

A Probabilistic Framework for Curve Evolution

Vedrana Andersen Dahl^(✉) and Anders Bjorholm Dahl

Department of Applied Mathematics and Computer Science,
Technical University of Denmark, Kongens Lyngby, Denmark
`{vand,abda}@dtu.dk`

Abstract. In this work, we propose a nonparametric probabilistic framework for image segmentation using deformable models. We estimate an underlying probability distributions of image features from regions defined by a deformable curve. We then evolve the curve such that the distance between the distributions is increasing. The resulting active contour resembles a well studied piecewise constant Mumford-Shah model, but in a probabilistic setting. An important property of our framework is that it does not require a particular type of distributions in different image regions. Additional advantages of our approach include ability to handle textured images, simple generalization to multiple regions, and efficiency in computation. We test our probabilistic framework in combination with parametric (snakes) and geometric (level-sets) curves. The experimental results on composed and natural images demonstrate excellent properties of our framework.

1 Introduction

A well studied variational approach to image segmentation is the Mumford-Shah model [10], often combined with a level-set curve representation [11]. Especially the two-phase piecewise constant case has been extensively studied. The goal is to partition the image into two regions, each represented by a single intensity value. The curve evolution algorithm proposed by Chan and Vese [3,4] estimates the mean intensities of image regions. The curve is then evolved depending on whether intensities under the curve lie closer to one, or the other, estimated mean.

Our segmentation approach is strongly inspired by [3,4] and a related model from [13,15]. Like them, we obtain an estimate on image regions, followed by the deformation of the curve. However, instead of operating on pixel intensities, we estimate the underlying distribution of image features. By doing so, we obtain a pixel-wise probability of belonging to one, or the other, region, and we subsequently use those probabilities for deforming the curve.

An important property of our segmentation model is that we estimate the underlying distributions without making any assumptions, e.g. assuming it to be unimodal. Instead, our model uses distributions that are learned and dynamically adapted in a nonparametric way. In this aspect, our work resembles curve evolution proposed in [9], where they evolve the curve by assuming that segmented regions are homogeneous. We assume that regions have different distributions.

Furthermore, we perform averaging on clusters of image patches and are therefore able to handle textured regions. The typical approach to texture segmentation involves mapping the image to a texture descriptor space. Here the assumption is that descriptors within textures are similar while they differ between textures. Such an approach was already suggested by [4] using texture orientation, and has been extended in [12] with the structure tensor. For better performance, the scale of the structure tensor is automatically estimated in [2], while [1] utilizes diffusion.

More recently [7] suggested to use sparse dictionaries together with an user-initiated active contour. Another patch-based approach [14] uses a patch subspace model for active contours. For each segment they learn a set of orthonormal basis vectors, and use this for modeling image patches and evolving the curve.

Our texture model was introduced in [5,6]. In this approach, clustering of image patches allows for computation of pixel-wise probabilities, which then deform the curve. In the work presented here, we formulate a more general histogram-based framework for using empirical probabilities in both intensity-based and texture-based deformable models. This study also brings the mathematical foundations which prove the correctness of our approach.

2 Method

We consider an image where the task is to separate the foreground from the background, and we use subscripts F and B for the corresponding image entities, such as for the image domain Ω consisting of Ω_F and Ω_B . At the same time, a curve C divides the image into inside and outside region, and for those regions we use subscripts in and out. In few places we use the set notation, where $\Omega_{\text{in} \cap F}$ is a part of the foreground region which is inside the curve.

The terms we use for image regions are chosen for the simplicity of the presentation, but the models we describe do not rely on any special topology of the object or the curve, and can handle a general two-phase segmentation.

The segmentation energy E should be defined such that the desired segmentation has a minimal energy. Segmentation is obtained by moving the curve to minimize the energy, and the essence of the approach is in deriving suitable curve deformation forces $F = -\nabla E$.

The contributions to the segmentation energy may be divided into an external energy and an internal energy. The external energy is determined by the image and is the focus of our upcoming method description.

The internal energy is determined solely by the shape of the curve. A typical choice is a length of the curve $E_{\text{int}} = \alpha \int_C ds$, here weighted with parameter α . This discourages stretching of the curve with the force $F_{\text{int}} = -\alpha \kappa N$ where N denotes an outward unit normal, and κ is the signed curvature. Those regulatory forces are the key to success of deformable models. Therefore, even though expressions presented below show only external contributions, bear in mind that those are supplemented by internal forces.

2.1 Inspiration

Before defining our segmentation energy and curve deformation, we revise two models which inspired this work.

An external energy closely related to the two-phase piecewise constant Mumford-Shah model is

$$E_{\text{ext}} = \int_{\Omega_{\text{in}}} (I - m_{\text{in}})^2 dx + \int_{\Omega_{\text{out}}} (I - m_{\text{out}})^2 dx \quad (1)$$

where I is an image intensity as a function of the pixel position, while m_{in} and m_{out} are mean intensities of the inside and the outside region. This energy finds the best (in a squared-error sense) piecewise constant approximation of I . An evolution that will contract a curve toward a minimum is derived in [3] as

$$F_{\text{ext}} = \frac{1}{2}(m_{\text{in}} - m_{\text{out}})(I - m_{\text{in}} + I - m_{\text{out}})N. \quad (2)$$

A different energy proposed by [15] uses the distance between estimated mean intensities as a measure of region separation

$$E_{\text{ext}} = -\frac{1}{2}(m_{\text{in}} - m_{\text{out}})^2. \quad (3)$$

The corresponding flow, similar to (2) but with area normalization (we use A for areas of the regions), is

$$F_{\text{ext}} = (m_{\text{in}} - m_{\text{out}}) \left(\frac{I - m_{\text{in}}}{A_{\text{in}}} + \frac{I - m_{\text{out}}}{A_{\text{out}}} \right) N. \quad (4)$$

In case of an image truly consisting of two regions of constant intensity, the minimizer is the same for the two energy functions. However, under noise or blurred boundaries (2) minimizes the intra-region variance while (4) maximizes the inter-region difference.

2.2 Probability Framework

Let us first explain the probability framework on a simple intensity-based example. Assume that foreground region contains pixels with intensities that follow one discrete probability distribution, while background intensities follow another distribution. In our examples intensities are from $K = \{1, \dots, 256\}$.

Given an arbitrary curve, we can extract the absolute histogram of the inside and the outside region. Dividing those by the areas inside and outside the curve respectively, we obtain normalized frequencies $f_{\text{in}}(k)$ and $f_{\text{out}}(k)$. Those empirical probabilities will (to some degree, depending on the position of the curve) reflect the difference between the foreground and the background distributions.

If we consider a pixel of intensity $k \in K$, the probability of it belonging to the inside distribution is

$$p_{\text{in}}(k) = \frac{1}{Z} f_{\text{in}}(k), \quad (5)$$

where a normalization constant $Z(k) = f_{\text{in}}(k) + f_{\text{out}}(k)$ ensures that probabilities sum to 1.

Consider constructing a probability image P_{in} from the image I by replacing each pixel values k with $p_{\text{in}}(k)$, for all $k \in K$. This is illustrated for three images in Fig. 1. In the first image the foreground and the background regions follow two non-overlapping distributions. This demonstrates a situation where P_{in} is an excellent indicator of foreground-background segmentation. The second image shows two overlapping, but rather different distributions. Here P_{in} is still a good indicator for segmentation, especially if combined with some regularization. The last image shows two similar distributions, which is a challenging situation and motivates our patch-based approach in Sect. 2.5.

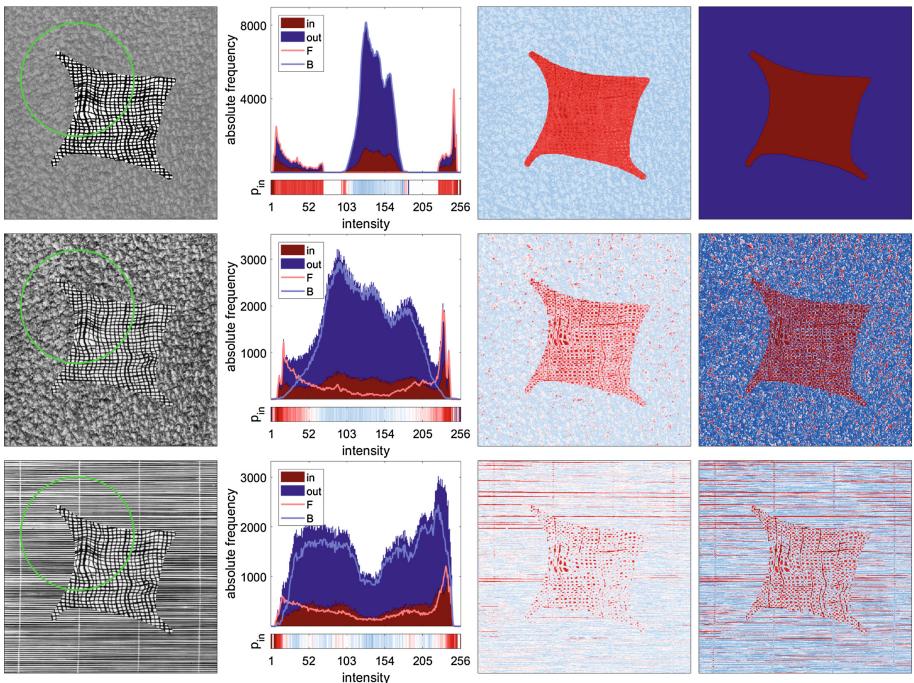


Fig. 1. Distributions of intensity values for the three input images show in the *left column*. The absolute histograms (*second column*) of each image are divided into inside and outside part according to the separation indicated on the input by the green curve. Foreground and background histograms are added as curves for comparison. The colored bars under the histograms show p_{in} in shades of blue for values under 0.5, white for 0.5 and red for values above 0.5. (The same color scheme is used throughout the paper.) *Third column* shows P_{in} computed for inside-outside separation indicated on the input image by the green curve, while the *rightmost column* shows the probability image when the curve aligns with the boundary between the foreground and the background. (Color figure online)

2.3 Fundamentals

Let us demonstrate two statements on probability distributions which are fundamental for our approach.

Statement 1 (Maximal distance). *Assume that f_F and f_B are probability distribution functions of foreground and background regions. A curve partitions the image in outside and inside, such that the induced subregions (e.g. $\Omega_{\text{in} \cap F}$) are representative. The total variation distance*

$$D(f_{\text{in}}, f_{\text{out}}) = \frac{1}{2} \sum_K |f_{\text{in}}(k) - f_{\text{out}}(k)| \quad (6)$$

is largest if the curve is aligned with the boundary between foreground and background.

Proof. Notice that f_{in} is a mixture density, i.e. a convex combination of f_F and f_B with the coefficients given by the proportions of foreground and background inside the curve, and similar is valid for f_{out} . We introduce convex weights $\lambda = A_{\text{in} \cap B}/A_{\text{in}}$ and $\mu = A_{\text{out} \cap F}/A_{\text{out}}$ and write

$$\begin{aligned} f_{\text{in}} &= (1 - \lambda)f_F + \lambda f_B \\ f_{\text{out}} &= \mu f_F + (1 - \mu)f_B. \end{aligned} \quad (7)$$

Consequently, the distance between f_{in} and f_{out} is smaller than the distance between f_F and f_B . More precisely, we have

$$D(f_{\text{in}}, f_{\text{out}}) = |1 - \lambda - \mu| \frac{1}{2} \sum_K |f_F(k) - f_B(k)| \quad (8)$$

which is maximal when $\lambda = \mu = 0$ corresponding to $\Omega_{\text{in}} = \Omega_F$ and $\Omega_{\text{out}} = \Omega_B$ (or the flipped solution for $\lambda = \mu = 1$). \square

Note that the distance between the normalized histograms for arbitrary pixel partitioning may be larger than this limit, if the separation in inside and outside is systematic. In the third case on Fig. 1, such situation would occur if inside contains all low-intensity pixels from both foreground and background. However, a curve should be rather convoluted to allow for such not-representative partitioning.

Statement 2 (Maximization). *Consider two empirical probability distribution functions (normalized histograms) f_{in} and f_{out} obtained by collecting intensity values inside and outside the curve. Assume there is a pixel with intensity \hat{k} in the outside region and $f_{\text{in}}(\hat{k}) > f_{\text{out}}(\hat{k})$. The total variation distance*

$$D(f_{\text{in}}, f_{\text{out}}) = \frac{1}{2} \sum_K |f_{\text{in}}(k) - f_{\text{out}}(k)|. \quad (9)$$

will increase if we move this pixel from outside to inside.

Proof. Values for normalized histograms after the move are

$$\begin{aligned} f'_{\text{in}}(k) &= \frac{A_{\text{in}}}{A_{\text{in}}+1} f_{\text{in}}(k) + \frac{1}{A_{\text{in}}+1} \delta(k - \hat{k}) \\ f'_{\text{out}}(k) &= \frac{A_{\text{out}}}{A_{\text{out}}-1} f_{\text{out}}(k) - \frac{1}{A_{\text{out}}-1} \delta(k - \hat{k}). \end{aligned} \quad (10)$$

We consider the sign of differences $(f_{\text{in}}(k) - f_{\text{out}}(k))$ and $(f'_{\text{in}}(k) - f'_{\text{out}}(k))$. For $k = \hat{k}$ the difference is, and will remain, positive. All negative differences will stay negative, while some positive differences for $k \neq \hat{k}$ might become negative. This allows us to partition K in three subsets

$$\begin{aligned} K^{++} &= \{k \mid f_{\text{in}}(k) > f_{\text{out}}(k) \text{ and } f'_{\text{in}}(k) \geq f'_{\text{out}}(k)\} \\ K^{+-} &= \{k \mid f_{\text{in}}(k) > f_{\text{out}}(k) \text{ and } f'_{\text{in}}(k) < f'_{\text{out}}(k)\} \\ K^- &= \{k \mid f_{\text{in}}(k) \leq f_{\text{out}}(k)\}. \end{aligned} \quad (11)$$

Furthermore, we define a sum $S_{\text{in}}^{++} = \sum_{K^{++}} f_{\text{in}}(k)$ and similar sums for S_{in}^{+-} , S_{in}^- , S_{out}^{++} , S_{out}^{+-} and S_{out}^- .

The total variation distance before the move now reduces to

$$D(f_{\text{in}}, f_{\text{out}}) = \frac{1}{2} (S_{\text{in}}^{++} - S_{\text{out}}^{++} + S_{\text{in}}^{+-} - S_{\text{out}}^{+-} + S_{\text{out}}^- - S_{\text{in}}^-) = S_{\text{out}}^- - S_{\text{in}}^-, \quad (12)$$

where last equality uses the fact that values of f_{in} and f_{out} sum to 1. For the total variation distance after the move we have

$$\begin{aligned} D(f'_{\text{in}}, f'_{\text{out}}) &= \frac{1}{2} \left(\frac{A_{\text{in}} S_{\text{in}}^{++}}{A_{\text{in}}+1} - \frac{A_{\text{out}} S_{\text{out}}^{++}}{A_{\text{out}}-1} + \frac{1}{A_{\text{in}}+1} + \frac{1}{A_{\text{out}}-1} + \right. \\ &\quad \left. + \frac{A_{\text{out}} S_{\text{out}}^{+-}}{A_{\text{out}}-1} - \frac{A_{\text{in}} S_{\text{in}}^{+-}}{A_{\text{in}}+1} + \frac{A_{\text{out}} S_{\text{out}}^-}{A_{\text{out}}-1} - \frac{A_{\text{in}} S_{\text{in}}^-}{A_{\text{in}}+1} \right) \\ &= \frac{A_{\text{out}}}{A_{\text{out}}-1} (S_{\text{out}}^- + S_{\text{out}}^{+-}) - \frac{A_{\text{in}}}{A_{\text{in}}+1} (S_{\text{in}}^- + S_{\text{in}}^{+-}) \end{aligned} \quad (13)$$

Comparing (12) to (13), and after removing inequalities which hold due to change in areas, we are left with needing to establish

$$\frac{A_{\text{in}}}{A_{\text{in}}+1} S_{\text{in}}^{+-} < \frac{A_{\text{out}}}{A_{\text{out}}-1} S_{\text{out}}^{+-}. \quad (14)$$

This is a consequence of $\hat{k} \notin K^{+-}$, (10) and (11). \square

Statement 2 shows that moving pixels according to the difference of normalized histograms increases the distance between the distributions, while Statement 1 claims that under the assumption of representative sub-regions this leads to foreground–background segmentation. If we assure that the curve is relatively smooth (regularized), this assumption will hold.

2.4 Probabilities for Curve Evolution

Inspired by the insights presented in Sect. 2.3 we propose external forces for curve evolution

$$F_{\text{ext}} = (P_{\text{in}} - P_{\text{out}})N, \quad (15)$$

where empirical probabilities P_{in} and P_{out} are estimated from inside and outside regions as in Sect. 2.2. Note that by using $(p_{\text{in}} - p_{\text{out}})$ instead of $(f_{\text{in}} - f_{\text{out}})$ we still move the curve in the correct direction, but we eliminate the effect of the overall pixel frequency. We plan to investigate the relationship between the two flows in the future work. Let us comment on a couple of aspects of our probabilistic approach.

First, the computation of P_{in} from the curve may be implemented in terms of a linear mapping. To see this, consider ordering image pixels in some way, e.g. column-wise, and defining a binary matrix \mathbf{B} by $b_{ik} = 1$ if a pixel i has an intensity value k , and 0 otherwise. Consider also a binary inside-outside mask defined by the curve, and unfolded using the same ordering as with image pixels. We denote this vector \mathbf{c}_{in} . The product $\mathbf{B}^T \mathbf{c}_{\text{in}}$ contains an absolute histogram of the inside region. We have

$$\mathbf{f}_{\text{in}} = \frac{\mathbf{B}^T \mathbf{c}_{\text{in}}}{A_{\text{in}}} \quad \text{and} \quad \mathbf{f}_{\text{out}} = \frac{\mathbf{B}^T (1 - \mathbf{c}_{\text{in}})}{A_{\text{out}}}. \quad (16)$$

The bold font indicates a vector, but its elements are values of $f_{\text{in}}(k)$. After element-wise normalization of vector elements we have a vector \mathbf{p}_{in} containing probabilities $p_{\text{in}}(k)$. The product $\mathbf{B} \mathbf{p}_{\text{in}}$ now contains elements of P_{in} , which only need to be arranged in the image grid.

Said in other words, a biadjacency matrix encoding a pixel-value relation defines a mapping from the image domain to the intensity domain and back. This is useful for an efficient implementation of our deformable model, and is important since many iterations might be needed during evolution.

The second aspect to mention is the discrete nature of our model. For an image with continuous intensities, e.g. from the interval $[0, 1]$ our approach still applies provided that we perform a suitable binning of pixels values. In the upcoming text we take this strategy a step further.

2.5 Patch-Based Probabilities

Looking back at the third case in Fig. 1 we notice that a distribution of image features might be a better descriptor for image regions. To use a frequency approach, we first decide on a suitable binning strategy. For this we extract M -by- M patches from the image I , choose a number of bins, and perform k-means clustering. We use Euclidian distance on vectors obtained by collecting patch intensities e.g. column-wise. This gives us a set of cluster centers (we call this a *dictionary*) and every cluster center can be rearranged on a grid and visualized as an image patch. Furthermore, for every patch from the image I we know which cluster it belongs to. This can be represented as an assignment

image S , which contains a bin number assigned to the central pixel of each image patch. (We reserve the value 0 for pixels at image boundary.)

We may proceed as if S is an intensity image, illustrated in the first part of Fig. 2. The probability of each patch is concentrated in a central pixel which creates abrupt changes in the probability image. One may argue for filtering P_{in} with M -by- M kernel, to dissolve each probability across a whole patch. Instead, we propose a solution which is better for an accurate placement of the segmentation boundary.

The final improvement in our patch-based probability model is motivated by the realization that some of the dictionary elements represent a spacial transition between the foreground and the background. We therefore allow that pixels in the dictionary patches have individual probabilities. In other words, the histogram now contains M^2 values per cluster, and it makes sense to visualize it as a collection of patches. To estimate a histogram we locate all image patches clustered together under a dictionary element, and for every pixel position within the patch we count how many times it appears inside (or outside) the curve. Dictionary probabilities computed in this way are shown in Fig. 2, third column. When going back from dictionary probabilities to the image, for every image pixel we consider all histogram values it contributed to, and average their contributions.

It is worth mentioning that calculation of the probability image from a curve still may be implemented in terms of a linear mapping. The biadjacency matrix \mathbf{B} should now represent the relation between image pixels and dictionary pixels. Each row of \mathbf{B} corresponds to a pixel from the image, while columns correspond to pixels in the dictionary (unwrapped in some order). The computation of dictionary probabilities is still obtain through multiplication with \mathbf{B}^T . For computation of image probabilities note that elements of $\mathbf{B}\mathbf{p}_{\text{in}}$ need to be divided with sum of rows from \mathbf{B} . The elements of P_{in} are therefore given by the product $\text{diag}(\mathbf{B}\mathbf{1})^{-1}\mathbf{B}\mathbf{p}_{\text{in}}$.

In other words, also when using patch probabilities, we can efficiently compute probabilities using a biadjacency matrix \mathbf{B} . And since \mathbf{B} does not change with curve evolution, it is enough to construct it just once, after clustering.

2.6 Multi-phase Segmentation

Our probabilistic framework can rather straightforwardly handle multiple phases. Given a partitioning of the image in multiple regions, we can compute normalized frequencies corresponding to the inside of each region. This leads to a set of probability images P_j , one for each region. A parametric curve will then be moved according to (15), but where we compare the probabilities for the two regions the curve is dividing.

In case of a geometric curve, we adopt a strategy of using one level-set for each phase. For an evolution without a vacuum or overlap, we use an additional transformation of probability images as described in [6]. This is applied for each phase, and transforms J probabilities (one for each phase) into two, one for the phase in question and another for all other phases.

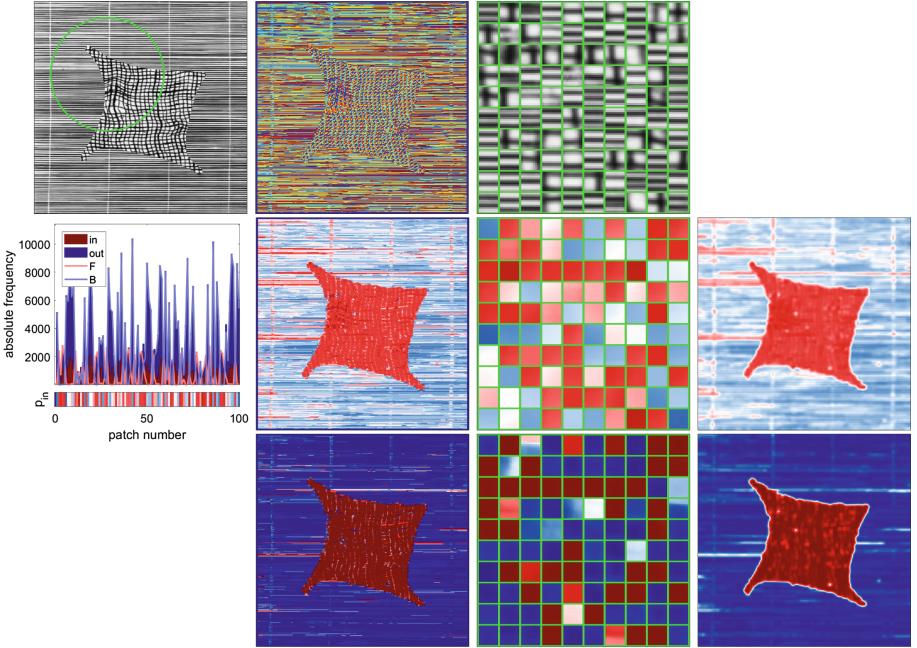


Fig. 2. Distribution of patches for a textured image. *First row* shows an input image (left), an assignment image (middle) and a dictionary containing 100 patches obtained from the image (right). The size of the image is 640-by-640 pixels, and patches are 11-by-11 pixels. *Second row* shows histograms for patch centers (left) and P_{in} using a central-pixel approach, for inside indicated by the green curve (middle left). The histogram appears ragged due to lack of obvious sorting of the clusters. The colored bar below the histogram indicates p_{in} for each patch center. Also in *second row*, dictionary probabilities (middle right) and P_{in} (right) using a patch-probabilities approach are shown. *Last row* shows similar results but for inside being the foreground region of the image. Note that probabilities indicate that some patches often appear on the boundary between foreground and background. (Color figure online)

3 Results and Discussion

In our current implementation the initialization is provided as a curve (usually a small circle) vaguely outlining a region in the image. It is our experience that even a weak initialization leads to a reasonable estimate of probability densities for different image regions – see a small test in Fig. 3.

The effect of the regularization parameter α is shown in Fig. 4. In the case of intensity-based segmentation, a strong regularization is needed for good results. This might have an undesirable effect where the boundary between foreground and background has a high curvature. With the patch-based model we can use a weaker regularization and obtain a more accurate segmentation result.

We have implemented our approach as both a parametric (snakes [8]) and geometric (level-sets [11]) deformable model. In Fig. 5 we compare the two

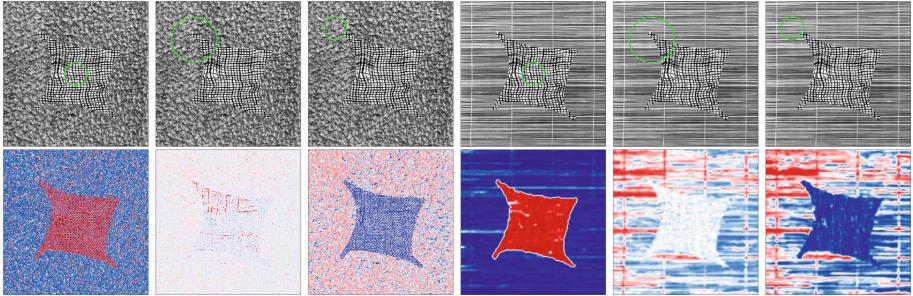


Fig. 3. Initialization analysis shown on two images from Figs. 1 and 2. *Top row* shows input images and initialization curves. *Bottom row* shows calculated P_{in} . First test image (columns 1–3) uses the intensity-based approach, while the second image (columns 4–6) uses the patch distributions.

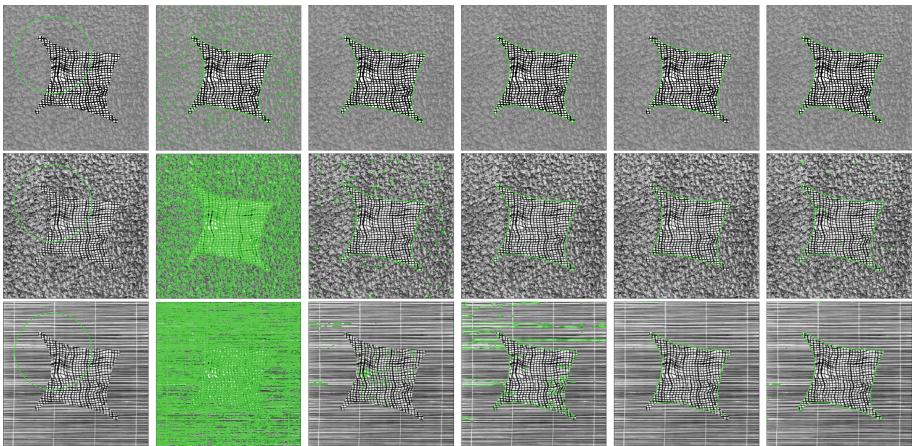


Fig. 4. Effect of regularization on results for images from Figs. 1 and 2. *First column* shows input image and curve initialization. *Columns 2–5* show converged results of intensity based segmentation with $\alpha = 0$, $\alpha = 1$, $\alpha = 2$, and $\alpha = 5$ respectively. *Right-most column* brings results of the patch-based approach, with dictionary containing 1554 image patches of size 7-by-7, and with $\alpha = 0$ for the first two rows and $\alpha = 1$ for the last.

implementations on a pair of natural images. When the foreground consists of a single object, the results obtained with the two curve representations are comparable, with level-sets being slightly more prone to noise. When an image contains multiple foreground regions, level-sets are capable of adapting the topology and capturing multiple foreground objects. The snakes approach is constrained by a single curve topology and does not capture the additional foreground regions. Furthermore, the presence of foreground regions outside the curve makes the probability estimation more blurry.

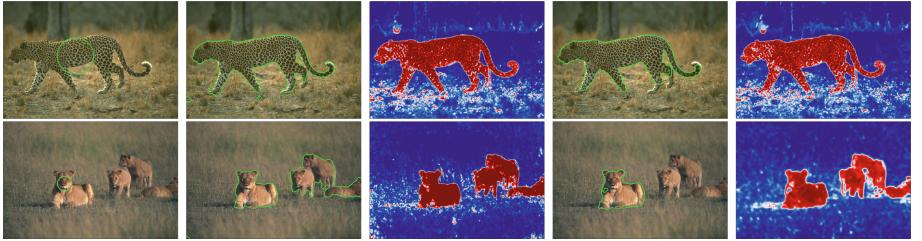


Fig. 5. Snakes and level-set segmentations of natural images. *First column:* initialization. *Columns 2–3:* results using level-sets and the final P_{in} . *Columns 4–5:* results using snakes and the final P_{in} . Patches of size 3-by-3 are used for the top image and 7-by-7 for the bottom image. Both dictionaries contain 1554 patches. Regularization parameters are $\alpha = 2$ for the level-set model and $\alpha = 1$ for the snakes model, but those cannot be directly compared due to differences in implementation.

Results of a single multi-phase segmentation are shown in Fig. 6.

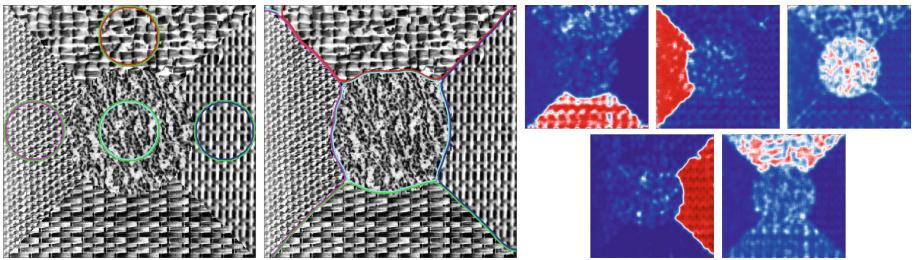


Fig. 6. Segmentation using a multiphase level-set method. Big images shows the initialization and the obtained result. Remaining images show the pixel-wise probabilities of the five regions. A dictionary contains 1554 9-by-9 patches and $\alpha = 2$.

4 Conclusion

We reformulate the established deformable models for image segmentation in a probabilistic framework. The similarity of our framework with the deformation model presented in (3) and (4) lies in obtaining region estimates by averaging, and then moving the curve to increase the distance between those estimates. While they estimate intensity means, we estimate intensity distributions. Averaging is also fundamental for the model in (1) and (2), and additional similarity to our model is that pixel move to the image region to which they fit better. While they use a difference between pixel intensity and an estimated mean as a measure of fit, we use probabilities. In both cases, moving pixels will improve the current estimate of the region property.

Just like those established models, our approach is simple, based on counting and averaging, and with an efficient implementation. Still, the model is very

powerful and provides good segmentation. We believe that our framework would be a useful addition to the family of deformable models.

References

1. Brox, T., Rousson, M., Deriche, R., Weickert, J.: Colour, texture, and motion in level set based segmentation and tracking. *Image Vis. Comput.* **28**(3), 376–390 (2010)
2. Brox, T., Weickert, J.: A TV flow based local scale estimate and its application to texture discrimination. *J. Vis. Commun. Image Represent.* **17**(5), 1053–1073 (2006)
3. Chan, T., Vese, L.: An active contour model without edges. In: Nielsen, M., Johansen, P., Olsen, O.F., Weickert, J. (eds.) *Scale-Space 1999*. LNCS, vol. 1682, pp. 141–151. Springer, Heidelberg (1999). doi:[10.1007/3-540-48236-9_13](https://doi.org/10.1007/3-540-48236-9_13)
4. Chan, T.F., Vese, L.A.: Active contours without edges. *IEEE Trans. Image Process.* **10**(2), 266–277 (2001)
5. Dahl, A.B., Dahl, V.A.: Dictionary snakes. In: International Conference on Pattern Recognition (ICPR), pp. 142–147. IEEE (2014)
6. Dahl, A.B., Dahl, V.A.: Dictionary based image segmentation. In: Paulsen, R.R., Pedersen, K.S. (eds.) *SCIA 2015*. LNCS, vol. 9127, pp. 26–37. Springer, Cham (2015). doi:[10.1007/978-3-319-19665-7_3](https://doi.org/10.1007/978-3-319-19665-7_3)
7. Gao, Y., Bouix, S., Shenton, M., Tannenbaum, A.: Sparse texture active contour. *IEEE Trans. Image Process.* **22**(10), 3866–3878 (2013)
8. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: active contour models. *Int. J. Comput. Vis.* **1**(4), 321–331 (1988)
9. Kim, J., Fisher, J.W., Yezzi, A., Çetin, M., Willsky, A.S.: A nonparametric statistical method for image segmentation using information theory and curve evolution. *IEEE Trans. Image Process.* **14**(10), 1486–1502 (2005)
10. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. *Commun. Pure Appl. Math.* **42**(5), 577–685 (1989)
11. Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *J. Comput. Phys.* **79**(1), 12–49 (1988)
12. Rousson, M., Brox, T., Deriche, R.: Active unsupervised texture segmentation on a diffusion based feature space. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. II-699. IEEE (2003)
13. Tsai, A., Yezzi Jr., A., Willsky, A.S.: Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification. *IEEE Trans. Image Process.* **10**(8), 1169–1186 (2001)
14. Wang, J., Chan, K.L.: Incorporating patch subspace model in Mumford-Shah type active contours. *IEEE Trans. Image Process.* **22**(11), 4473–4485 (2013)
15. Yezzi Jr., A., Tsai, A., Willsky, A.: A statistical approach to snakes for bimodal and trimodal imagery. In: International Conference on Computer Vision (ICCV), vol. 2, pp. 898–903. IEEE (1999)