

A Statistical Approach to Curve Evolution for Image Segmentation *

Anthony Yezzi, Jr. Andy Tsai Alan Willsky
Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139

Abstract

In this paper, we present a new class of deformable contour models for the segmentation of images which exhibit a known number of features. The key idea behind our approach is to use geometric curve evolutions which maximally separate a predetermined set of statistics within the image. Some attractive elements of this technique include a natural incorporation of both global and local image information and a robustness to noise which comes from avoiding the sensitive differential operators commonly employed in other snake-like algorithms to detect edges. In addition, our utilization of level set techniques in the implementation of our flows yields a number of numerical flexibilities which greatly simplify the algorithmic overhead associated with tracking multiple, dynamically changing regions.

By separating the image data into a predetermined number of classes with respect to given geometric constraints, we obtain both a segmentation algorithm and a geometric clustering algorithm. In this sense our methodology shares a common paradigm with the Zhu-Yuille region competition method. While the two approaches lead to different flows, an interesting comparison can be made when exactly two regions are used to segment a bimodal image. The approaches diverge, however, in their treatment of more general forms of imagery.

Index Terms: *active contours, boundary detection, clustering, curve evolution, gradient flows, region competition, segmentation, snakes, statistics.*

*This work was supported by ONR grant N00014-91-J-1004, by subcontract GC123919NGD from Boston University under the AFOSR Multidisciplinary Research Program on Reduced Signature Target Recognition, and by ARO grant DAAH04-96-1-0494 through Washington University.

1 Introduction

In recent years, curve evolution methods have been introduced for a variety of purposes including image denoising, enhancement, optical flow, stereo disparity, morphology, and image segmentation. This last topic will be the focus of the present paper. The problem of segmentation has been approached using a number of different techniques including curve evolution methods as well as statistical methods¹. In this work, we propose a new class of variational algorithms which combine curve evolution and statistics in a natural way.

The snake methodology was introduced in the mid-1980's [12, 29]. In the past few years, a number of approaches have been proposed for the problem of *snakes* or *active contours*. The fundamental idea underlying these works is the utilization of deformable contours which conform to various object shapes and motions. Snakes have been used for edge detection, segmentation, shape modelling, and visual tracking (see [2] and the references therein).

This paper formulates a novel statistical approach to snakes that incorporates both *local and global information* in a first principles manner via a special class of energy functionals. The key idea is to assume that an image consists of a finite number of regions, characterizable by a predetermined set of features (e.g. means, variances, textures) which may be inferred or estimated from the image data, to construct energy functionals which favor a maximal separation of these features, and to evolve active contours via the corresponding gradient descent equations. Introducing a penalty on the length of the active contours gives rise to a class of geometrically constrained clustering algorithms in which data elements are grouped both by value and by mutual proximity.

A particularly attractive feature of this approach is that it avoids any computations to explicitly detect edges. The evolving curves are attracted to edges in a much more indirect manner than in most previous snake-like algorithms [4, 5, 8, 12, 13, 18, 28, 29, 30]. Since differential edge-detection operators are, in general, highly sensitive to most common types of image noise, an extra element of robustness is achieved by avoiding them altogether (see also [25, 31, 32]). See [6, 7] for heterogeneous approaches that blend both direct and indirect edge-detection methods.

Finally, we should point out that the approach presented in this paper shares common aspects with the region competition approach of Zhu-Yuille [31, 32], which also regards an image as the composition of a finite number of regions. In their approach, each region is treated as a random field derived from a parameterized distribution. Neighboring regions are allowed to “compete” for pixels along mutual boundaries via the likelihood ratio test. Although the philosophies and the energy functionals behind their approach and ours are quite different, the final gradient descent equations are strikingly similar for the special case of two regions. The similarities and differences between the two approaches will be elucidated in Section 3.

The contents of this paper are summarized as follows. In Section 2 we outline a class of flows for the segmentation of imagery characterized by two different region types. We introduce this section with the instructive example of greyscale imagery in which regions may be distinguished by two different mean intensity values. We continue with additional

¹An extensive discussion of various segmentation methods as well as a large set of references on the subject may be found in the book [19].

flows for more general types of bimodal imagery. In Section 3 we formulate a probabilistic analogue of the introductory flow which turns out to be equivalent to a special case of the Zhu-Yuille region competition algorithm for exactly two regions. In Section 4, we return to the model used in Section 2, but generalize it to incorporate more than two region types. In the numerical implementation of our flows, we employ level set techniques; see Osher and Sethian [24, 26] and the references therein. The level set implementations are discussed in Section 5 followed by simulations in Section 6.

2 Binary Flows

In this section, we present gradient flows designed to segment bimodal images via an evolving curve. The flow presented in the first part of this section for simple binary intensity images is offered for the purpose of illustration (since such images are already segmented) and to develop an intuition for more general flows designed for less trivial forms of bimodal imagery.

2.1 Flows for binary images

We begin with the assumption that the domain of an image $I(x, y)$ consists of a foreground region R of intensity I^r and a complementary background region R^c of intensity $I^c \neq I^r$. We wish to determine an evolution that will continuously attract any initial closed curve \vec{C} toward the boundary ∂R of R .

Since an arbitrary closed curve over the domain of I will enclose some portion of R and some portion of R^c , the average intensities u and v inside and outside the curve respectively are bounded above and below by I^r and I^c . Consequently, using the distance between u and v to measure how well \vec{C} has separated the foreground from the background will ensure an upper-bound of $|I^r - I^c|$ that is uniquely attained when $\vec{C} = \partial R$. A related strategy, which also assumes no previous knowledge of I^r or I^c , would be to descend along the following quadratic energy functional:

$$E = -\frac{1}{2}(u - v)^2. \quad (1)$$

Letting $S_u = \int_{R^u} I dA$ and $A_u = \int_{R^u} dA$, where R^u denotes the interior of \vec{C} , and expressing their first variations² as $\nabla S_u = I\vec{N}$ and $\nabla A_u = \vec{N}$ (see Appendix A.1), where \vec{N} denotes the outward unit normal of \vec{C} , allows us to compute the first variation of $u = S_u/A_u$ as follows:

$$\nabla u = \frac{A_u \nabla S_u - S_u \nabla A_u}{A_u^2} = \frac{A_u I - S_u}{A_u^2} \vec{N} = \frac{I - u}{A_u} \vec{N}.$$

By a similar computation, the first variation of v is found to be

$$\nabla v = -\frac{I - v}{A_v} \vec{N}$$

²Technically ∇S_u and ∇A_u denote the gradient directions over the space of smooth curves which come from maximizing the first variations of the integrals S_u and A_u respectively. However, to avoid cumbersome language, especially in later sections, we simply refer to these quantities as first variations.

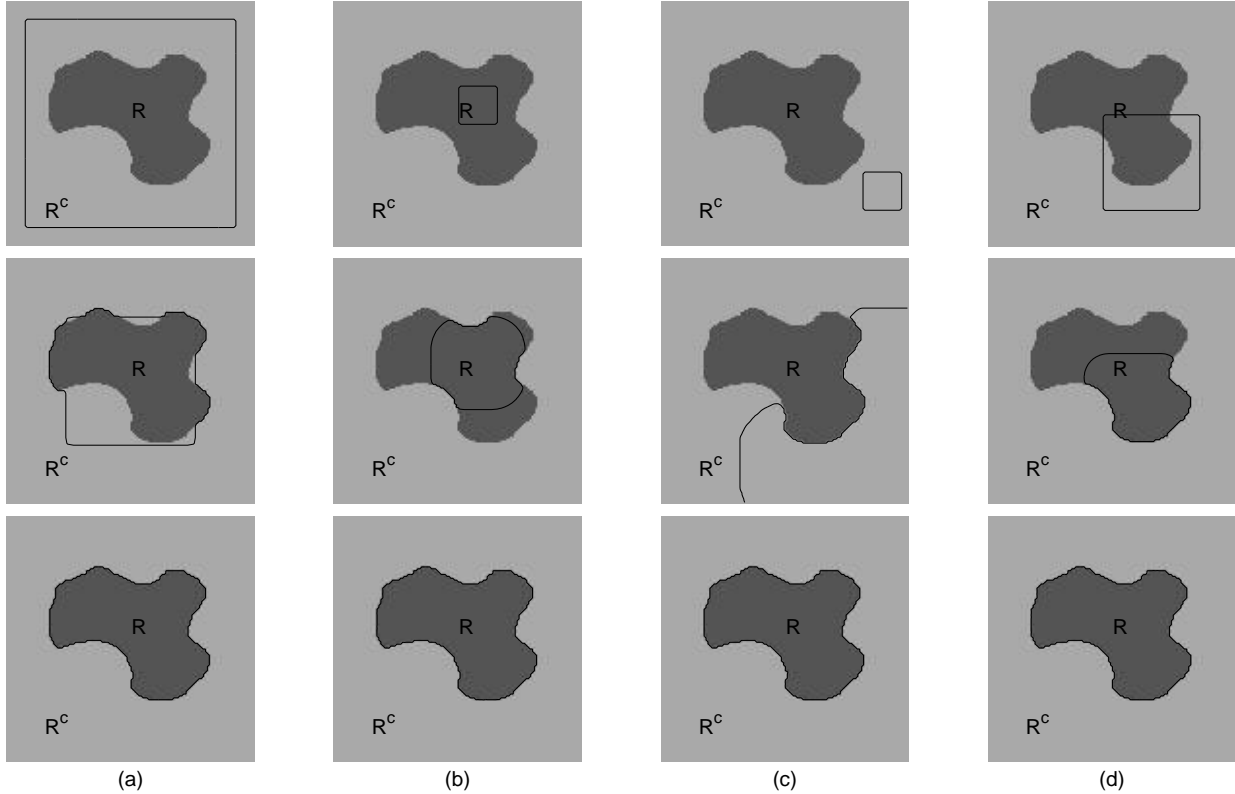


Figure 1: Flow (2) automatically proceeds in the correct direction. (a) inward flow from outside (b) outward flow from inside (c) outward flow from outside (d) bi-directional flow

where A_v denotes the area of R^v (the exterior of \vec{C}). Notice the minus sign in front due to the fact that \vec{N} is the outward normal with respect to R^u , and so the outward normal with respect to its complement R^v is in turn $-\vec{N}$. We now make use of these expressions to compute the gradient descent curve evolution for E :

$$\frac{d\vec{C}}{dt} = -\nabla E = (u - v) \left(\frac{I - u}{A_u} + \frac{I - v}{A_v} \right) \vec{N}. \quad (2)$$

This flow essentially pulls apart the mean intensities inside the curve and outside the curve as it evolves.

In the case that R consists of a single, simply connected region, E possesses just one minimum and a number of local maxima. These local maxima all exhibit the same value of $E = 0$ and only occur when an initial contour bisects both the foreground and the background so that $u = v = (I^r + I^c)/2$. Note that any other initial contour will converge under this gradient flow toward the boundary of R , attaining the global minimum of $E = -\frac{1}{2}(I^r - I^c)^2$. Problematic initial contours are, therefore, easily detected by checking whether $u = v$ and may be “corrected” via any perturbation which changes u or v .

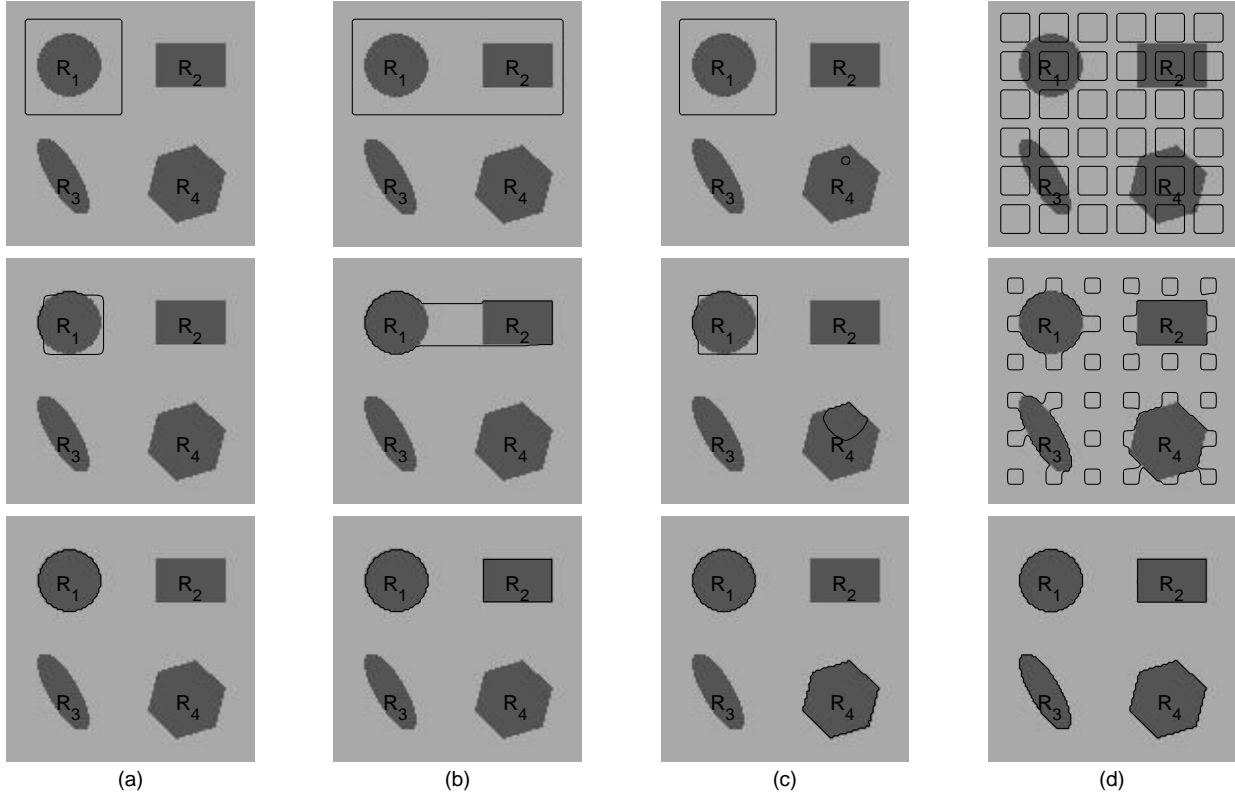


Figure 2: (a)-(c) *Semiautomatic segmentation*: Placing initial contours near desired boundaries influences the flow toward local minima. (d) *Automatic segmentation*: Seeding the entire image with multiple contours influences the flow toward the global minimum.

2.2 Some remarks on binary flows

At this point, the highly restricted class of imagery (binary with a single, simply connected foreground region) for which the present flow has been designed makes it appear to be little more than a nice mathematical result with very few useful applications. In the following sections, we will modify this flow and the corresponding energy functional to handle more general classes of imagery. However, this simple flow already exhibits some very attractive features which will be inherited by its more general forms.

First and foremost, the flow automatically proceeds in the correct direction without relying upon an additional inflationary term, common to many snake algorithms. An initial contour encompassing the region R will flow inward toward the boundary (Fig. 1a); a contour contained inside R will flow outward toward the boundary (Fig. 1b); a contour contained outside R will flow outward and wrap around the boundary (Fig. 1c); and finally, a contour partially inside and partially outside R , will flow in both directions toward the boundary (Fig. 1d). No initial inflationary step or prior knowledge of the evolution direction is ever needed. The initial contour may, therefore, be placed completely arbitrarily.

Before going on to discuss more general “binary” flows, let’s consider what happens when R consists of multiple subregions R_1, \dots, R_n . In this case the energy functional E contains

a number of local minima. The global minimum is still attained by the overall boundary ∂R . However, the boundary, ∂R_i , of each connected subregion, R_i , gives rise to a local minimum as does any union, $\partial R_{i_1} \cup \partial R_{i_2} \cup \dots \cup \partial R_{i_m}$, of such subregion boundaries. This phenomenon is desirable if one wishes to control which objects are captured by the flow. Through strategic placement of initial contours, the user can influence the flow toward the local minimum associated with the boundaries of just the desired objects (Fig. 2a-c). This yields the flexibility to devise semiautomatic segmentation algorithms.

A completely automatic method can still be employed to detect the entire set of subregion boundaries in the case of a multiply connected foreground (the global minimum of E) if the minimum subregion size can be anticipated. To do so, one simply seeds the entire image with small contours with diameters less than the smallest anticipated subregion diameter and separated by distances less than that same diameter. This will ensure that each subregion contains at least one initial contour. Once the evolution begins, those contours inside the subregions will grow and those outside will disappear (Fig. 2d) or vice-versa, depending upon the ratio of foreground to background captured by the initial contours.

The merging and splitting of contours shown in Fig. 2 illustrates the significant advantage of selecting a numerical implementation method which allows topological changes. Level set methods [24, 26] offer a very natural framework for handling such effects. We will discuss these methods and other implementation issues in Section 5.

2.3 Flows for binary images corrupted by noise

In this section, we modify the flow (2) and its associated energy functional (1) to handle a larger, more relaxed class of bimodal imagery. A number of interesting technologies produce images which are well approximated by binary images even though the images themselves are not binary. An important class of ultrasound imagery, for example, consists of light and dark regions clustered around two different average intensities plus varying levels of impulsive-noise. A similar class of bimodal imagery arises in many applications of synthetic aperture radar technology. MRI images which distinguish the white and grey-matter of the brain also fall into this category.

Our previous model may be easily modified to handle this more useful class of imagery when the contaminating noise is zero-mean. Clearly, since our cost functional is based on *average* intensity values inside and outside the evolving contour, zero-mean noise away from the contour will not have a significant effect on its evolution. This is not the case for noise in the vicinity of the contour. Its influence depends strongly upon its distribution. Additive Gaussian noise with small variance compared to the contrast in the underlying binary image, for example, may change the trajectory of the evolving contour, but will have little effect on the final, steady state contour (Fig. 3b). Gaussian noise with higher variance (Fig. 3c) or uniform noise (Fig. 3d) is more troublesome. In the presence of more severely distributed noise, the contour may end up weaving around or encircling extremely small regions, clustering isolated pixels outside the region of interest with the region itself and/or omitting isolated pixels within the region of interest in order to gain tiny decreases in the cost-functional. To counter these effects, we follow the philosophy of Mumford and Shah [21, 22] by incorporating a geometric constraint on the evolving contour via an additional

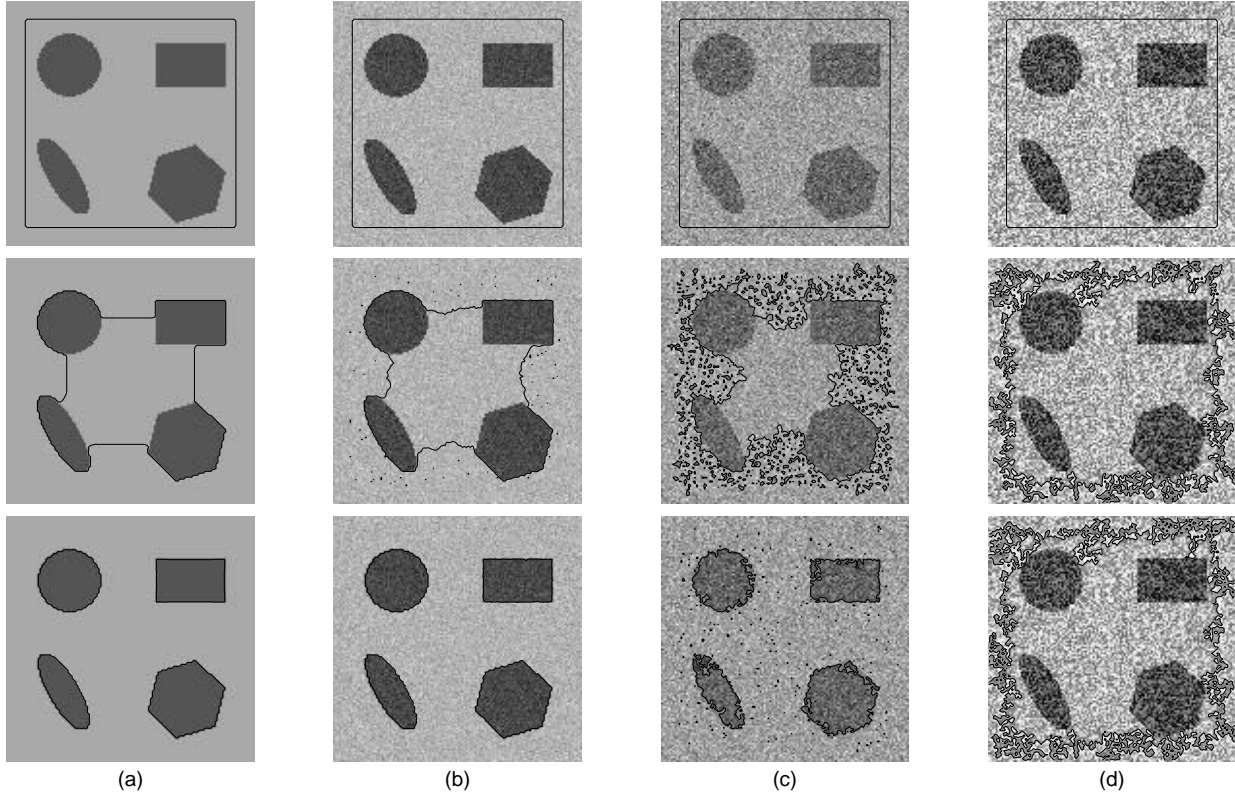


Figure 3: The effects of additive noise depend upon its distribution. (a) noise-free image (b) Gaussian noise: SNR=1dB (c) Gaussian noise: SNR=-0.1dB (d) uniform noise

term in the energy functional (1) which penalizes its arclength. Doing so yields the following new energy

$$E = -\frac{1}{2}(u - v)^2 + \alpha \int_{\vec{C}} ds, \quad (3)$$

where $\alpha \geq 0$ and s represents the arclength parameter of \vec{C} . Noting that the gradient direction for length is given by $\kappa \vec{N}$, where κ denotes the signed curvature of \vec{C} , the corresponding gradient descent on E is given by

$$\frac{d\vec{C}}{dt} = (u - v) \left(\frac{I - u}{A_u} + \frac{I - v}{A_v} \right) \vec{N} - \alpha \kappa \vec{N}. \quad (4)$$

The influence of the second term in this flow is most strongly felt along points of the evolving contour where the magnitude of the curvature is very large. This clearly helps prevent the contour from wrapping around tiny pieces of noise, with the tradeoff that sharp corners, if they exist, in the underlying binary image may be rounded off by the final contour (see Fig. 4).

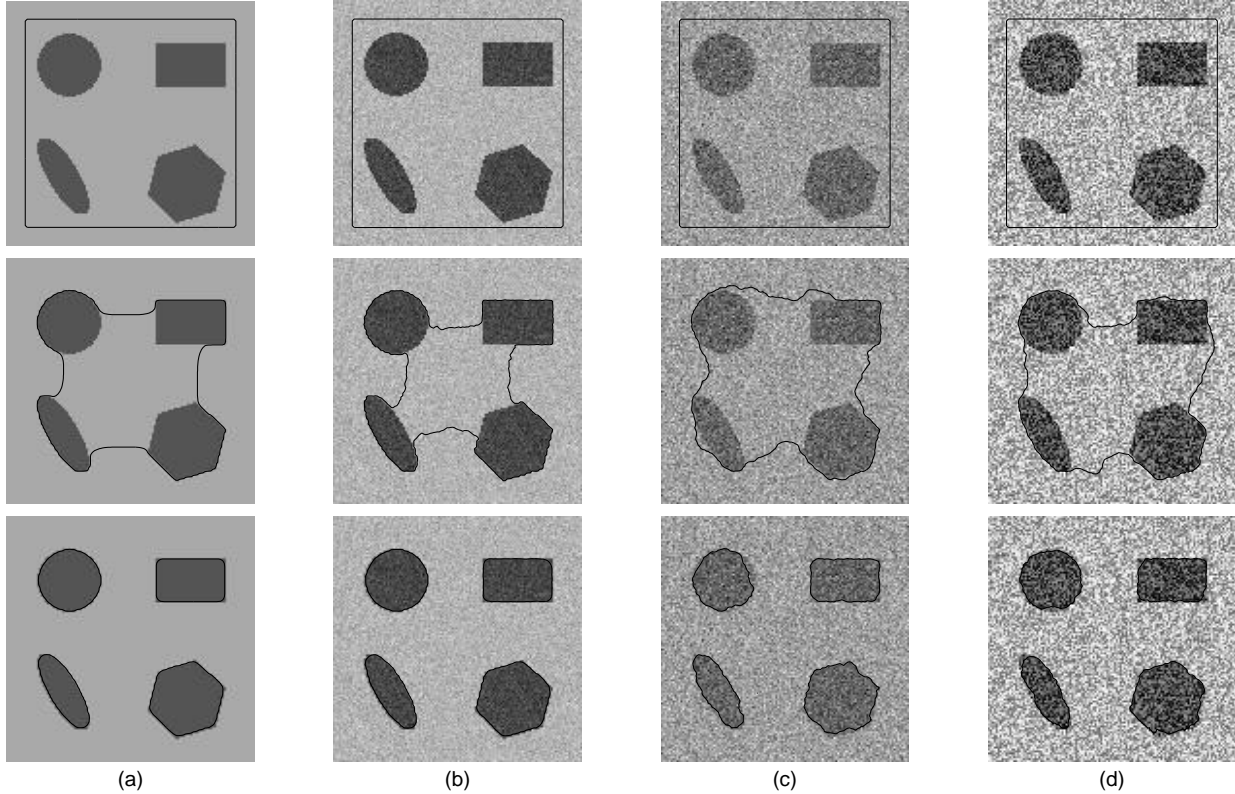


Figure 4: A penalty on length counters the fractalizing effect of additive noise. (a) noise-free image (b) Gaussian noise: SNR=1dB (c) Gaussian noise: SNR=-0.1dB (d) uniform noise

2.4 More general binary flows

Up to now we have used the term binary to suggest two separate scalar intensities (greylevels). We may readily generalize our results to vector-valued bimodal imagery (e.g. color images composed of regions clustered around two different colors) by employing the following more general energy functional:

$$E = -\frac{1}{2}\|u - v\|^2 + \alpha \int_{\vec{C}} ds, \quad (5)$$

where u and v are now vector-valued averages of the image inside and outside the curve respectively. In the case of an n -vector valued image, $u = (u_1, \dots, u_n)$, $v = (v_1, \dots, v_n)$, $I(x, y) = (I_1(x, y), \dots, I_n(x, y))$ and the gradient descent becomes

$$\frac{d\vec{C}}{dt} = \sum_{i=1}^n (u_i - v_i) \left(\frac{I_i - u_i}{A_u} + \frac{I_i - v_i}{A_v} \right) \vec{N} - \alpha \kappa \vec{N}. \quad (6)$$

Note that the vector components I_1, \dots, I_n in this formulation do not necessarily have to represent image intensity values (as in a color image). They may represent wavelet coefficients from a greyscale image or other forms of multi-spectral measurements. With this observation, one could easily segment an image consisting of two different textures using (6) so long as a distinguishing “texture vector” can be derived. The wood grain textures in

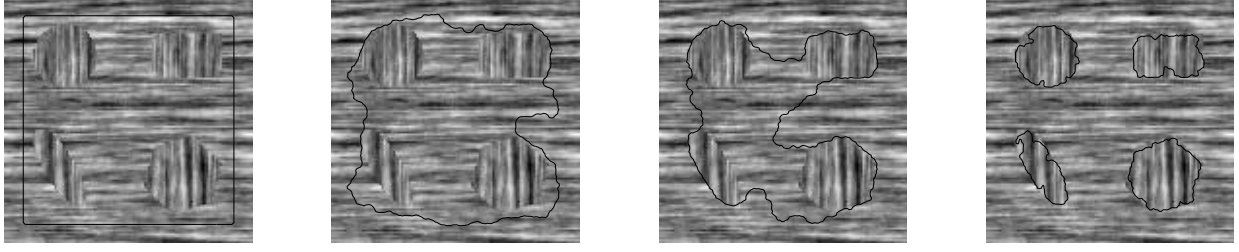


Figure 5: Using flow (6) in conjunction with the vector-valued orientation map (7) to capture two different wood grain textures.

Fig. 5, for example, comprise a foreground of vertical grains over a background of horizontal grains. Thus, even though the image is not binary in intensity, it is essentially binary in terms of local orientation. To capture the boundaries of these regions using (6) one merely transforms the initial intensity image $\hat{I}(x, y)$, into a vector-valued orientation image $I(x, y)$ either by applying a series of orientation filters or through some other sort of map which assigns a unique vector to each orientation. The simulation shown in Fig. 5 employed the following map

$$I = \frac{(\hat{I}_x^2 - \hat{I}_y^2, 2\hat{I}_x\hat{I}_y)}{\hat{I}_x^2 + \hat{I}_y^2}, \quad (7)$$

obtained by doubling the angle of the polar representation of the normalized gradient of \hat{I} (so that gradient vectors pointing in the same or opposite directions are mapped to the same point) and then expressing the result in cartesian coordinates.

Finally, the binary approach may be further generalized by considering different statistical parameters (other than means). The basic idea is to formulate a set of statistics which distinguish the foreground and background regions from each other and then derive curve evolutions to “pull them apart”. Suppose, for example, that an image consists of two regions with identical means but different variances. In this case, a curve may be attracted toward the boundary by descending along the following energy functional

$$E = -\frac{1}{2}(\sigma_u^2 - \sigma_v^2)^2 + \alpha \int_{\vec{C}} ds, \quad (8)$$

where $\sigma_u^2 = \frac{1}{A_u} \int_{R^u} (I - u)^2 dA$ denotes the sample variance inside the curve \vec{C} and $\sigma_v^2 = \frac{1}{A_v} \int_{R^v} (I - v)^2 dA$ denotes the sample variance outside the curve. By the result in Appendix A.1, the first variations of these parameters may be written as

$$\nabla \sigma_u^2 = \frac{(I - u)^2 - \sigma_u^2}{A_u} \vec{N} \quad \text{and} \quad \nabla \sigma_v^2 = -\frac{(I - v)^2 - \sigma_v^2}{A_v} \vec{N};$$

yielding the following gradient flow, demonstrated in Fig. 6.

$$\frac{d\vec{C}}{dt} = \left\{ (\sigma_u^2 - \sigma_v^2) \left(\frac{(I - u)^2 - \sigma_u^2}{A_u} + \frac{(I - v)^2 - \sigma_v^2}{A_v} \right) - \alpha \kappa \right\} \vec{N} \quad (9)$$

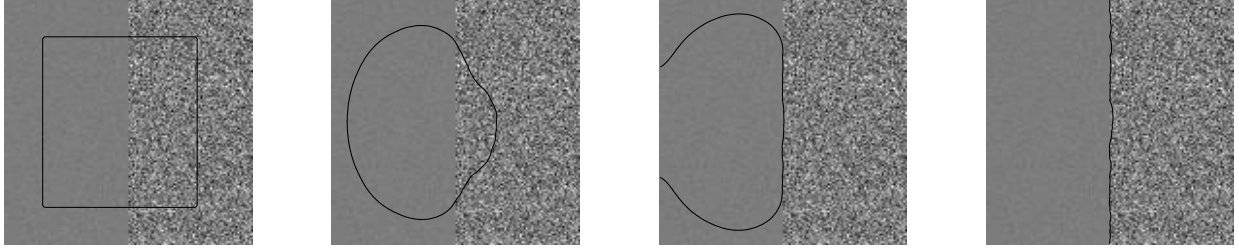


Figure 6: Using flow (9) to capture two regions of identical means but different variances. The left half of the image contains Gaussian noise with variance 1, while the right half contains Gaussian noise with variance 100.

3 Probabilistic Analogues (Region Competition)

In the previous section, we presented a number of different curve evolutions designed to partition an image into two regions (motivating the term “binary flow”), across which, the variation of some computed statistic was maximized. In particular, the first flow (2) was constructed to separate the means of the image inside and outside the evolving curve. Note that the only stipulation made about the distribution of additive noise, if present, was that it be zero-mean.

In the first part of this section, we reformulate the binary segmentation problem from a more formal probabilistic perspective. This will lead to a different, maximum likelihood-based cost functional for which we derive the corresponding gradient flow. The resulting curve evolution will be quite similar to (2). It turns out that this new flow, when combined with a curvature term to penalize arclength, arises in a special case of the region competition algorithm presented in a very elegant paper by Zhu and Yuille [32]. This is no surprise, since their segmentation algorithm is driven by a minimum description length energy functional for which maximum likelihood estimates play a critical minimizing role. The similarity between these flows suggests a strong tie between the binary flows presented in Section 2 and the region competition algorithm presented below. We will therefore compare and contrast these approaches in the second part of this section.

3.1 Maximum-likelihood flows

Let us define a bimodal image $I(x, y)$ as an outcome of a random function \mathbf{I} which assigns a random variable $\mathbf{I}(x, y)$ from one distribution P^r to each point within a region R and from a second distribution P^c to each point within the complementary region R^c . Assume further that these random variables are independent from point to point³. Our objective then is to process the observed image values $I(x, y)$ in order to estimate the partition R and R^c . A natural strategy, given prior knowledge of P^r and P^c , would be to choose the estimates \hat{R} and \hat{R}^c to maximize the likelihood of I assuming P^r in \hat{R} and P^c in \hat{R}^c . More generally, if P^r and P^c are only known up to a finite number of parameters, $\alpha_1^r, \dots, \alpha_m^r$ and $\alpha_1^c, \dots, \alpha_n^c$,

³Discrete and continuous spatial domains incur different notions of independence which will be clarified as we develop both cases.

the related strategy would be to estimate not only the region R , but the parameter values as well to maximize the likelihood of I given $P^r(\hat{\alpha}_1^r, \dots, \hat{\alpha}_m^r)$ in \hat{R} and $P^c(\hat{\alpha}_1^c, \dots, \hat{\alpha}_n^c)$ in \hat{R}^c , where $\hat{\cdot}$ denotes an estimate for the corresponding unknown variable.

Let us be more specific now and consider the case of a bimodal image $I(x, y)$ where the corresponding random variables $\mathbf{I}(x, y)$ are drawn from two Gaussian distributions, both with known variance σ^2 but with unknown means μ^r and μ^c . For simplicity of exposition in what follows, let us assume a discrete spatial domain (i.e. the spatial points (x, y) correspond to discrete pixel locations, and R and R^c are sets of these discrete points). Then, given any candidate partition \hat{R} and \hat{R}^c , and candidate means $\hat{\mu}^r$ and $\hat{\mu}^c$, the likelihood of I , assuming $\mathbf{I}(x, y) \sim N(\hat{\mu}^r, \sigma^2)$ for $(x, y) \in \hat{R}$ and $\mathbf{I}(x, y) \sim N(\hat{\mu}^c, \sigma^2)$ for $(x, y) \in \hat{R}^c$ is given by

$$\left(\prod_{(x,y) \in \hat{R}} \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{(I(x,y) - \hat{\mu}^r)^2}{2\sigma^2} \right) \left(\prod_{(x,y) \in \hat{R}^c} \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{(I(x,y) - \hat{\mu}^c)^2}{2\sigma^2} \right).$$

Choosing \hat{R} , $\hat{\mu}^r$, and $\hat{\mu}^c$ to maximize this, or equivalently the log-likelihood of I ,

$$\sum_{(x,y) \in \hat{R}} \left(\frac{-(I(x,y) - \hat{\mu}^r)^2}{2\sigma^2} - \log \sqrt{2\pi}\sigma \right) + \sum_{(x,y) \in \hat{R}^c} \left(\frac{-(I(x,y) - \hat{\mu}^c)^2}{2\sigma^2} - \log \sqrt{2\pi}\sigma \right),$$

is also equivalent to minimizing

$$\begin{aligned} & \sum_{(x,y) \in \hat{R}} (I(x,y) - \hat{\mu}^r)^2 + \sum_{(x,y) \in \hat{R}^c} (I(x,y) - \hat{\mu}^c)^2 = \\ & \sum_{(x,y) \in \hat{R}} (I(x,y)^2 - 2\hat{\mu}^r I(x,y) + (\hat{\mu}^r)^2) + \sum_{(x,y) \in \hat{R}^c} (I(x,y)^2 - 2\hat{\mu}^c I(x,y) + (\hat{\mu}^c)^2). \end{aligned}$$

This, in turn, is equivalent to minimizing the following energy functional:

$$\hat{E}(\hat{R}, \hat{\mu}^r, \hat{\mu}^c) = \sum_{(x,y) \in \hat{R}} ((\hat{\mu}^r)^2 - 2\hat{\mu}^r I(x,y)) + \sum_{(x,y) \in \hat{R}^c} ((\hat{\mu}^c)^2 - 2\hat{\mu}^c I(x,y)).$$

Notice that the partial derivatives

$$\frac{\partial \hat{E}}{\partial \hat{\mu}^r} = 2 \sum_{\hat{R}} (\hat{\mu}^r - I(x,y)) \quad \text{and} \quad \frac{\partial \hat{E}}{\partial \hat{\mu}^c} = 2 \sum_{\hat{R}^c} (\hat{\mu}^c - I(x,y))$$

vanish if and only if $\hat{\mu}^r$ and $\hat{\mu}^c$ are chosen to be the sample means, u and v , of I over \hat{R} and \hat{R}^c respectively. Therefore, a local minimum of \hat{E} must be a local minimum of

$$E(\hat{R}) = \hat{E}(\hat{R}, u, v) = \sum_{\hat{R}} (u^2 - 2uI) + \sum_{\hat{R}^c} (v^2 - 2vI) = -\left(\sum_{\hat{R}} u^2 + \sum_{\hat{R}^c} v^2 \right).$$

The continuous spatial domain version of the preceding development consists of modeling $\mathbf{I}(x, y)$ as the sum of a stationary white noise process plus one of two mean values, μ^r inside

the region R and μ^c outside the region R . In this case, the energy functional to be minimized in order to obtain the maximum likelihood estimates of R , μ^r , and μ^c is given by

$$\hat{E}(\hat{R}, \hat{\mu}^r, \hat{\mu}^c) = \int_{\hat{R}} (I(x, y) - \hat{\mu}^r)^2 dx dy + \int_{\hat{R}^c} (I(x, y) - \hat{\mu}^c)^2 dx dy.$$

The minimization of this with respect to $\hat{\mu}^r$ and $\hat{\mu}^c$ follows in exactly the same manner. That is, if we let u and v again denote the samples means, i.e. $u = S_u/A_u$ and $v = S_v/A_v$ (where $S_u = \int_{\hat{R}} I dA$ and $A_u = \int_{\hat{R}} dA$, and where $S_v = \int_{\hat{R}^c} I dA$ and $A_v = \int_{\hat{R}^c} dA$), then a local minimum of \hat{E} must also be a local minimum of

$$E(\hat{R}) = -(A_u u^2 + A_v v^2), \quad (10)$$

whose first variation may be written as

$$-\nabla E = \nabla A_u u^2 + \nabla A_v v^2 + 2A_u u \nabla u + 2A_v v \nabla v.$$

Letting \vec{N} denote the outward unit normal of the mutual boundary \vec{C} of \hat{R} and \hat{R}^c with respect to \hat{R} (and consequently letting $-\vec{N}$ denote the outward unit normal with respect to \hat{R}^c) and substituting the expressions for ∇A_u , ∇A_v , ∇u , and ∇v given in Section 2.1 to obtain

$$\begin{aligned} -\nabla E &= \left\{ u^2 - v^2 + 2A_u u \frac{I - u}{A_u} - 2A_v v \frac{I - v}{A_v} \right\} \vec{N} \\ &= \{ -(u^2 - v^2) + 2I(u - v) \} \vec{N} \\ &= (u - v)(I - u + I - v) \vec{N} \end{aligned}$$

yields the following gradient flow for (10):

$$\frac{d\vec{C}}{dt} = (u - v)(I - u + I - v) \vec{N}. \quad (11)$$

Notice the similarity between this *probabilistic* binary flow and the *statistical* binary flow, given by (2), which was obtained via *deterministic* considerations. The only difference is that (11) does not directly depend upon the areas inside and outside the evolving curve.

3.2 Region competition

In this section, we will elucidate the region competition approach of Zhu and Yuille [32] and compare it to the approach we have just described.

First of all, the technique in [32] is based on the concept of “minimum description length.” Such an idea has appeared before, for example, in the work of Leclerc [15]. The idea is to consider the segmentation problem as a partitioning problem, where the criterion for choosing one partition instead of another is the *description length*. As is usual, the measure of the description length must be accomplished according to some *a priori* language. Thus in fact, the Minimum Description Length (MDL) is essentially equivalent to the Maximum A Posteriori (MAP) estimate from the Bayesian paradigm. It may be regarded as an information interpretation of this classical method [1, 20].

Zhu and Yuille, building on the work of Leclerc propose an MDL-based energy functional which is the continuum limit of Leclerc’s functional. Taking the gradient direction, they obtain a system similar to ours with a smoothing term based on Euclidean curve shortening, and a term which determines the motion of a point on the common boundary of two regions via the likelihood ratio test.

The fact that one ends up with similar equations is of course straightforward. The bottom line is that all of these methods are derived from finding the “best” solution of all ill-posed linear equation of the form

$$Hf = g.$$

This may be in the continuous or discrete domain (e.g. continuously H may represent an integral operator). Maximum likelihood then amounts to finding a solution which maximizes the global likelihood (or log-likelihood) $p(g|f)$. The MAP method will try to maximize the posterior $p(f|g)$. The MDL method is an information theoretic variant of this. Under standard assumptions of an independent Gaussian or Poisson noise process, they all lead to very similar solutions. In fact, flow (11) is precisely the region competition flow, modulo a curvature term, for the special case of two regions described by a Gaussian prior model with unit variance.

Thus, for the case of bimodal imagery, our method is related to the Zhu-Yuille region competition approach through the similar structures of the flows (2) and (11) even though our starting point is quite different. The contrast between the two philosophies will become more apparent in the treatment of more complicated imagery. It is important to note that our initial binary flows were formulated deterministically, without the need for prior probabilistic models. In the following section, we will generalize this deterministic interpretation and will obtain flows which are very different from those proposed in [32].

4 Generalization

In this section, we generalize the methodology of Section 2 to develop flows for segmenting trimodal or more general forms of multimodal imagery. The *binary flows* (2), (4), (6), and (9) of Section 2 partition an image domain into exactly two regions. These regions may be multiply connected, consisting of many individual *subregions*; however, the evolving contour distinguishes just two *classes* of regions at any given time.

In the first part of this section, we present a framework for handling *ternary flows*, which partition an image domain into three different region classes. Later, the approach is generalized for an arbitrary number of classes.

4.1 Ternary flows

We begin our discussion of ternary flows by assuming (for now) that the domain of an image $I(x, y)$ consists of two disjoint, simply connected, foreground regions R^a and R^b and a background region R^c (the complement of $R^a \cup R^b$) with mutually distinct intensities I^a , I^b , and I^c , respectively. A closed curve \vec{C}_u in the domain of I will generally enclose some portion of each region; thus, the average intensity u inside \vec{C}_u can be written as a convex

combination of I^a, I^b, I^c (i.e. $u = \alpha I^a + \beta I^b + \gamma I^c$ where $0 \leq \alpha, \beta, \gamma \leq 1$ and $\alpha + \beta + \gamma = 1$). Unfortunately, if I takes its values in \mathbf{R} , there is no unique convex combination since any three points in \mathbf{R} are obviously collinear. This poses a problem since the algorithm we are about to present relies upon geometrically independent⁴ statistics to distinguish the regions R^a, R^b , and R^c .

To be geometrically independent I^a, I^b , and I^c must belong to \mathbf{R}^2 or a higher dimensional space. Accordingly, assume that I is a vector-valued image with vectors in \mathbf{R}^2 and that $I^a = (I_1^a, I_2^a)$, $I^b = (I_1^b, I_2^b)$, and $I^c = (I_1^c, I_2^c)$ are geometrically independent. We may now represent $u = (u_1, u_2)$ as a unique convex combination of these three values. The same situation applies to the average intensity v within the interior of a second curve \vec{C}_v and to the average intensity w within the mutual exterior of \vec{C}_u and \vec{C}_v . Our segmentation goal is to construct coupled flows that will continuously attract \vec{C}_u toward one of the boundaries ∂R^a or ∂R^b (of R^a and R^b respectively) while simultaneously attracting \vec{C}_v toward the other.

By virtue of their geometric independence, I^a, I^b , and I^c form the vertices of a triangle T_{abc} . As convex combinations of these three values, u, v , and w lie within this triangle, forming another triangle T_{uvw} completely contained in T_{abc} . (This is true even if the interiors of \vec{C}_u and \vec{C}_v overlap, providing a flexibility to our approach that is not provided by region competition in which evolving regions must be disjoint.) As such, the area of the triangle T_{uvw} will always be less than or equal to the area of the triangle T_{abc} , with equality holding if and only if $\vec{C}_u = \partial R^a$ and $\vec{C}_v = \partial R^b$ or vice-versa. We may therefore attract \vec{C}_u and \vec{C}_v toward the desired boundaries without any prior knowledge of I^a, I^b , or I^c by trying to maximize the area of T_{uvw} using the following tri-quadratic energy functional:

$$E = -\frac{1}{2} \det^2(u - w, v - w) = -2 \text{ area}^2(T_{uvw}). \quad (12)$$

If u, v , and w are geometrically independent, then $u - w$ and $v - w$ are linearly independent and therefore yield a nonzero determinant. A more symmetric representation, in terms of vector components, is given by

$$E = -\frac{1}{2} (u_1 v_2 - u_1 w_2 + v_1 w_2 - v_1 u_2 + w_1 u_2 - w_1 v_2)^2. \quad (13)$$

We obtain coupled gradient flows by computing the partial variations $\nabla_{\vec{C}_u} E$ and $\nabla_{\vec{C}_v} E$ (with respect to \vec{C}_u and \vec{C}_v). Noting that $\nabla_{\vec{C}_u} v_1 = \nabla_{\vec{C}_u} v_2 = 0$ (since v does not depend upon \vec{C}_u) and that $\nabla_{\vec{C}_v} u_1 = \nabla_{\vec{C}_v} u_2 = 0$ (since u does not depend upon \vec{C}_v) yields

$$\begin{aligned} -\nabla_{\vec{C}_u} E &= (u_1 v_2 - u_1 w_2 + v_1 w_2 - v_1 u_2 + w_1 u_2 - w_1 v_2) \times \\ &\quad \left\{ (v_2 - w_2) \nabla_{\vec{C}_u} u_1 - (v_1 - w_1) \nabla_{\vec{C}_u} u_2 + (u_2 - v_2) \nabla_{\vec{C}_u} w_1 - (u_1 - v_1) \nabla_{\vec{C}_u} w_2 \right\} \\ -\nabla_{\vec{C}_v} E &= (u_1 v_2 - u_1 w_2 + v_1 w_2 - v_1 u_2 + w_1 u_2 - w_1 v_2) \times \\ &\quad \left\{ (w_2 - u_2) \nabla_{\vec{C}_v} v_1 - (w_1 - u_1) \nabla_{\vec{C}_v} v_2 + (u_2 - v_2) \nabla_{\vec{C}_v} w_1 - (u_1 - v_1) \nabla_{\vec{C}_v} w_2 \right\}. \end{aligned}$$

⁴Noncollinear in this context.

The partial variations of u and v may be derived along the lines presented in Section 2.1:

$$\begin{aligned}\nabla_{\vec{C}_u} u_1 &= \frac{I_1 - u_1}{A_u} \vec{N}_u, & \nabla_{\vec{C}_u} u_2 &= \frac{I_2 - u_2}{A_u} \vec{N}_u, \\ \nabla_{\vec{C}_v} v_1 &= \frac{I_1 - v_1}{A_v} \vec{N}_v, & \nabla_{\vec{C}_v} v_2 &= \frac{I_2 - v_2}{A_v} \vec{N}_v\end{aligned}$$

where \vec{N}_u and \vec{N}_v denote the outward unit normals of \vec{C}_u and \vec{C}_v , respectively, while A_u and A_v denote their areas. The partial variations of w must be derived more carefully, since the interiors of \vec{C}_u and \vec{C}_v may overlap. In this case, the gradient directions are no longer given by smooth, or even continuous, variations of \vec{C}_u and \vec{C}_v . The following limits are obtained by the procedure in Appendix A.2:

$$\begin{aligned}\nabla_{\vec{C}_u} w_1 &= -\frac{I_1 - w_1}{A_w} (1 - \chi_v) \vec{N}_u, & \nabla_{\vec{C}_u} w_2 &= -\frac{I_2 - w_2}{A_w} (1 - \chi_v) \vec{N}_u, \\ \nabla_{\vec{C}_v} w_1 &= -\frac{I_1 - w_1}{A_w} (1 - \chi_u) \vec{N}_v, & \nabla_{\vec{C}_v} w_2 &= -\frac{I_2 - w_2}{A_w} (1 - \chi_u) \vec{N}_v,\end{aligned}$$

where χ_u and χ_v denote the characteristic functions over R^u and R^v (the interiors of \vec{C}_u and \vec{C}_v , respectively). Substituting these into the previous gradient expressions for E yields the following pair of coupled gradient flows for \vec{C}_u and \vec{C}_v .

$$\begin{aligned}\frac{d\vec{C}_u}{dt} &= (u_1 v_2 - u_1 w_2 + v_1 w_2 - v_1 u_2 + w_1 u_2 - w_1 v_2) \times \\ &\left\{ (v_2 - w_2) \frac{I_1 - u_1}{A_u} - (v_1 - w_1) \frac{I_2 - u_2}{A_u} - \right. \\ &\left. (u_2 - v_2) \frac{I_1 - w_1}{A_w} (1 - \chi_v) + (u_1 - v_1) \frac{I_2 - w_2}{A_w} (1 - \chi_v) \right\} \vec{N}_u\end{aligned}\tag{14}$$

$$\begin{aligned}\frac{d\vec{C}_v}{dt} &= (u_1 v_2 - u_1 w_2 + v_1 w_2 - v_1 u_2 + w_1 u_2 - w_1 v_2) \times \\ &\left\{ (w_2 - u_2) \frac{I_1 - v_1}{A_v} - (w_1 - u_1) \frac{I_2 - v_2}{A_v} - \right. \\ &\left. (u_2 - v_2) \frac{I_1 - w_1}{A_w} (1 - \chi_u) + (u_1 - v_1) \frac{I_2 - w_2}{A_w} (1 - \chi_u) \right\} \vec{N}_v\end{aligned}\tag{15}$$

4.2 Some remarks on ternary flows

In this section we discuss the ramifications of our ternary model. In particular, we show how it differs from region competition.

When R^u and R^v are disjoint, the evolution of each curve is not directly tied to the other curve. The coupling, arising from the common set of parameters u , v , and w , is indirect. The characteristic functions χ_u and χ_v yield a more direct coupling when the curves overlap. Nevertheless, in both cases, each curve evolves as a separate entity, enabling

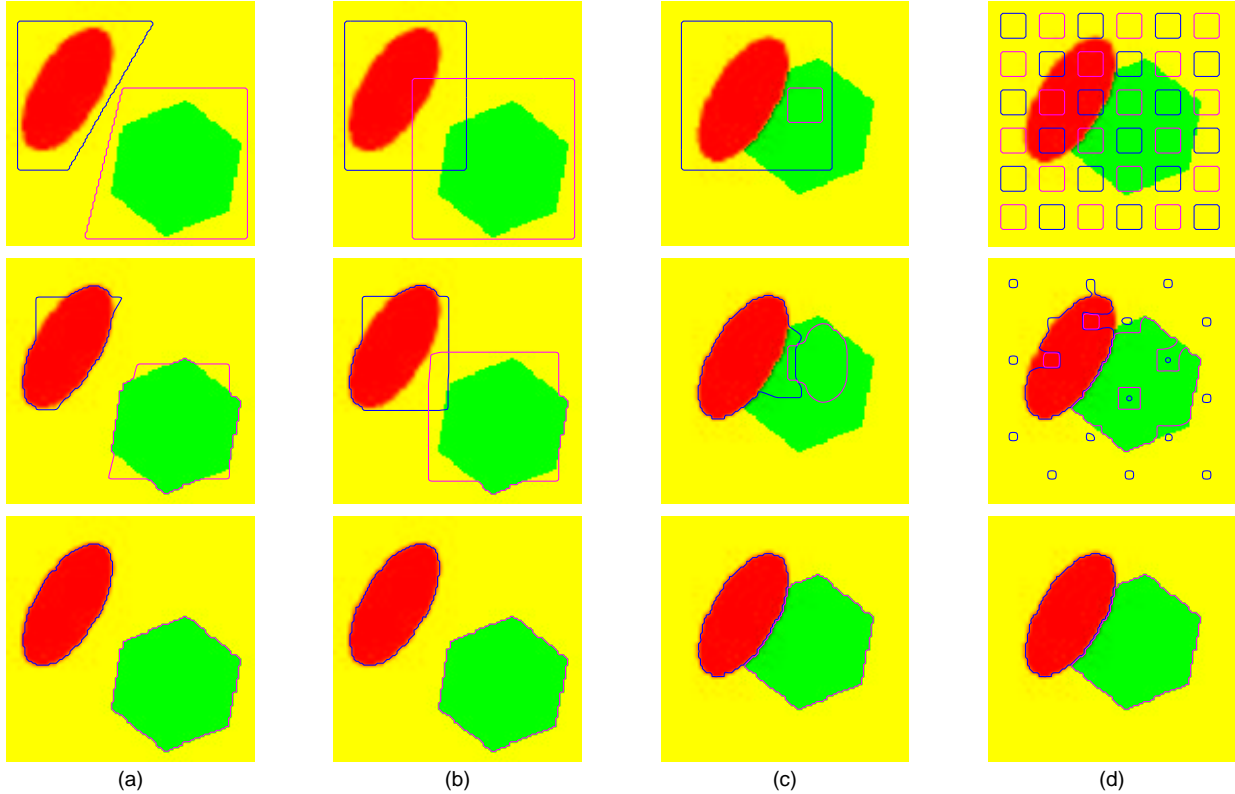


Figure 7: The coupled flows (14) and (15) on a red-green image: (a) Two separated initial contours flow interdependently toward separated regions. (b) Two overlapping initial contours flow toward separated regions. (c) Two overlapping initial contours flow toward neighboring regions. (d) Uniform seeding with two sets of contours comprises an automatic initialization technique.

the use of curve evolution rather than region-based methods. Level set implementations in particular (Section 5), allow automatic merging and splitting of initial contours, whereas region competition requires an additional “outer loop” to check whether pairs of neighboring regions should be merged to decrease the MDL energy functional. By using separate level set functions for \vec{C}_u and \vec{C}_v , regions are permitted to overlap as well. This approach precludes the development of triple points and avoids other geometric difficulties inherent to discrete region-based algorithms (e.g. the computation of curvature).

We see an even sharper contrast between the two models when considering the interdependence of evolving regions. In region competition, the evolution of a particular region depends only upon its own statistics and those of its immediate neighbors. In the ternary model, the evolution of a particular region depends upon the statistics of every single region, including those further away. In this sense, the coupled evolution equations (14) and (15) comprise a more global model for segmentation. On the other hand, the need for vector-valued statistics imposes a restriction on the types of data acceptable to our algorithm, a restriction that does not apply to region competition which can handle three (or more) classes given a single statistic. We note, however, that the need for a vector-valued statistic does

not necessarily require a vector-valued image. Ternary flows may be applied to greyscale images, for example, by considering both means and variances.

4.3 More general ternary flows

We now modify the flows (14) and (15) and their associated energy functional (12) to handle more general forms of trimodal imagery.

First, we allow the vector-valued data I to take its values in \mathbf{R}^n where $n \geq 2$ as opposed to just \mathbf{R}^2 . Unfortunately, the determinant in (12) no longer makes sense when $n > 2$. However, three noncollinear points, $I^a, I^b, I^c \in \mathbf{R}^n$ still comprise a triangle in \mathbf{R}^n , and $u, v, w \in \mathbf{R}^n$, as convex combinations of these values, will always lie inside this triangle (within the context of its two-dimensional plane). We may therefore generalize the ternary energy functional with the same goal of maximizing the area of the triangle T_{uvw} :

$$|\text{area}(T_{uvw})| = \frac{1}{2} \|u - w\| \|v - w\| \sin \theta$$

$$4 \text{ area}^2(T_{uvw}) = \|u - w\|^2 \|v - w\|^2 - ((u - w) \cdot (v - w))^2$$

where θ denotes the angle between $u - w$ and $v - w$.

Next, we attach a geometric penalty on the lengths of \vec{C}_u and \vec{C}_v (as in Section 2.3) to handle the presence of zero-mean noise in the image. In general one may penalize the two lengths differently; here we consider an equal penalty and rewrite (12) more generally as

$$E = -2 \text{ area}^2(T_{uvw}) + \alpha \left(\int_{\vec{C}_u} ds + \int_{\vec{C}_v} ds \right) \quad (16)$$

where $\alpha \geq 0$. We now use the previous expression to compute the variation of the first term

$$\nabla(2 \text{ area}^2(T_{uvw})) = \{\bar{w} \cdot \nabla u + \bar{u} \cdot \nabla v + \bar{v} \cdot \nabla w\} \vec{N}$$

with the following definitions:

$$\nabla u = (\nabla u_1 \cdot \vec{N}, \dots, \nabla u_n \cdot \vec{N})$$

(likewise for ∇v and ∇w)

$$\begin{aligned} \bar{u} &= \tilde{u} - \tilde{v} & \tilde{u} &= \hat{u}(\hat{v} \cdot \hat{w}) & \hat{u} &= u - v \\ \bar{v} &= \tilde{v} - \tilde{w} & \tilde{v} &= \hat{v}(\hat{w} \cdot \hat{u}) & \hat{v} &= v - w \\ \bar{w} &= \tilde{w} - \tilde{u} & \tilde{w} &= \hat{w}(\hat{u} \cdot \hat{v}) & \hat{w} &= w - u \end{aligned}$$

Since $\nabla_{\vec{C}_u} v = \nabla_{\vec{C}_v} u = 0$ the gradient descent equations for E become

$$\frac{d\vec{C}_u}{dt} = \left\{ \sum_{i=1}^n \left(\bar{w}_i \frac{I_i - u_i}{A_u} - \bar{v}_i (1 - \chi_v) \frac{I_i - w_i}{A_w} \right) - \alpha \kappa_u \right\} \vec{N}_u \quad (17)$$

$$\frac{d\vec{C}_v}{dt} = \left\{ \sum_{i=1}^n \left(\bar{u}_i \frac{I_i - v_i}{A_v} - \bar{w}_i (1 - \chi_u) \frac{I_i - w_i}{A_w} \right) - \alpha \kappa_v \right\} \vec{N}_v \quad (18)$$

where κ_u and κ_v denote the signed curvatures of \vec{C}_u and \vec{C}_v respectively.

4.4 Segmenting an arbitrary number of regions

In general one may wish to partition an image domain into m different types of regions, where m is an arbitrarily large number. In this subsection we discuss two different approaches based on the paradigms presented thusfar.

The first approach is a straight-forward generalization of the binary and ternary models. One may adhere to the same philosophy of associating the preferred segmentation with a maximal separation of some statistic over each region. To do this, a vector-valued statistic, U , with at least $m - 1$ components is required. If the m distinct values, U^1, \dots, U^m , of this statistic constitute a set of geometrically independent points in the preferred segmentation of the image, and if the statistic is chosen such that an arbitrary segmentation yields values u^1, \dots, u^m , which are convex combinations of U^1, \dots, U^m (which is the case if we are considering means of a vector-valued image) then the natural energy functional will relate to the volume of the $m - 1$ dimensional simplex whose vertices are given by u^1, \dots, u^m . The corresponding gradient flow equations will yield a coupled evolution of $m - 1$ curves which tend to maximize the volume of this simplex, with the interiors of each curve representing $m - 1$ regions and their mutual exteriors representing the m 'th region.

A second approach would be to use the binary or ternary models within the context of a dyadic or triadic region splitting, segmentation algorithm. The idea is as follows. First use a binary flow (or ternary or higher) to segment the initial image into two regions. Now employ two different binary flows, one over each of these segmented regions, to obtain four regions, and so on. A nice feature of this approach when implemented via level set techniques, is that the same level set may be used to simultaneously evolve each independent set of curves during a given stage of the algorithm, since each of the sets evolve over disjoint domains.

5 Numerical Implementations

In this section we address the numerical issues concerning the implementation of the flows presented in this paper. There are basically two problems that must be addressed. First, contours may undergo topological changes (merging or splitting) as they evolve. Second, the first order, data driven terms in these flows do not in general, on account of their nonlinearity, admit smooth solutions for all time even when given smooth initial conditions. If we omit the second order, curvature driven term, the characteristics of the resulting first order P.D.E. will typically intersect (in finite time), at which point, the solution may no longer be uniquely continued. When this occurs, the evolving curve develops points which are no longer differentiable (e.g. corners). At such points, some sort of weak solution is required in order to continue propagating the interface. If we now reintroduce the curvature driven term for its regularizing effect and determine limit of smooth solutions as we allow the strength of the curvature term to go to zero, we obtain a unique, physically meaningful, weak solution known as the *entropy solution* in the theory of hyperbolic conservation laws [16, 23] or the *viscosity solution* in the more general theory of Hamilton-Jacobi equations [9, 10, 17]. The level set methods of Osher and Sethian [24, 26], in conjunction with upwind, monotone differencing schemes [18, 23, 26] offer a natural and numerically reliable implementation of these solutions within a context that handles topological changes in the interface without

any additional effort.

The basic idea of the level set approach is to embed the contour as the level set of a graph $\Psi : \mathbf{R}^2 \rightarrow \mathbf{R}$ and then evolve the graph so that this level set moves according to the prescribed flow. In this manner, the level set of interest may develop singularities and change topology while Ψ itself remains smooth and maintains the form of a graph. Formulating the correct evolution for Ψ amounts to solving

$$\frac{d\Psi}{dt} = \Psi_t + \nabla\Psi \cdot \frac{d\vec{C}}{dt} = 0$$

so that the interface (the level set of interest) maintains a constant value as the graph, Ψ , evolves. The level set implementation of (4), for example, would look like

$$\Psi_t = -\frac{d\vec{C}}{dt} \cdot \nabla\Psi = \left\{ (v - u)\left(\frac{I - u}{A_u} + \frac{I - v}{A_v}\right) + \alpha\kappa \right\} \vec{N} \cdot \nabla\Psi.$$

Choosing the zero level set of Ψ to define \vec{C} and choosing Ψ to be negative inside of \vec{C} and positive outside of \vec{C} , allows us to write $\vec{N} = \nabla\Psi/\|\nabla\Psi\|$ and $\kappa = \nabla \cdot (\nabla\Psi/\|\nabla\Psi\|)$ and therefore

$$\Psi_t = \left\{ (v - u)\left(\frac{I - u}{A_u} + \frac{I - v}{A_v}\right) + \alpha\nabla \cdot \left(\frac{\nabla\Psi}{\|\nabla\Psi\|}\right) \right\} \|\nabla\Psi\|. \quad (19)$$

The level surface evolution shown above must be interpreted carefully since the curve evolution (4) depends only upon the values of I along the curve itself (the zero level set), whereas (19) make use of I everywhere. If we desire the evolution of every level set to follow the evolution of the zero level set, then we must extend the non-geometric influences (in this case, the values of the image) on the zero level set to the remaining level sets when propagating the level set function Ψ . Following the approach described in [18], we interpret I in (19) to be the intensity of the image over the closest point of the zero level set. See [26] for some other possibilities.

Finally, given an initial curve, one must generate an initial level set function. The curve only specifies the location of the zero level set of Ψ , and there is no unique way to initialize Ψ away from the curve. A well known scheme [18, 26, 27] is to use a signed distance function: the initial value of Ψ at each point is chosen to be the signed distance between that point and the initial curve, using negative distances for points inside the curve and positive distances for points outside the curve.

6 Simulations

The simulations presented up to this point (using synthetically generated images) were primarily intended to illustrate selected properties and behaviors of binary and ternary flows. In this section, we evaluate the performance of these flows on real-world images. We also discuss a “local” implementation of these flows which performs well on images that are not globally bimodal (or trimodal, etc...).

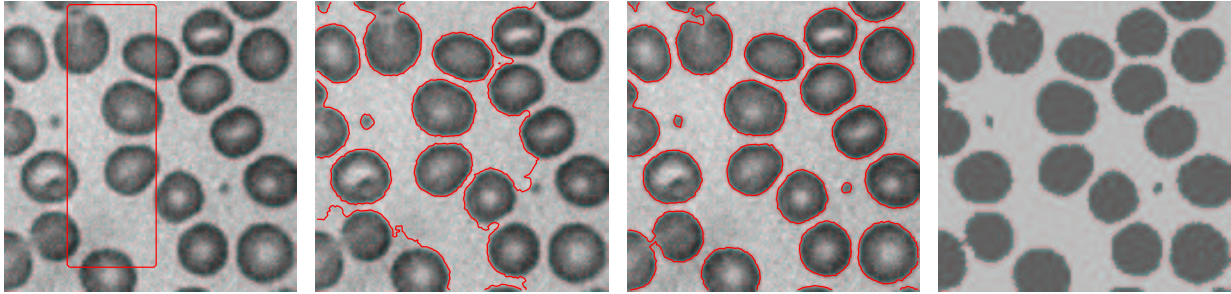


Figure 8: Multiple red blood cells are captured by a single contour using flow (4).

Flow (4) is used in Fig. 8, to segment a microscopic image of red blood cells, providing a compelling demonstration of the topological transitions allowed by its level set implementation (19). A single initial contour appears in the first frame; the multiple steady state contours and the resulting segmentation, showing the steady state mean intensity values, appear in the last two frames.

The synthetic aperture radar (SAR) image of a forest's tree line in Fig. 9 constitutes a bimodal image of a rather different nature. Means cannot be used here to distinguish one region from the other. The forest region in the lower left half of the image and the grassy region in the upper right half of the image give rise to two different textures with approximately the same greyscale mean, but with different variances. Flow (9), therefore, is able to segment the image quite successfully by separating variances rather than means.

Means and variances may also be used together in this methodology as components of a two-dimensional vector which must be chosen to minimize the energy functional (5). However, due to the dissimilarity between these two statistics, their first variations have different forms. Thus the gradient flow equation is not given by (6) but by a hybrid flow using the sum of the image-driven terms of (4) and (9). Such a flow was used to capture the tadpole dermal cells in the optical coherence tomography (OCT) image of Fig. 10.

We now present an example of a bimodal image for which our algorithm fails. The mammogram image in Fig. 11 shows a cyst as a bright region over the darker background tissue. However, the brightness of the cyst is not very uniform. The center of the cyst is

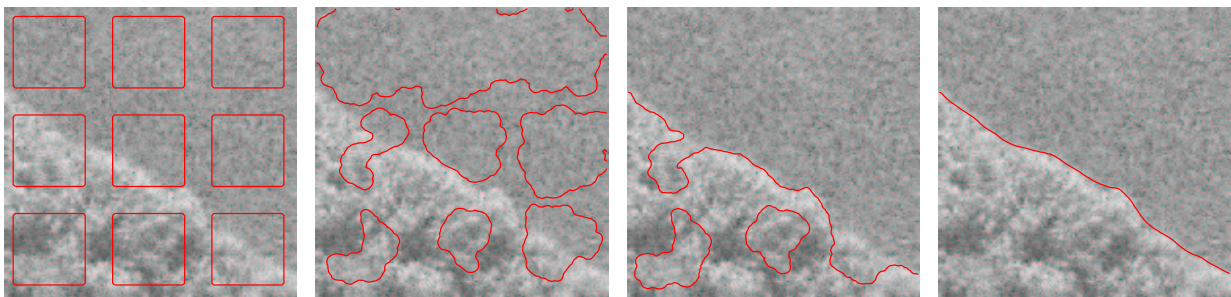


Figure 9: The tree line shown here is captured using flow (9) to separate variances.

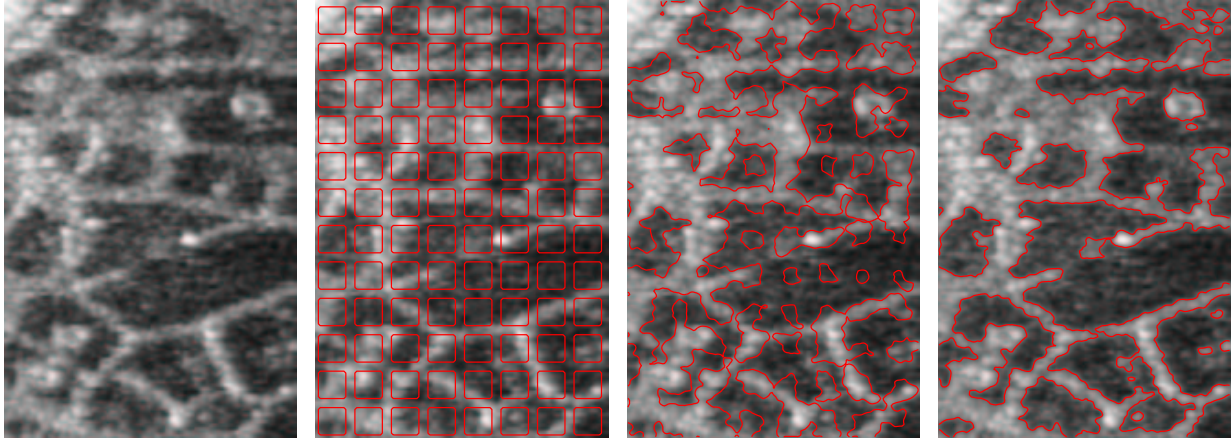


Figure 10: Both means and variances are used to segment this OCT image of tadpole cells (image courtesy: S. Boppart and J. Fujimoto [3, 11] of MIT).

brighter than the portions near its surface. This, coupled with the fact that the area of the background tissue region is relatively larger, causes flow (4) to capture the “hot spot” in the middle of the cyst. This happens because the slight increase in the mean of the background region incurred by including the outer part of the cyst is more than offset by the larger increase in the mean of the foreground region incurred by excluding the outer part of the cyst. In this example, we would expect flow (11), a special form of the Zhu-Yuille region competition flow [32], to perform better since the image-based forces in (11) are not weighted by the areas of the two regions. An alternative strategy would be to compute the means u and v in (4) only using the values of the image within a given radius of the evolving curve. If the radius is small enough, the flow will be “unaware” of the relatively brighter spot in the middle of the cyst. Indeed, a local implementation of (4), using a radius of 10 pixels, is shown capturing the boundary of the cyst more accurately in Fig. 12.

This local implementation technique allows us to use binary flows semiautomatically (i.e. with strategic initial contour placement) on images which are not actually bimodal, but contain bimodal subregions. The CT bone image in Fig. 13, for example, was segmented using a local implementation of (4) with a radius of 10 pixels. Ternary flows may also be

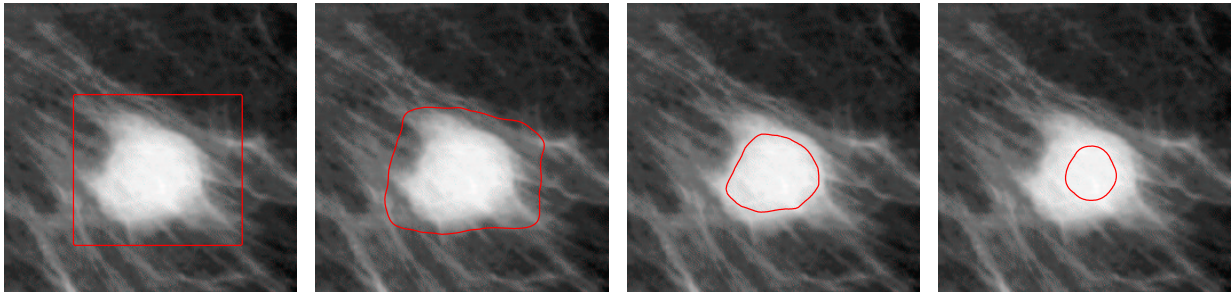


Figure 11: An exact implementation of flow (4) captures the brightest spot inside the cyst.

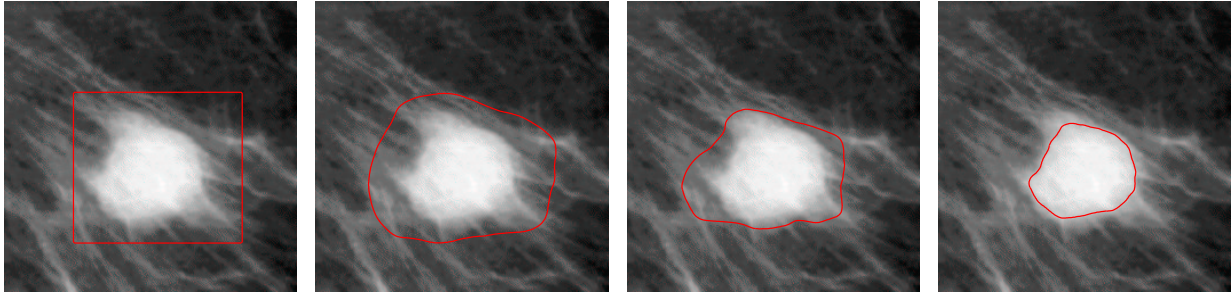


Figure 12: A “local” implementation of (4) does a better job of capturing the entire cyst.

implemented locally for the purpose of semiautomatic segmentation. The coupled ternary flows (17) and (18) were used to segment the clouds, the sky, and the B-2 bomber from the color image shown in Fig. 14.

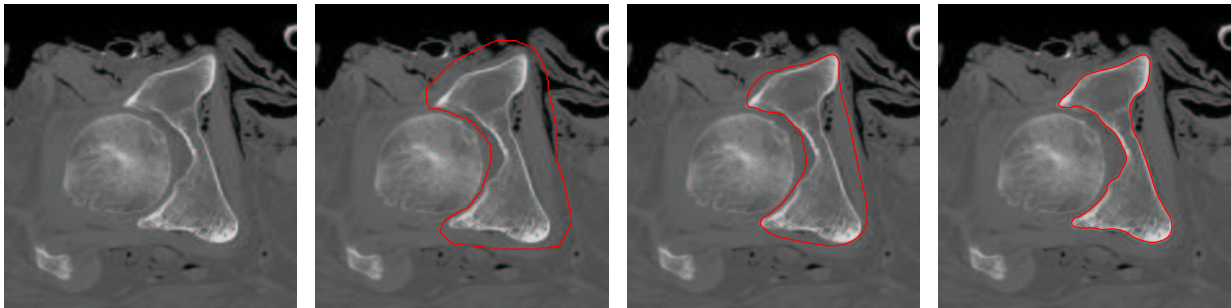


Figure 13: A “local” implementation of (4) captures the bone in this non-bimodal CT image.

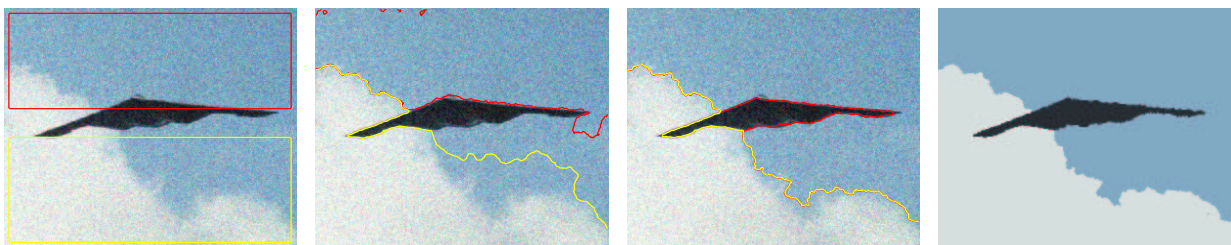


Figure 14: The B-2 bomber, clouds, and sky are captured by coupled flows (17) and (18).

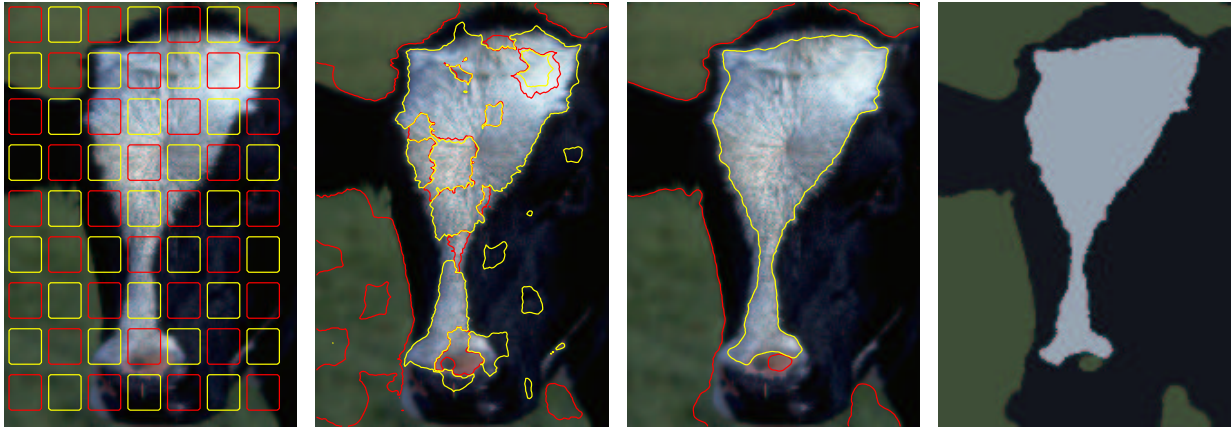


Figure 15: A “global-to-local” implementation of the coupled flows (17) and (18) is used to segment this cow photograph (photo courtesy: Philip Greenspun, <http://photo.net/philg>).

Finally, the local technique can also be used to segment automatically by seeding the image with regularly spaced initial contours and choosing a radius large enough so that the initial means inside and outside the contours are computed using every single pixel in the image. Then as the contours evolve, smaller portions of the image are used to compute these statistics via a purely data adaptive selection process. The color image in Fig. 15 was segmented in this way using a local implementation of the ternary flows (17) and (18).

7 Conclusions

We have presented in this paper a novel statistical approach to snakes for the segmentation of images which are known *a priori* to consist of a given number of regions distinguishable by a given set of statistics. The resulting gradient flows, derived from *deterministic* considerations, were designed to essentially pull the values of these statistics as far apart as the data in a given image would allow, subject to geometric constraints on the length of the active contour(s).

We began by introducing a number of “binary flows” for segmenting bimodal images and found that the resulting curve evolution equations resembled those associated with the region competition algorithm of Zhu and Yuille [31, 32]. The different philosophies behind our deterministic approach and their information theoretic approach were then connected through a probabilistic reformulation (Section 3.1) of the initial binary flow (Section 2.1) which led to precisely the region competition flow (for the special case of two regions and a prior Gaussian distribution model with unit variance). The formulations were seen to diverge when more than two regions were considered. This led to coupled curve evolution equations in which the evolution of a particular region depended not only upon its neighboring regions, but upon every other region bounded by every other curve.

Two key attractions of the flows in this paper were a natural use of both local and global information in the image and a deliberate avoidance of differential operators for detecting edges. In addition, our adherence to separate (although coupled) curve evolution equations

enabled the use of level set techniques in the implementation of our flows. This allowed a more natural implementation of curvature terms when compared to discrete region-based schemes while avoiding the algorithmic complexities associated with marker particle schemes in the event of topological changes.

To summarize, we have outlined a very general curve evolution approach to segmentation that clusters pixels in an image based upon both geometric and statistical considerations. The performance of our algorithm depends upon how well the chosen set of statistics is able to distinguish the various regions within a given image. Specifically, we have demonstrated the use of means, variances, and orientations as the discriminating statistics. However, our approach may be applied to any computed statistic. This fact underlies most of our future research interests in this algorithm, namely to investigate the use of wavelets, filter banks, and other feature detectors to obtain the statistics to drive binary, ternary, and analogous flows.

A Appendix: Gradient Flows for Region Integrals

A.1 Integrals over smooth regions

Here we derive the first variation and the corresponding L_2 gradient flow for functionals of the form

$$E = \iint_R f dA$$

where R denotes the region corresponding to the interior of a smooth, closed curve \vec{C} , and $f : \mathbf{R}^2 \rightarrow \mathbf{R}$ is a continuous function. (See also [14] for a similar computation.) Let $\vec{C}(p, t)$ denote a family of smooth curves, where $p \in [0, 1]$ parameterizes each individual curve and where t parameterizes the family. Using the divergence theorem, we may rewrite E (as a function of t) via the following contour integral around $\vec{C}(p, t)$:

$$E(t) = \frac{1}{2} \int_{\vec{C}} \langle \vec{F}, \vec{N} \rangle ds = \frac{1}{2} \int_0^1 \langle \vec{F}, \vec{N} \rangle \|\vec{C}_p\| dp$$

where s is the arclength parameter, \vec{N} is the outward unit normal, and $\vec{F} = (F_1, F_2)$ is the vector field formed by integrating f along each coordinate direction.

$$F_1(x, y) = \int_0^x f(\lambda, y) d\lambda \quad \text{and} \quad F_2(x, y) = \int_0^y f(x, \lambda) d\lambda$$

Then, using the fact that $\vec{N} \|\vec{C}_p\| = J \vec{C}_p$ where $J = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ is the 90 degree rotation matrix we may express the first variation of E as

$$E'(t) = \frac{1}{2} \int_0^1 (\langle \vec{F}_t, J \vec{C}_p \rangle + \langle \vec{F}, J \vec{C}_{pt} \rangle) dp.$$

Using integration by parts on the second term of the integrand and then applying the chain rule on the derivatives of \vec{F} yields

$$E'(t) = \frac{1}{2} \int_0^1 (\langle (D\vec{F})C_t, J\vec{C}_p \rangle - \langle (D\vec{F})C_p, J\vec{C}_t \rangle) dp$$

where $D\vec{F}$ denotes the 2x2 Jacobian matrix of \vec{F} . Rearranging and combining terms to isolate \vec{C}_t on one side of the inner-product yields

$$E'(t) = \frac{1}{2} \int_0^1 \langle \vec{C}_t, [(J^T(D\vec{F}))^T - J^T(D\vec{F})]\vec{C}_p \rangle dp.$$

Since the matrix inside the square brackets is antisymmetric, it must have the form ωJ , and hence

$$E'(t) = \frac{1}{2} \int_0^1 \langle \vec{C}_t, \omega J\vec{C}_p \rangle dp = \frac{1}{2} \int_C \langle \vec{C}_t, \omega \vec{N} \rangle ds.$$

Finally, a straightforward computation shows that $\omega = \nabla \cdot F = 2f$, hence

$$E'(t) = \int_{\vec{C}} \langle \vec{C}_t, f\vec{N} \rangle ds,$$

from which it is clear that the corresponding flow of maximum ascent is given by

$$\vec{C}_t = f\vec{N}.$$

A.2 Integrals over nonsmooth regions

Here, using the notation of Section 4.1, we derive the partial variations of the mean intensity w of an image $I(x, y)$ over the mutual exterior of two curves \vec{C}_u and \vec{C}_v (we denote this region by R^w) given that the curves may overlap each other. As in Section 4.1, we assume that $I(x, y) \in \mathbf{R}^2$, hence $w = (w_1, w_2)$.

We may express the first component of w as $w_1 = S_{w_1}/A_w$ where $S_{w_1} = \int_{R^w} I_1 dA$ and $A_w = \int_{R^w} dA$. Unfortunately, this region of integration does not always have a smooth boundary if \vec{C}_u and \vec{C}_v overlap, so the divergence theorem used in Appendix A.1 no longer applies when computing the variations of these integrals. However, to compute the partial variations with respect to \vec{C}_u we may rewrite these integrals as $S_{w_1} = \int_{\Omega \setminus R^u} I_1(1 - \chi_v) dA$ and $A_w = \int_{\Omega \setminus R^u} (1 - \chi_v) dA$ where Ω denotes the overall domain of I , R^u the interior of \vec{C}_u and χ_v the characteristic function over R^v (the interior of \vec{C}_v). We have now expressed these quantities as integrals over a region whose boundary is \vec{C}_u , which we assume to be smooth. However, we still cannot technically use the final result of A.1 to compute their variations since χ_v is not continuous. On the other hand, we may consider a sequence of continuous approximations to χ_v which converge pointwise to χ_v and construct two sequences of variations obtained (using the result in A.1) by replacing χ_v with its continuous approximation in the integral expressions for S_{w_1} and A_w . The resulting sequences of approximate variations will converge pointwise to the following discontinuous partial variations:

$$\nabla_{\vec{C}_u} S_{w_1} = -I_1(1 - \chi_v)\vec{N}_u \quad \text{and} \quad \nabla_{\vec{C}_u} A_w = -(1 - \chi_v)\vec{N}_u.$$

Finally, by the same technique, we can compute the discontinuous limit of approximate variations of w_1 to obtain⁵

$$\nabla_{\vec{c}_u} w_1 = \nabla_{\vec{c}_u} \frac{S_{w_1}}{A_w} = \frac{\nabla_{\vec{c}_u} S_{w_1}}{A_w} - \frac{S_{w_1} \nabla_{\vec{c}_u} A_w}{A_w^2} = -\frac{I_1 - w_1}{A_w} (1 - \chi_v) \vec{N}_u.$$

References

- [1] A. Barron, J. Rissanen, and B. Yu, “The Minimum Description Length Principle in Coding and Modeling,” *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2743–2760, Oct. 1998.
- [2] A. Blake and A. Yuille, *Active Vision*, MIT Press, Cambridge, MA, 1992.
- [3] S. Boppart, B. Bouma, C. Pitris, J. Southern, M. Brezinski, and J. Fujimoto, “In vivo cellular optical coherence tomography imaging,” *Nature Medicine*, vol. 4, pp. 861–865, July 1998.
- [4] V. Caselles, F. Catte, T. Coll, and F. Dibos, “A geometric model for active contours in image processing,” *Numerische Mathematik*, vol. 66, pp. 1–31, 1993.
- [5] V. Caselles, R. Kimmel, and G. Sapiro, “Geodesic snakes,” *Int. J. Computer Vision*, 1998.
- [6] A. Chakraborty and J. Duncan, “Game-Theoretic Integration for Image Segmentation,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 21, no. 1, pp. 12–30, Jan. 1999.
- [7] A. Chakraborty, L. Staib, and J. Duncan, “Deformable Boundary Finding in Medical Images by Integrating Gradient and Region Information,” *IEEE Trans. Medical Imaging*, vol. 15, no. 6, pp. 859–870, Dec. 1996.
- [8] L. Cohen, “On active contour models and balloons,” *CVGIP: Image Understanding*, vol. 53, pp. 211–218, 1991.
- [9] M. Crandall, H. Ishii, and P. Lions, “Users guide to viscosity solutions of second order partial differential equations,” *Bulletin of Amer. Math. Soc.*, vol. 27, pp. 1–67, 1992.
- [10] W. Fleming and H. Soner, *Controlled Markov processes and viscosity solutions*. Springer-Verlag, New York, 1993.
- [11] D. Huang, E. Swanson, C. Lin, J. Schuman, W. Stinson, W. Chang, M. Hee, T. Flotte, K. Gregory, C. Puliafito, and J. Fujimoto, “Optical coherence tomography,” *Science*, vol. 254, pp. 1178–1181, November 1991.

⁵We have abused notation slightly in this expression by implying the use of the chain rule on the limits of sequences which converge to S_{w_1} and A_w . Technically, this chain rule operation is performed, legally, on the members of the two sequences to generate a new sequence which converges pointwise to the final expression shown here.

- [12] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: active contour models,” *Int. Journal of Computer Vision*, vol. 1, pp. 321–331, 1987.
- [13] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, “Conformal Curvature Flows: From Phase Transitions to Active Vision,” *Arch. Rational Mech. Anal.*, vol. 134, pp. 275–301, 1996.
- [14] K. Siddiqi, Y. Lauziere, A. Tannenbaum, and S. Zucker, “Area and length minimizing flows for segmentation,” *IEEE Trans. Image Processing*, vol. 7, pp. 433–444, 1998.
- [15] Y. Leclerc, “Constructing stable descriptions for image partitioning,” *Int. J. Computer Vision*, vol. 3, pp. 73–102, 1989.
- [16] R. J. LeVeque, *Numerical Methods for Conservation Laws*, Birkhäuser, Boston, 1992.
- [17] P. L. Lions, *Generalized Solutions of Hamilton-Jacobi Equations*, Pitman Publishing, Boston, 1982.
- [18] R. Malladi, J. Sethian, and B. Vemuri, “Shape modeling with front propagation: a level set approach,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 158–175, 1995.
- [19] J. M. Morel and S. Solimini, *Variational Methods in Image Segmentation*, Birkhauser, Boston, 1995.
- [20] D. Mumford, “Pattern Theory: A unifying perspective,” *Perception as Bayesian Inference*, edited by David C. Knill and Whitman Richards, Cambridge University Press, 1996.
- [21] D. Mumford and J. Shah, “Optimal approximations by piecewise smooth functions and associated variational problems,” *Communications in Pure and Applied Mathematics*, vol. 42, no. 4, 1989.
- [22] D. Mumford and J. Shah, “Boundary detection by minimizing functionals,” *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, 1985.
- [23] S. Osher, “Riemann solvers, the entropy condition, and difference approximations,” *SIAM J. Numer. Anal.*, vol. 21, pp. 217–235, 1984.
- [24] S. Osher and J. Sethian, “Fronts propagation with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations,” *Journal of Computational Physics*, vol. 79, pp. 12–49, 1988.
- [25] R. Ronfard, “Region-Based Strategies for Active Contour Models,” *Int. J. Computer Vision*, vol. 13, no. 2, pp. 229–251, 1994.
- [26] J. Sethian, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Material Science*, Cambridge University Press, 1996.
- [27] M. Sussman, P. Smereka, and S. Osher, “A Level Set Method for Computing Solutions to Incompressible Two-Phase Flow,” *J. Comp. Phys.*, vol. 114, pp. 146–159, 1994.

- [28] H. Tek and B. Kimia, “Image segmentation by reaction diffusion bubbles,” *Proc. Int. Conf. Computer Vision*, pp. 156–162, 1995.
- [29] D. Terzopoulos and A. Witkin, “Constraints on deformable models: recovering shape and non-rigid motion,” *Artificial Intelligence*, vol. 36, pp. 91–123, 1988.
- [30] A. Yezzi, S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum, “A Geometric Snake Model for Segmentation of Medical Imagery”, *IEEE Trans. Medical Imaging*, vol. 16, no. 2, pp. 199–209, 1997.
- [31] S. C. Zhu, T. S. Lee, and A. L. Yuille, “Region Competition: Unifying snakes, Region Growing, and Bayes/MDL for Multiband Image Segmentation,” *Proc. Int. Conf. Computer Vision*, pp. 416–423, 1995.
- [32] S. Zhu and A. Yuille, “Region Competition: Unifying snakes, Region Growing, and Bayes/MDL for Multiband Image Segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, pp. 884–900, Sep. 1996.