\mho$ of the boundary will increase as the window width ω becomes bigger. For a square window $\omega = \sqrt{m}$, and for a circular window $\omega = 2\sqrt{m/\pi}$. Figure (8) shows the intuitive idea. When m increase, the fluctuation decreases, but $E[f_{(x)}]$ becomes more horizontal around zero. This makes the thick line segment grow.

In summary, the above two points indicate that the uncertainty of the boundary cannot be reduced without limit by selecting the window. This corresponds to the general uncertainty principle in image processing[36]. The window size is chosen to be the smallest m which satisfies the inequalities (26,27).

So far the analyses are based on square windows, in fact, we are using circular

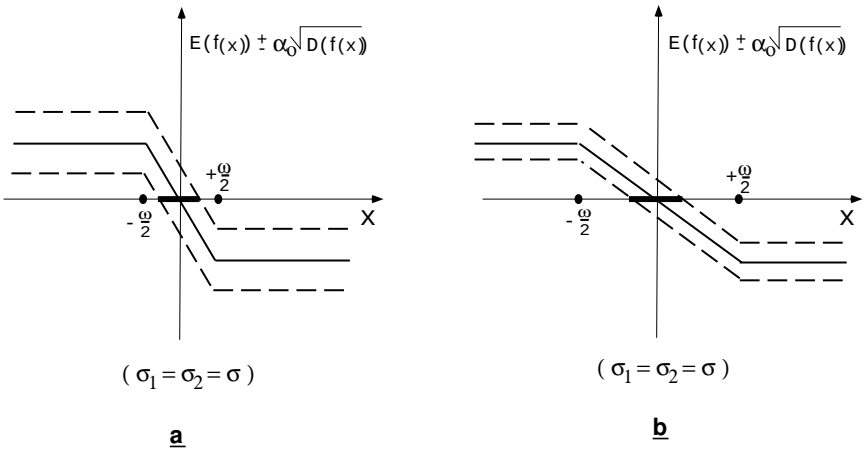


Figure 8 Compared with a, after increasing the window size in b, the variance of the force decreases, but the force becomes more horizontal around zero, thus the uncertainty interval of the boundary becomes even bigger.

---

windows. Then $\Pi_{(x)}$ will be slightly different for $x \in (-\omega/2, \omega/2)$. As shown in figure (9.b), it will no longer be a straight line. On the other hand, as seen in equation (25), in order to make $\mho$ smaller, we can deform the shape of the window so that the window width ω is smaller while keeping the size m unchanged. This can be done by choosing an ellipse window with its elongated axis aligned to the boundary (shown in figure (9.a)). But the aspect ratio of the ellipse will be bounded by the requirement for spatial resolution of the boundary. Further analysis about the optimal shape of the window is beyond the scope of this paper.

The analyses in this section provide two heuristics to improve the precision of the segmentation.

Heuristic 1. Using $f_{(x)} - E[f_{(0)}]$ as the statistics force such that the estimation is unbiased. $E[f_{(0)}]$ is given by plugging in $\Pi = \frac{1}{2}$ into equation (24) in case of the Gaussian models.

Heuristic 2. Using ellipse windows with their elongated axis aligned to the tangent of boundary Γ such that the uncertainty interval $\mho$ decreases.

But unfortunately, after these heuristics are used, it becomes unclear whether

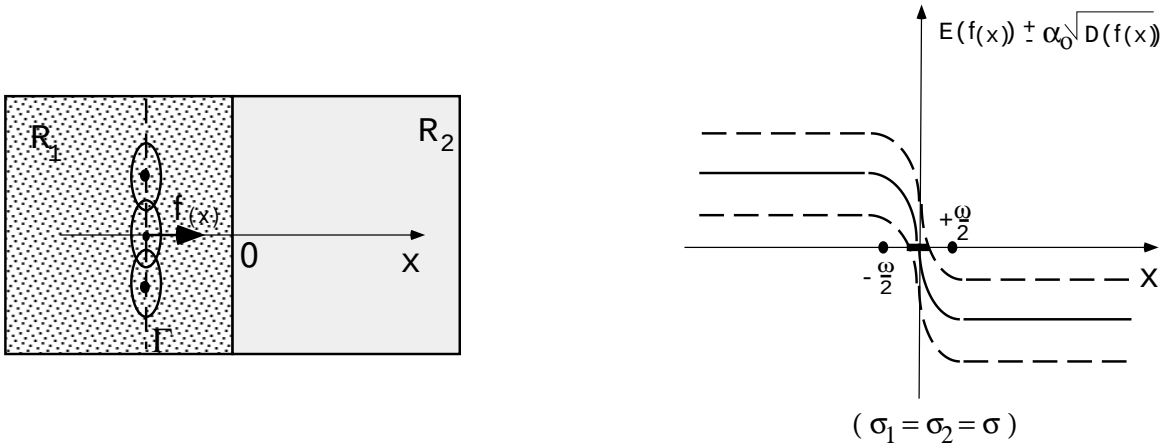


Figure 9 When an ellipse window is used with the same area m, the uncertainty interval shown by the thick line segment becomes smaller.

updating the boundary still minimizes an explicit global criterion. Enormous efforts have been made to write down the corresponding energy functional but they have been in vain. The major difficulty is that the window $\mathcal{W}_{(x,y)}$ will become a function of the boundary Γ, which make the calculus of variational difficult. We open this problem to the reader for further research.

The region competition can be related to the local edge detection approaches, if we think the window used here is a kind of edge detector, especially when heuristic 2 is used. By using various surface models in the two adjacent regions, our window can detect step edges, crease edges, or even c^2 edges. But such edge detection differs from the local edge filter in that we are using a global criterion.

5 The Analysis of Seeds

As we know, the performance of a greedy algorithm will highly depend on the initial conditions. As for region competition, this means that the choice of the seeds will affect the final segmentation. Although the region growing algorithm has been used for more than a decade, the analytical results about seed selection haven't been addressed in the literature. In this section, we discuss some empirical results about

how to choose seeds.

5.1 what is a "good" seed ?

In this paper, we assume the image consists of M homogeneous regions, and each region is generated by one of K families of pre-specified probability distributions. The precondition for an image segmentation algorithm to work is that the algorithm should know all the underlying models based on which the image regions are generated. In other words, if K models are not sufficient for an input image, then there is no reason to expect the algorithm to produce satisfactory segmentation. This is true for all image segmentation algorithms.

In fact, each initial seed is a hypothesis about the *location* of a certain *probability distribution family (or a model)*. Therefore, we define a seed to be "good", if

1. it is completely inside a 'true' region, and
2. it picks up the correct probability family (or model).

Failing in satisfying either of these two conditions, a seed (or a hypothesis) is 'bad'.

Thus to test whether a seed is good or not is to test how good the sample characteristics (in most case, the empirical distribution function) inside the seed fit a probability model. In statistics, there are many standard methods to test the goodness of fit, such as the moment test, Chi-square test, and Kolmogorov test [24]. The size of the seed will depend on the complexity of the probability distribution and on the image features based on which the hypothesis is made.

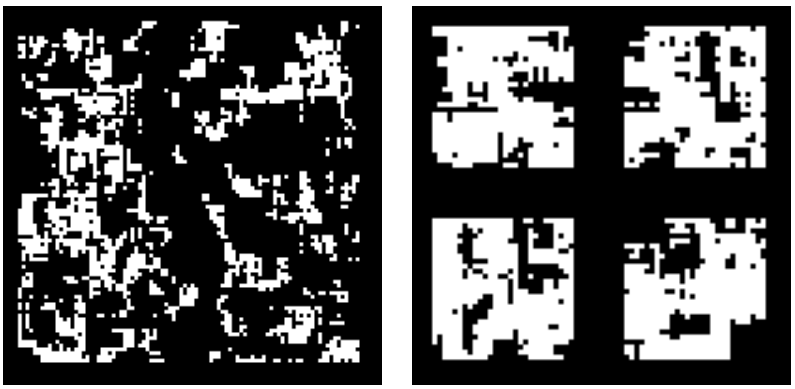
In the rest of this subsection, we show how to test the simplest probability family–the Gaussian distribution based on intensity. we choose the moment test.

Suppose at a seed location we have samples $\{x_1, x_2, ..., x_n\}$, then we measure the *skewness* g_1 and *kurtosis* g_2, with

$$g_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^3}{nS^{3/2}}$$

$$g_2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^4}{nS^2} - 3$$

where $\bar{x}$ and S are the sample mean and sample variance of the seed. If the seed is inside a true Gaussian region, then the asymptotic distribution of g_1 and g_2 are $N(0, 6/n)$ and $N(0, 24/n)$. By cutting the upper and lower $\alpha/2$ percentage points of both g_1 and g_2, we can reject the bad seeds effectively. figure (10.a,b) shows the results of the test on the two grey images shown in figure (5) and (6) respectively. We choose $\alpha = 20$, the white points indicate good seeds, while black point indicate rejected seeds. It is clear the the pixels around the boundaries ('S'-shape for a. crossing lines for b.) are all black.



a b

Figure 10 The normality tests. a). the good seeds for the same mean but different variance image. b). the good seeds for the four regions image

Empirically we observe that the sufficient condition for an optimal segmentation is that at the beginning each 'true' region includes at least one good seed. But this is not a necessary condition. The following subsection introduces an interesting domino effect of transferring bad seeds into good ones.

5.2 The Domino Effect of Seed Transferring

In figure (11), we show several steps of the region competition running on the image shown in figure (6). Now all the four seeds are bad ones, which striddle over the boundaries. But seed 1 is relatively better than the rest, since the two upper regions have less intensity difference. The final result is shown at $t = 180$ which is not a desirable segmentation.

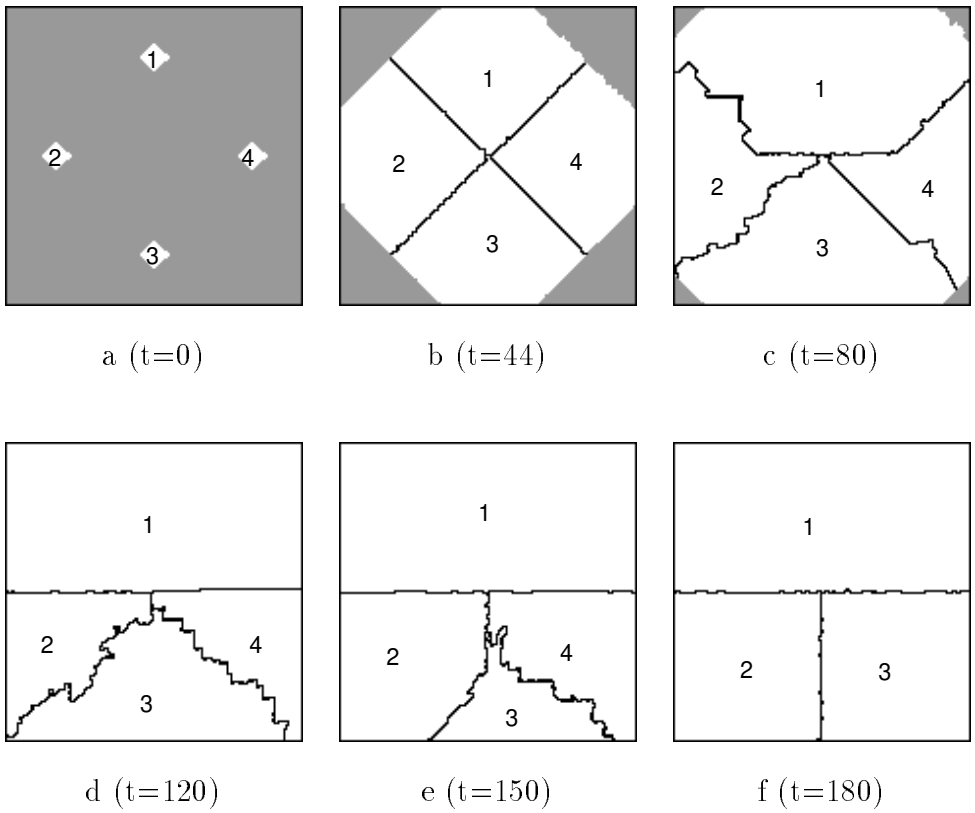


Figure 11 All four seeds are bad

Then we introduce one good seed–seed 5 at the upper-left region, we observe an interesting domino effect, as shown in figure (12). At the first stage, all bad seeds grow across the boundaries(see t=20). After they contact each other, seed 5 drives seed 1, and seed 2 out of the upper-left region (see t=50), then at time step $t = 84$,

we observe that seed 1 and 2 are completely driven out of the upper-left region and both become good seeds while seed 3 and 4 are still bad seeds. Once seed 1 and 2 become good, they keep kicking out seed 3 and 4 from their territories until time step t=126 when both seeds 3 and 4 are driven into the lower-right region, and both become good seeds. Finally they are merged into a single region at t=150. Such domino effects will continue when more regions are involved, and we think this is a key difference between the region competition algorithm with the ordinary region growing approaches. Further analysis about the sufficient and necessary conditions for choosing initial seeds to reach optimal segmentation is under investigation.

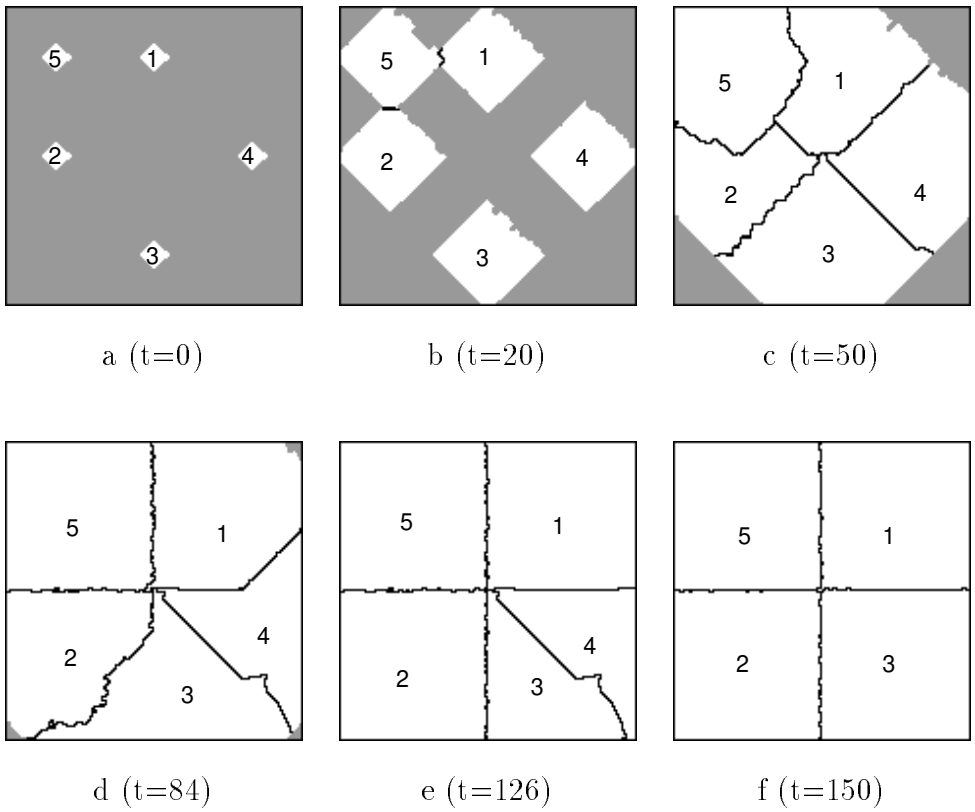


Figure 12 Four bad seeds and one good seed

6 Multivariate Models: Color

The previous sections have described segmentation techniques where regions are considered to have homogeneous intensity properties. But the image intensity is a function both of the material properties of the object, its albedo, and the surface orientation and lighting. This is expressed by the image irradiance equation:

$$I(\lambda, \vec{r}_i) = \rho(\lambda, \vec{r}_s) F(\vec{v}, \vec{n}, \vec{s}) E(\lambda, \vec{n}, \vec{s}) + noise(\vec{r}_i) \tag{28}$$

where I is the image as function of the image position $\vec{r}_i$ and wavelength λ, ρ is the surface albedo as a function of λ and surface location $\vec{r}_s$, F is the geometric viewing term as a function of λ the surface normal $\vec{n}$ and the light source direction $\vec{s}$, and E is the surface irradiance as a function of λ, $\vec{n}$ and $\vec{s}$.

For many applications image segmentation should be based on the albedo, because it corresponds to properties of the material independent of the lighting conditions or the geometric configuration of the material. For example, if we are looking at a person then the skin or each patch of the clothes will have constant albedo properties but the intensity of the skin or clothes may change dramatically due to the changes of the geometric configuration.

For grey level images obtaining the albedo from the intensity is an ill-posed problem. But the problem becomes tractable when we have color images as input, and it can be assumed that the geometric reflectance F is the same for all color bands. Color will allow us to segment regions with constant albedo.

6.1 The Color Model

If we include specularity in the image formation process then the color image irradiance equation is:

$$I(\lambda, \vec{r}_i) = \rho(\lambda, \vec{r}_s) F_b(\vec{v}, \vec{n}, \vec{s}) E(\lambda, \vec{n}, \vec{s}) + F_s(\vec{v}, \vec{n}, \vec{s}) E(\lambda, \vec{n}, \vec{s}) + noise(\vec{r}_i) \tag{29}$$

where the first term corresponds to the body reflectance, as in Equation (28), and

the second term corresponds to specularities (typically highlights).

If we adopt the red, green, blue color bands, we can simplify Equation (29) at each point (x, y) to:

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix}_{(x,y)} = \begin{pmatrix} r \\ g \\ b \end{pmatrix} I_{(x,y)} + \begin{pmatrix} r_s \\ g_s \\ b_s \end{pmatrix} I_{S,(x,y)} + \begin{pmatrix} e_r \\ e_g \\ e_b \end{pmatrix}_{(x,y)} \tag{30}$$

where (r, g, b) is the (assumed) spatially constant body color within each region, (r_s, g_s, b_s) is the (assumed) spatially constant color of the light source, $I_{(x,y)}$ is the illumination, $I_{S,(x,y)}$ is the specular component, and the $(e_r, e_g, e_b)_{(x,y)}$ are the residues (or noise).

Although there have been many attempts to separate the body color (r, g, b) from the specular color (r_s, g_s, b_s) [17] [34], these approaches mainly work on plastic and metal objects with nice illumination. If the noise level is quite high and the intensity around the specular regions are approximately saturated, such conditions are commonly observed for skin and clothes, it will be very difficult to separate out the specularity using these approaches.

For now we simply combine the highlights into the body reflectance and the residues (later we will show how our method will enable us to detect specularities) giving:

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix}_{(x,y)} = \begin{pmatrix} r \\ g \\ b \end{pmatrix} I_{(x,y)} + \begin{pmatrix} e_r \\ e_g \\ e_b \end{pmatrix}_{(x,y)} . \tag{31}$$

We now determine the residues by assuming that they are as small as possible, in the least squares sense, while requiring that Equation (31) is satisfied. The residues, the albedo (r, g, b) and the intensity $I(x, y)$ can be found by minimizing the sum of

the squared error over all point (x, y) within region R_i:

$$(r^*, g^*, b^*, I^*_{(x,y)}) = \arg\min_{r,g,b,I} \sum_{(x,y)\in R_i} \left| \begin{pmatrix} R \\ G \\ B \end{pmatrix}_{(x,y)} - \begin{pmatrix} r \\ g \\ b \end{pmatrix} I_{(x,y)} \right|^2 . \tag{32}$$

We minimize this energy by steepest descent. The solutions, $(r^*, g^*, b^*, I^*_{(x,y)})$, at the minimum must satisfy:

$$(r^*, g^*, b^*) = \frac{\int\int_{R_i} I^*_{(x,y)} (R, G, B)_{(x,y)} dxdy}{\int\int_{R_i} {I^*_{(x,y)}}^2 dxdy}$$
$$I^*_{(x,y)} = (r^*, g^*, b^*) \cdot (R, G, B)^\tau_{(x,y)} \tag{33}$$

where we have imposed the normalization constraint $r^{*2} + b^{*2} + g^{*2} = 1$.

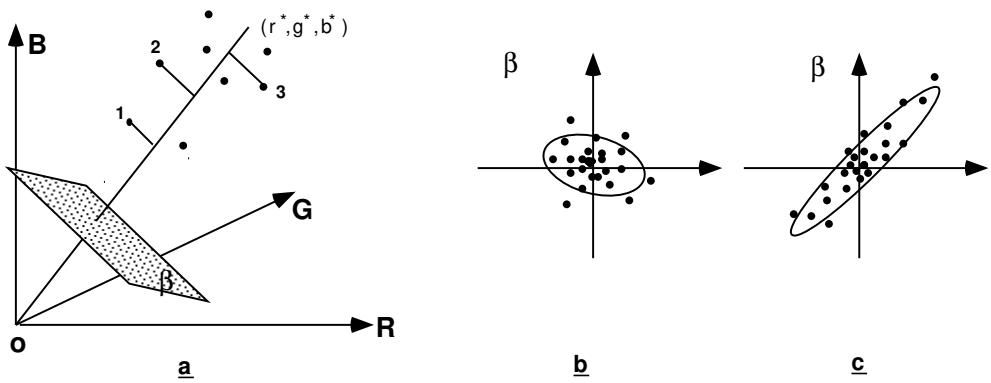


Figure 13 The Color models, a). fitting the irradiance equation, b).residuals distribution for homogeneous color regions, c).residual distribution for a highlight regions

---

In other words, as seen in Figure (13a), we find a unit vector (r^*, g^*, b^*) in the RGB space and project each pixel color vector $(R, G, B)_{(x,y)}$ onto this vector. The projected length will be their inner product, i.e. $I_{(x,y)} = (r^*, g^*, b^*) \cdot (R, G, B)^\tau_{(x,y)}$. Then the residue vector $(e_r, e_g, e_b)_{(x,y)}$ is perpendicular to the color vector shown in Figure (13a). Therefore all the residuals lie on a 2D plane β whose normal is (r^*, g^*, b^*).

Once we find the (r^*, g^*, b^*) for a region we transfer the residuals $(e_r, e_g, e_b)_{(x,y)}$ for each pixels (x, y) onto a two dimensional residual vector $\vec{X}$ in the β plane. This

is done by defining:

$$\vec{X}_{(x,y)} = \begin{pmatrix} \cos\beta, -\sin\beta, 0 \\ \sin\beta\cos\alpha, \cos\beta\cos\alpha, -\sin\alpha \end{pmatrix} \begin{pmatrix} e_r \\ e_g \\ e_b \end{pmatrix}_{(x,y)} \tag{34}$$

with $\alpha = \tan^{-1}(b^*/\sqrt{r^{*2} + g^{*2}})$ and $\beta = \tan^{-1}(g^*/r^*)$.

It is easy to check that the two row vectors in the 2×3 matrix are perpendicular both to the vector (r^*, g^*, b^*) and to each other. In fact, these two row vectors define the axes of a coordinate frame in β.

By analyzing the distribution (histogram) of residuals $\{\vec{X}\}$ in plane β over our dataset of regions with constant albedo[6], we found that if the region size is large enough, say larger than 100 pixels, then these residuals can typically be modeled by a 2D normal distribution $N(\mu, \Sigma)$ with μ very close to zero. In our experiments, we calculate the 2×2 covariance matrix Σ. Σ can be intuitively represented by an ellipse in plane β with its eigenvalues λ_1, λ_2 ($\lambda_1 \geq \lambda_2$) as the squared long and short axes of the ellipse and the rotational angle θ as the orientation of the ellipse.

Figure (13b) shows a typical residual distribution for a homogeneous color region. By contrast Figure (13c) is a typical residual distribution for a highlight region. The difference can be seen in the aspect ratio $a = \lambda_1/\lambda_2$. For a region without a highlight we typically find $1 \leq a \leq 2$, but in a highlight region we usually find $a \geq 4$. In other words, highlight residuals tend to have a larger component introduced by the specularity vector $I_s \cdot (r_s, g_s, b_s)^\tau$, i.e. the second term in equation (30). This property will be exploited later to allow us to detect highlight regions.

This color model has allowed us to decompose the input color image into a constant color term, a varying intensity term, and the residues on a 2D plane. Our method can be contrasted to the currently existing approaches, for example [34], which tries to eliminate the intensity term by normalizing the input color images, i.e.

[6] This dataset consisted of cloth and skin patches chosen from real images.

sending $R_{(x,y)} \mapsto \frac{R_{(x,y)}}{(R+G+B)_{(x,y)}}$. We claim that our method has advantages because the standard normalization becomes very noise sensitive at places where $(R+G+B)_{(x,y)}$ is small, but in our method such points have little weight, see Figure (13a) where pixels (1) and (2) have the same color but different intensity I – the residual error is smaller for (1) and therefore (1) has less weight in determining the region color (r, g, b). Both approaches, however, share the disadvantage that they cannot detect intensity boundaries, because they have factored out the intensity.

Our method for detecting specularities is different from the one developed in [17] which calculates the 3×3 covariance matrix on each patch independently and classifies the color patches into $2^3 = 8$ classes based on the size of the three eigenvalues. We argue that this approach is not applicable to our application and the physical meaning of the eigenvectors is not always clear. For example, in a region with constant color but heavy gradient shading the biggest eigenvalue will correspond to illumination changes while in a region with less shading but big noise fluctuations the largest eigenvalue will instead correspond to noise fluctuations. For these reasons we argue that the eigenvalue analysis should be done in 2D instead of 3D.

6.2 Color Segmentation

In color images, the probability family for each homogeneous region R_i includes the color model, therefore it is defined as $P((R,G,B)_{(x,y)} : (\vec{\mu}_i, \Sigma_i), (r_i, g_i, b_i))$ (see appendix II). As before the region competition algorithm iterates the following two stages. In the first stage, we fix the boundary Γ, and compute the best fitting of color (r_i^*, g_i^*, b_i^*) for each region R_i. Then we calculate the fitting residues $\vec{X}_{(x,y)}$ on the β_i plane, whose normal is (r_i^*, g_i^*, b_i^*), for each $(x,y) \in R_i$. we define the mean and covariance of region R_i (with $|R_i| = N$):

$$\vec{\mu}_i = \frac{1}{N} \sum_{(x,y)\in R_i} \vec{X}_{(x,y)}, \quad \Sigma_i = \frac{1}{N} \sum_{(x,y)\in R_i} (\vec{X}_{(x,y)} - \vec{\mu}_i)(\vec{X}_{(x,y)} - \vec{\mu}_i)^\tau \qquad (35)$$

In appendix II, we show that the color (r_i^*, g_i^*, b_i^*) computed by the least square method described in section(6.1), and the sample mean μ and sample variance Σ maximize the probability $P((R,G,B)_{(x,y)} : (\vec{\mu}_i, \Sigma_i), (r_i, g_i, b_i))$ and thus minimize the energy functional, provided that we assume the components in the residues vectors have zero means and identical variances. Empirically this assumption is usually well observed in color images. If the assumption is not observed, our approach still maximize the probability approximately, and this approximation is good for the situations we consider and significantly reduces the computation. People who are interested in the details are referred to appendix II.

The rest of this subsection explains the second stage, which is to move the boundary Γ by minimizing the MDL criterion.

For a pixel $p = (u, v)$ on the boundary of region R_i we first calculate the projection of $(R,G,B)_{(u,v)}$ onto the (r_i^*, g_i^*, b_i^*) vector, and then we calculate the fitting residue $\vec{Y}_{(u,v)}$ in the β_i plane.

As for the grey level case, we sample a window containing m pixels and obtain a set of measurements $(\vec{Y}_1, ..., \vec{Y}_m)$. We define the probability for these pixels to belong to region R_i to be:

$$\prod_{j=1}^{m} P(\vec{Y}_j \mid \vec{\mu}_i, \Sigma_i, (r_i^*, g_i^*, b_i^*)) = \frac{1}{(\sqrt{2\pi})^m ||\Sigma_i||^{m/2}} \exp\{-\frac{1}{2}\sum_{j=1}^{m}(\vec{Y}_j - \vec{\mu}_i)^T \Sigma_i^{-1}(\vec{Y}_j - \vec{\mu}_i)\} \tag{36}$$

Plugging these distributions to equation (18), we can define a global optimization criterion for color.

Once again we can relate this approach to conventional region growing using statistical tests. We define the window mean and co-variance by:

$$\bar{\vec{Y}} = \frac{1}{m}\sum_{i=1}^{m}\vec{Y}_i, \quad S = \frac{1}{m}\sum_{i=1}^{m}(\vec{Y}_i - \bar{\vec{Y}})(\vec{Y}_i - \bar{\vec{Y}})^T. \tag{37}$$

These are sufficient statistics for the Gaussian distribution, thus:

$$\prod_{j=1}^{m} P(\vec{Y}_j \mid \vec{\mu}_i, \Sigma_i, (r_i^*, g_i^*, b_i^*)) = C \exp\{-\frac{m}{2}\{(\bar{\vec{Y}} - \vec{\mu}_i)^T \Sigma_i^{-1} (\bar{\vec{Y}} - \vec{\mu}_i) + tr(\Sigma_i^{-1} S)\}\} \tag{38}$$

where $C = \frac{1}{(\sqrt{2\pi})^m ||\Sigma||^{m/2}}$, and "$tr(.)$" denotes the matrix trace.

Thus the statistics force generated by single R_i at (x, y) is $f_i \cdot \vec{\mathbf{n}}_i$, with:

$$f_i = -\frac{1}{2}(\log(2\pi) + \log ||\Sigma|| + (\bar{\vec{Y}} - \vec{\mu}_i)\Sigma_i^{-1}(\bar{\vec{Y}} - \vec{\mu}_i) + tr(\Sigma_i^{-1} S)) \tag{39}$$

Observe that Equations (39) generalizes equation (20) to higher dimensions.

As for the grey level case we can identify the terms on the right hand side of Equation (39) as the standard generalizations of the T and F tests to higher dimensions.

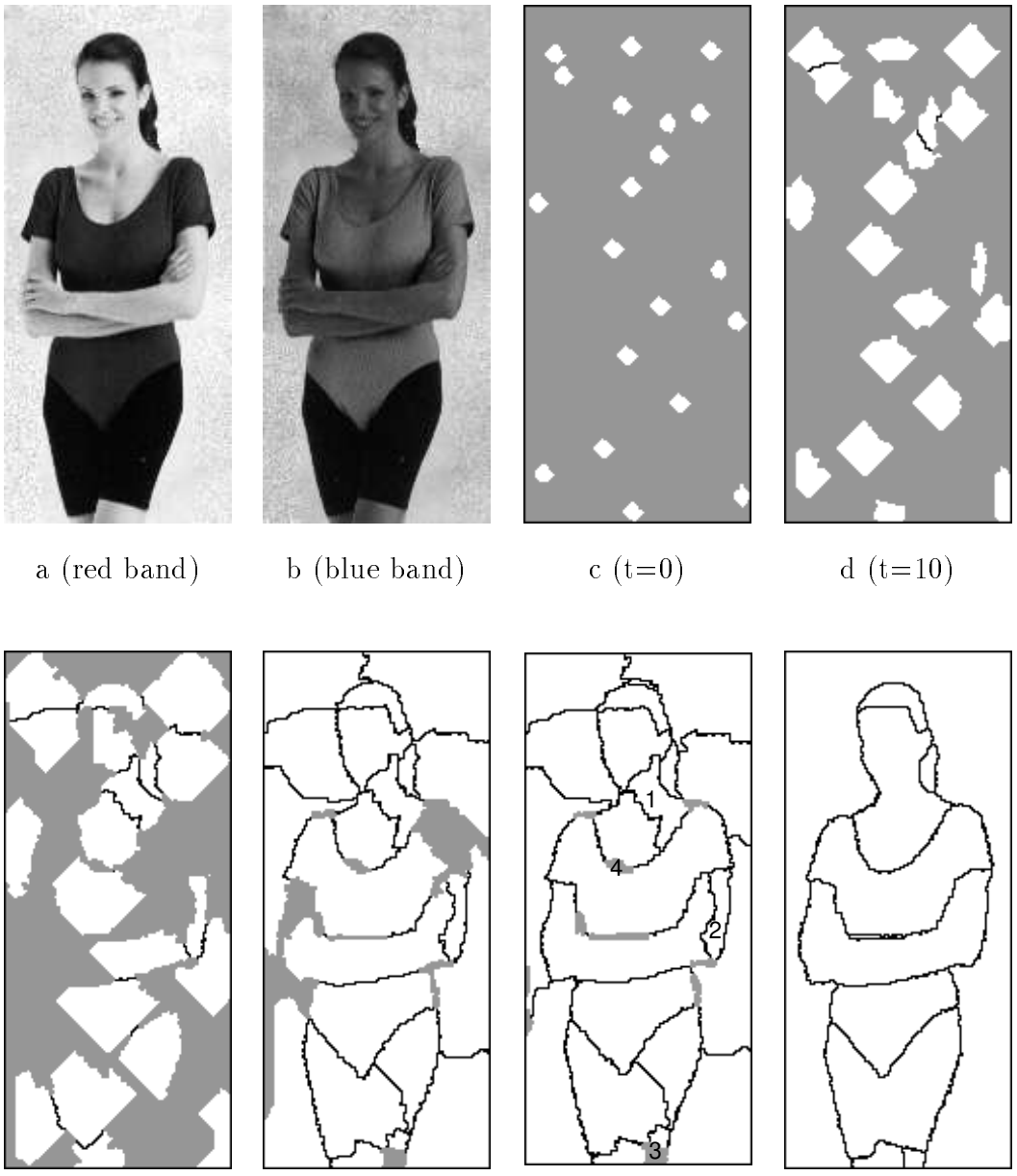
6.3 Color Images Experiments

Figures (14a,b) show the red and blue bands of a 261×116 color image. Observe the noise, shading in the cloth and the highlights on several parts of the skin.

The algorithm was implemented as for the grey level case. Figure (14c) shows the initial seeds at iteration step $t = 0$. Figures (14d,e,f,g,h) shows the sequence at steps $t = 10, 20, 60, 100, 135$ respectively. The competition between the seed regions converged at $t = 100$ as shown in Figure (14g) where the small shadow regions are the image spaces which are not occupied by the 20 seeds. In such regions we need to introduce additional seeds which can compete with the initial ones. We observe that the eyes and mouth are included in the face region because they are too small (about 10 25 pixels) and thus are treated as noise.

Observe in Figure (14g) that the algorithm detects the highlights correctly. These highlight regions are: (1) on the neck and shoulder, (2) on the left arm, (3) on the left leg, (4) on the breast.

Figure (14h) shows the final segmentation after spreading additional seeds and



a (red band) b (blue band) c (t=0) d (t=10)

e (t=20) f (t=60) g (t=100) h (t=130)

Figure 14 Segmentation of a color image

merging. In most case, it is desirable for the color model to eliminate shading effects, but in some cases, it become disadvantageous. For example, a black region like the hair will fit well to any color regions. In our implementation, we put one constraint in the merging step such that two regions with enormous intensity difference should not be merged. We consider it is a naive way for integrating the color and intensity clues. A more sophiscated data fusion algorithm is under implementation.

7 Texture Segmentation

We can directly adapt our algorithm to perform texture segmentation. First we apply a set of p texture filters to the input image to obtain a set of p texture images. This gives us a multi-band input $(s_1, ..., s_p)_{(x,y)}$ at each pixel (x, y). Then $\vec{X} = (s_1, ..., s_p)$ is considered to be a p-dimensional random variable at each pixel. We then apply our multi-band model, as in color segmentation, using the $\vec{X}$ as input.

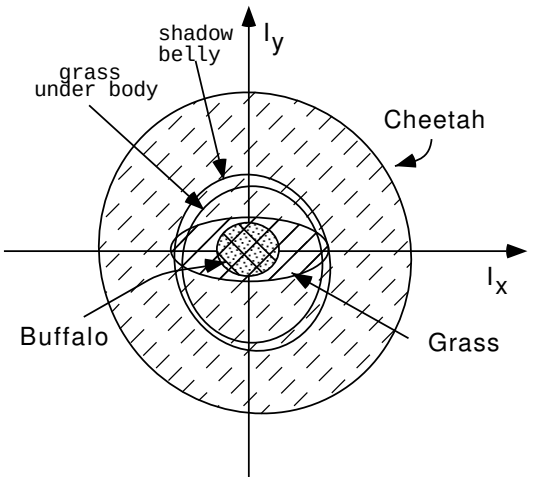
In this paper we set $p = 2$ and set $\vec{X} = (G_\sigma * I_x, G_\sigma * I_y)$ where G_σ is a zero mean Gaussian with standard deviation σ and $*$ denotes convolution, $(I_x, I_y) = \nabla I$ is the gradient of image I. This statistic captures the local orientation of the texture elements (or textons).

Figure (16a) shows a 172×247 textured image. For each region – such as the cheetah body, the grass and the buffalo– we study the second order texture statistics:

$$\bar{\vec{\mu}} = \frac{1}{N}\sum_{i=1}^{N}\vec{X}_i, \quad \Sigma = \frac{1}{N}\sum_{i=1}^{N}(\vec{X}_i - \bar{\vec{X}})(\vec{X}_i - \bar{\vec{X}})^\tau \tag{40}$$

The low half of figure (15) shows the histograms of $G_\sigma * I_x, G_\sigma * I_y$ on the three typical regions. Such Gaussian-like histograms indicate that the second order statistics are good discriminators. We found that the $\vec{\mu}$'s are very close to zero for all regions. The sample covariance Σ can be simply characterized by its eigenvalues λ_1, λ_2 and the direction θ between the largest eigenvector and the x axis. The upper

half of figure (15) shows the ellipse corresponding to various texture regions with $\lambda_1, \lambda_2, \theta$ listed on the right hand side.



	λ_1	λ_2	θ (degree)
grass	6.05	3.24	178.9
cheetah	14.4	13.9	130.5
buffalo	2.58	2.42	119.0
shadow belly	6.33	5.86	112.4
grass under body	5.24	4.58	96.6

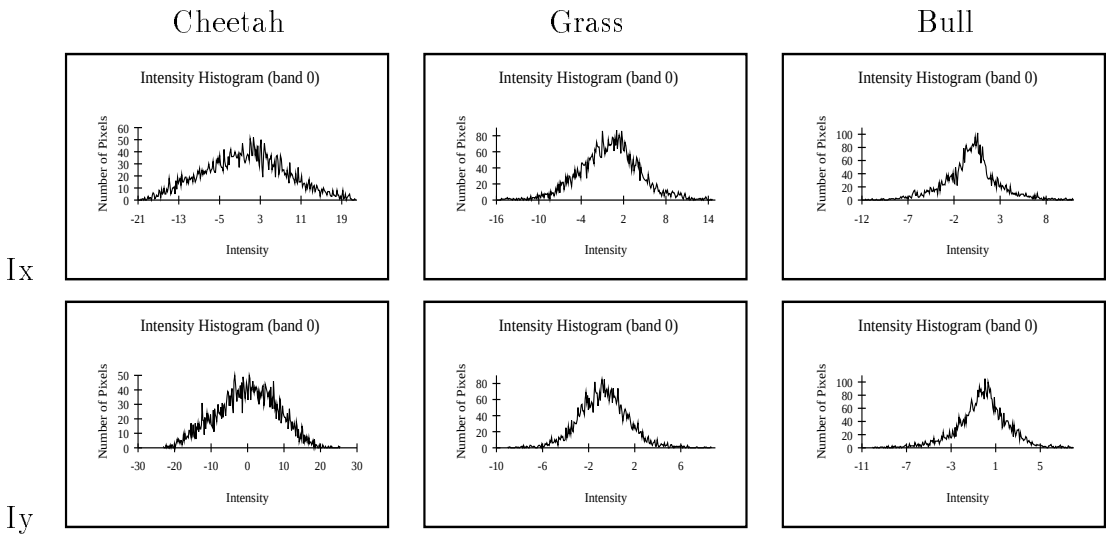


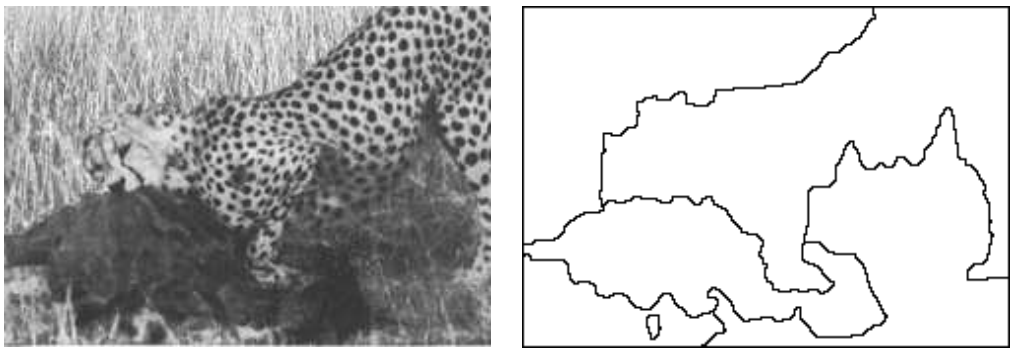
Figure 15 The statistics and histograms of $G_\Sigma * I_x$, $G_\Sigma * I_y$ on serveral regions of the texture image

Our algorithm proceeds as in the color case. First we compute the statistics $(\vec{\mu}_i, \Sigma_i)$ for each region R_i. The means for all the regions are all close to zero, see Figure (15), so the covariance is the major discriminator. Then we sample in a window surrounding a boundary pixel (x, y) and compute the sample mean and

the sample covariance. In order to obtain enough data to calculate these sample covariances reliably we choose the window size m to be large. We set $m = 68$ for our experiments.

These sample means and covariances can be used to compute the statistical forces exactly as in Equation (39). As before these forces, in conjunction with the smooth forces, will update the positions of the contour boundaries.

Figure (16b) is the final result after merging, where we observe that the belly of the cheetah, which is highly shadowed, is merged with the grass under the body. This is due to the similarity of their covariance matrices, see the corresponding ellipses shown in Figure (15). Since the variances change dramatically between regions, for example, the variance in the cheetah region is about 6^2 times larger than that in the bull region, as we analyzed in section (4.2), the bias of the boundary location will be noticable. In the images shon in this paper, we haven't used the heuristics.



a b

Figure 16 Segmentation of a texture image

This example has only used two texture filters. In general we would prefer to use a larger number of filters geared to detecting different texture patterns [22][25]. Moreover, with a larger number of filters it would be possible to perform a normal-

ization between texture bands, as we did for the color bands, and reduce the effect of gradients caused by shadows and geometric effects.

8 Conclusion

We have designed a novel algorithm, *region competition*, for image segmentation by minimizing a generalized MDL criterion. This algorithm combines some of the most attractive features of snakes/balloons and region growing. We have successfully demonstrated our approach on grey level images, color images, and texture images. For color images we have developed new ways to: (i) eliminate intensity gradients due to shadows and geometric factors, and (ii) to detect specularities.

We are currently extending this work by (i) integrating grey level, color and texture cues, (ii) integrating the filtering approaches to locate the edge more precisely. We are using this algorithm as a front end for our object recognition systems[38].

Acknowledgments

It is a pleasure to acknowledge many highly stimulating discussions with David Mumford, Alan Yuille and Tai sing Lee. Dr. Lee helps to run the program on the texture image. The author will also thanks Yingnian Wu for his consulting role in statistics. The author was supported by the NSF grant DMS-91-21266 to David Mumford, and this work was also supported in part by the Brown/Harvard/MIT Center for Intelligent Control Systems with U.S. Army Research Office grant number DAAL03-86-K-0171, and ARPA for an Air Force contract F49620-92-J-0466 to Alan Yuille.

Appendix I: Functional Derivatives

This paper involves taking functional derivatives of integrals along contours and integrals over regions. This appendix shows how to perform these derivatives.

Consider:

$$E[\Gamma_{(s)}] = \int\int_R f(x,y)dxdy, \tag{41}$$

where $\Gamma_{(s)} = \partial R = (x_{(s)}, y_{(s)})$ is the boundary contour of the region R, with $0 \leq s \leq \ell$ is the arc-length.

Green's theorem states that: for a planar region R, $(P_{(x,y)}, Q_{(x,y)})$ is any vector field with continuous first order derivatives, then

$$\int\int_R (\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y})dxdy = \int_{\partial R} Pdx + Qdy = \int_0^\ell (P\dot{x} + Q\dot{y})ds \tag{42}$$

where " denotes differentiation with respect to s, and $\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} = f(x,y)$. For example, $Q(x,y) = \frac{1}{2}\int_0^x f(t,y)dt$ and $P(x,y) = -\frac{1}{2}\int_0^y f(x,t)dt$ will satisfy the above requirement.

Thus, using (42), and let $L(x,\dot{x},y,\dot{y}) = Q(x,y)\dot{x} + P(x,y)\dot{y}$ we can write (41) as

$$E[\Gamma_{(s)}] = \int_0^\ell L(x,\dot{x},y,\dot{y})ds, \tag{43}$$

By Euler-Lagrange equation, we get the gradient of $E[\Gamma_{(s)}]$ with respect to any point $(x_{(s)}, y_{(s)}) \in \Gamma$:

$$\begin{aligned} \frac{\delta E}{\delta x} &= \frac{\partial L}{\partial x} - \frac{d}{ds}\frac{\partial L}{\partial \dot{x}} \\ \frac{\delta E}{\delta y} &= \frac{\partial L}{\partial y} - \frac{d}{ds}\frac{\partial L}{\partial \dot{y}} \end{aligned} \tag{44}$$

We find:

$$\begin{aligned} \frac{\partial L}{\partial x} - \frac{d}{ds}\frac{\partial L}{\partial \dot{x}} &= (\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y})\dot{y} = f(x,y)\dot{y} \\ \frac{\partial L}{\partial y} - \frac{d}{ds}\frac{\partial L}{\partial \dot{y}} &= (-\frac{\partial Q}{\partial x} + \frac{\partial P}{\partial y})\dot{x} = -f(x,y)\dot{x} \end{aligned} \tag{45}$$

Since $\vec{\mathbf{n}}_{(x,y)} = (\dot{y}, -\dot{x})$ is the normal along the contour Γ, let $\vec{v} = (x, y)$, thus we have:

$$\frac{\delta E}{\delta v} = f(x, y)\vec{\mathbf{n}}_{(x,y)} \tag{46}$$

To compute the functional derivative of an integral along the boundary contour we need start the derivation at Equation (43) by choosing $L = \sqrt{\dot{x}^2 + \dot{y}^2}$. It is straightforward, for example, to check that the functional derivative of the line integral $\int_{\partial R} ds$ is $-\kappa\vec{\mathbf{n}}$.

Appendix II: Is the least square fitting minimizing the energy functional ?

In section (6.1), we fit the color model, equation (31), by using a least squares fitting approach. This appendix is to show how least squares fitting is related to probabilistic estimation.

For brevity, let $\vec{\mathbf{X}}_{(x,y)} = (R, G, B)^{\tau}_{(x,y)}$ be the input at point (x, y), $\vec{\mathbf{C}} = (r, g, b)^{\tau}$ be the color for a region R_i, and $\vec{\mathbf{\Upsilon}}_{(x,y)} = (e_r, e_g, e_b)^{\tau}_{(x,y)}$ be the residue at (x, y). We can rewrite the equation (31) as:

$$\vec{\mathbf{X}}_{(x,y)} = I_{(x,y)} \cdot \vec{\mathbf{C}} + \vec{\mathbf{\Upsilon}}_{(x,y)} \tag{47}$$

where $\|\vec{\mathbf{C}}\| = 1$, and $I = \vec{\mathbf{C}}^{\tau} \cdot \vec{\mathbf{X}}$ is the illuminance. Then $\vec{\mathbf{\Upsilon}} = \vec{\mathbf{X}} - I\vec{\mathbf{C}} = (\mathcal{I} - \vec{\mathbf{C}}\vec{\mathbf{C}}^{\tau})\vec{\mathbf{X}}$, and $\vec{\mathbf{\Upsilon}} \perp \vec{\mathbf{C}}$. Here $\mathcal{I}$ is the 3×3 identity matrix. Thus we can rewrite the residue as $\vec{\mathbf{\Upsilon}} = e_1\vec{\mathbf{C}}_1^{\perp} + e_2\vec{\mathbf{C}}_2^{\perp}$, where $\vec{\mathbf{C}}_1^{\tau}, \vec{\mathbf{C}}_2^{\tau}$ are the unit axis vectors on the β plane (see figure (13a)). Because $\vec{\mathbf{C}}_1^{\perp} \perp \vec{\mathbf{C}}_2^{\perp}$ we have $\|\vec{\mathbf{\Upsilon}}\|^2 = e_1^2 + e_2^2$. We assume that the 2D residues $(e_1, e_2)^{\tau}$ can be modelled by a multivariate Gaussian distribution $\mathcal{N}(\vec{\mu}, \Sigma)$ and we can rotate $\vec{\mathbf{C}}_1^{\tau}, \vec{\mathbf{C}}_2^{\tau}$ such that Σ is diagonalized as $[\sigma_1^2; \sigma_2^2]$ and $\vec{\mu} = (\mu_1, \mu_2)$. It is easy to see that the least square fitting in equation (32) is actually equivalent to:

$$(r^*, g^*, b^*) = \arg\min_{r,g,b} \sum_{(x,y)\in R_i} \|\Upsilon_{(x,y)}\|^2 = \arg\min_{r,g,b} \{\sigma_1^2 + \mu_1^2 + \sigma_2^2 + \mu_2^2\}, \tag{48}$$

where, for the last equality, we have set $\vec{\mu} = 0$.

On the other hand, when we fix the segmentation Γ, then minimizing the energy functional with respect to $\alpha_i = (r_i, g_i, b_i, \vec{\mu}_i, \Sigma_i)$ $(i = 1, ..., M)$ is equal to maximizing the likelihood function $P(\{X_{(x,y)}, \forall (x,y) \in R_i\} \mid (r_i, g_i, b_i), (\vec{\mu}_i, \Sigma_i))$ for each $i = 1, ..., M$ respectively. It is straightforward to prove that for each region R_i the probability P reaches its maximum $\frac{1}{2\pi \|\Sigma\|^{\frac{1}{2}}} e^{-\frac{\|R_i\|}{2}}$, when μ, Σ are the sample mean and sample variance of the residues $\{(e_1, e_2)^\tau_{(x,y)} \ \forall (x,y) \in R_i\}$, where $\|R_i\|$ is the pixel number (or area) of R_i and $\|\Sigma\|$ is the determinant of Σ. Thus to maximize the probability P, we need to fit (r, g, b) such that:

$$(r^*, g^*, b^*) = \arg\min_{r,g,b} \|\Sigma\| = \arg\min_{r,g,b} \{\sigma_1^2 \cdot \sigma_2^2\} \tag{49}$$

We see that the (r, g, b) fitting criterion which we used in equation (48) or equation (32) is different from equation (49). But if we assume that $(e_1, e_2)^\tau$ has zero mean and $\sigma_1 = \sigma_2$, then the least square fitting that we used is equivalent to the criterion in equation (49). In fact, such an assumption is common in multi-variable statistics due to its simplicity. Empirically we find that our assumption is well satisfied in homogeneous color regions, the differences are slightly bigger in highlight regions. The key point is that the variations due to intensity fluctuations in the (r, g, b) directions are typically far larger than the variations in the residue directions. thus to determine these directions we only need to use the approximate criterion given by equation (48).

In the following, we derive the precise formulae for the covariance matrix in terms of (r, g, b):

$$\begin{aligned}
\Sigma &= \sum_{(x,y) \in R_i} T \vec{\Upsilon}_{(x,y)} \cdot \vec{\Upsilon}^\tau_{(x,y)} T^\tau \\
&= T(\mathcal{I} - \vec{\mathbf{C}}\vec{\mathbf{C}}^\tau) \sum_{(x,y) \in R_i} \mathbf{X}_{(x,y)} \cdot \vec{\mathbf{X}}^\tau_{(x,y)} (\mathcal{I} - \vec{\mathbf{C}}\vec{\mathbf{C}}^\tau) T^\tau \\
&= T \sum_{(x,y) \in R_i} \vec{\mathbf{X}} \cdot \vec{\mathbf{X}}^\tau T^\tau
\end{aligned}$$

where T is the 2×3 rotation matrix with $T\vec{\mathbf{C}} = \vec{\mathbf{0}}$ (see equation 34):

$$T = \begin{pmatrix} \cos\beta, -\sin\beta, 0 \\ \sin\beta\cos\alpha, \cos\beta\cos\alpha, -\sin\alpha \end{pmatrix} \tag{50}$$

with $\alpha = \tan^{-1}(b/\sqrt{r^2+g^2})$ and $\beta = \tan^{-1}(g/r)$. Minimizing the determinant of such matrix with respect to (r, g, b) is possible but nontrivial.

To summarize, our least square fitting algorithm minimizes the energy functional under the assumption that the residues $(e_1, e_2)^\tau$ have zero mean and $\sigma_1 = \sigma_2$. This assumption is well satisfied in most homogeneous color regions. Even if these assumptions are not fully satisfied our algorithm remains a good approximation to the real optimal solution, given by equation (49), provided the fluctuations of the illuminance are larger than those of the residues.

References

[1] R.Adams and L.Bischof. "Seeded region growing". *IEEE Trans. on PAMI.* vol.16. No.6. June 1994.

[2] J.R.Beveridge et al. "Segmenting images using localizing histograms and region merging". *Int.J.of Compt.vision.* vol.2. 1989.

[3] A. Blake and A. Zisserman. *Visual Reconstruction.* Cambridge, MA. the MIT press, 1987.

[4] J.F. Canny. "A computational approach to edge detection". *IEEE Trans. Patt. Anal. Mach. Intell.*. PAMI-8(6):679-698. 1986.

[5] A. Chakraborty, L.H. Straib and J.S. Duncan. "Deformable Boundary Finding Influenced by Region Homogeneity". *Proceedings of Computer Vision and Pattern Recognition CVPR94.* Seattle. 1994.

[6] L.D. Cohen and I. Cohen. "A finite element method applied to new active contour models and 3D reconstruction from cross sections". *Proceedings Third International Conference on Computer Vision ICCV90.* Osaka, Japan. 1990.

[7] L.D. Cohen. "On Active Contour Models and Balloons". *CVGIP: Image Understanding.* vol.53, No.2, March, pp 211-218. 1991.

[8] T.J.Darrell and A.P.Pentland, "Cooperative Robust Estimation Using Layers of Support". MIT Media Lab TR-163 (revised version) 1994.

[9] D.Geiger and A.L.Yuille. "A common framework for image segmentation". *Inter.J.of Computer Vision* 6(3), pp 227-243 1991.

[10] S. Geman and D. Geman. "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images". *IEE Trans. PAMI* **6**. pp 721-741. 1984.

[11] D.Geman,S.Geman and P.Dong. "Boundary Detection by Constrained Optimization". *IEEE PAMI* 12(7) pp 609-628 1990.

[12] M.Grayson, "The Heat Equation Shrinking Convex Plane Curves", *Journal of Differential Geomerty.* vol.23 pp285-314 1987.

[13] T.H. Hong and A. Rosenfeld. "Compact Region Extraction Using Weighted Pixel Linking in a Pyramid". *IEEE Trans. Patt. Anal. and Mach. Intell.* PAMI-6, No. 2. March. pp 222-229. 1984.

[14] R. Johnson and D. Wichem. *Applied Multivariate Statsitical Analysis.* Prentice-Hall, Inc. Eaglewood Cliffs. New Jersey. 1982.

[15] T. Kanungo, B.Dom, W.Niblack and D.Steele. "A fast algorithm for MDL-based multi-band image segmentation". *Proc.Comp. Vision and Patt. Recognition* CVPR 1994.

[16] M. Kass, A. Witkin and D. Terzopoulos. "Snakes: Active Contour Models". *Proceedings 1st Inter.Conf.on Computer Vision ICCV87*. London, England. 1987.

[17] G.J. Klinker, S.A. Shafer and T. Kanade. "A Physical Approach to Color Image Understanding". *International Journal of Computer Vision*, 4, pp 7-38. 1990.

[18] G. Koepfler, C. Lopez and J.M. Morel. "A Multiscale Algorithm for Image Segmentation by Variatyional Approach". *SIAM J. NUMER. ANAL.* Vol. 31, No. 1, pp 282-299. February, 1994.

[19] . Y.G. Leclerc. "Constructing Simple Stable Descriptions for Image Partitioning". *International Journal of Computer Vision*, 3, pp 73-102. 1989.

[20] Y.G.Leclerc "Region growing using the MDL principle". in *DARPA image understanding workshop*, 1990.

[21] J.M.Morel and S.Solimini. *Variational Methods in Image Segmentation*. Birkhauser publishing.inc. 1995.

[22] T.S.Lee,D.Mumford and A.L.Yuille. "Texture Segmentation By Minimizing Vector-valued Energy Functionals:the Coupled-membrane model". in G.Sandini (eds.) Lecture Notes in Computer Sciemce, no. 588, ECCV'92 pp 165-173 Springer-Verlag 1992.

[23] A. Leonardis, A. Gupta and R. Bajcsy. "Segmentation as the Search for the Best Description of the Image in Terms of Primitives". Preprint. GRASP Laboratory. Dept. Computer and Information Science. Univ. Pennsylvania. 1990.

[24] A.Madansky, *Presciptions for Working Statisticians* Springer-Verlag New York Inc. pp. 30-50, 1988.

[25] J.Malik and P.Perona. "Preattentive Texture Discrimination With Early Vision Mechanism." *J.Opt.Soc Am.A* 7(5) 923-932, 1990.

[26] D. Mumford and J. Shah. "Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems." *Comm. Pure Appl. Math.,* 42, pp 577-684. 1989.

[27] A.P. Pentland. "Automatic Extraction of Deformable Part Models". *International Journal of Computer Vision*, 4, pp 107-126. 1990.

[28] F. Perez and C. Koch. "Towards Color Image Segmentation in Analog VLSI: Algorithm and Hardware". *International Journal of Computer Vision*, 12:1, pp 17-42. 1994.

[29] G.Sapiro and A.Tannenbaum. "Area and length preserving geometric invariant scale-spaces". *IEEE Trans. on PAMI* 17(1). Jan.1995.

[30] J.Rissanen. "Modelling by shortest data description" *Automatica.* 14:465-471. 1978.

[31] J.Rissanen. *Stochastic Complexity in Statistical Inquiry*, World Scientific, sigapore, 1989.

[32] R.Ronfard. "Region-based strategies for active contour models". *Int.J.of comp.Vision.* vol.13. no.2. 1994.

[33] . T-Y Philips, A. Rosenfeld and A.C. Sher. "O(log n) Bimodality Analysis". *Pattern Recognition* 22, pp 741-746. 1989.

[34] K-K Sung. "A Vector Signal Processing Approach to Color". MIT Artificial Intelligence LAboratory. Technical Report 1349. 1992.

[35] Y. Wang, "Existence and Regularity of Solutions to a Variational Problem of Mumford and Shah", *SIAM J. of Optimization*, 1991

[36] R.Wilson and G.Granlund. "The uncertainty principle in image processing". *IEEE.trans.on PAMI* vol. 6, No.6, Nov.1984.

[37] G. Xu, E. Segawa and S. Tsuji. "Robust Active Contours with Insensitive Parameters". *Pattern Recognition.* Vol. 27, No. 7, pp 879-884. 1994.

[38] S. C. Zhu and A.L. Yuille. "FORMS: a Flexible Object Recognition and Modelling System". Harvard Robotics Laboratory Technical Report 94-1. 1994. To appear in IJCV.