

# Local Region Descriptors for Active Contours Evolution

Cristina Darolti, Alfred Mertins, *Senior Member, IEEE*, Christoph Bodensteiner, *Member, IEEE*, and Ulrich G. Hofmann

**Abstract**—Edge-based and region-based active contours are frequently used in image segmentation. While edges characterize small neighborhoods of pixels, region descriptors characterize entire image regions that may have overlapping probability densities. In this paper, we propose to characterize image regions locally by defining Local Region Descriptors (LRDs). These are essentially feature statistics from pixels located within windows centered on the evolving contour, and they may reduce the overlap between distributions. LRDs are used to define general-form energies based on level sets. In general, a particular energy is associated with an active contour by means of the logarithm of the probability density of features conditioned on the region. In order to reduce the number of local minima of such energies, we introduce two novel functions for constructing the energy functional which are both based on the assumption that local densities are approximately Gaussian. The first uses a similarity measure between features of pixels that involves confidence intervals. The second employs a local Markov Random Field (MRF) model. By minimizing the associated energies, we obtain active contours that can segment objects that have largely overlapping global probability densities. Our experiments show that the proposed method can accurately segment natural large images in very short time when using a fast level-set implementation.

**Index Terms**—Active contours, image segmentation, local statistics.

## I. INTRODUCTION

COMPUTER vision requires the automated segmentation of images for automatic object detection and object tracking. This goal is pursued in our work in a framework of variational methods for image and video segmentation known as active contours. In active contour methods, a contour is usually associated with an energy that is minimal when the contour coincides with the real boundary of the segmented object. Widely used energy types depend on the smoothness of the boundary curve and on image features. Image features may refer to the strengths of edges, yielding edge-based active contours, or to the characteristics of the regions occupied by objects in the processed image, yielding region-based active contours.

Manuscript received August 27, 2007; revised July 30, 2008. Current version published November 12, 2008. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Vicent Caselles.

C. Darolti, A. Mertins, and U. G. Hofmann are with the Institute for Signal Processing, University of Lübeck, 23538, Germany (e-mail: darolti@isip.uni-luebeck.de; mertins@isip.uni-luebeck.de; hofmann@isip.uni-luebeck.de).

C. Bodensteiner is with the Institute for Robotics, University of Lübeck, 23538, Germany (e-mail: bodensteiner@rob.uni-luebeck.de).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIP.2008.2006443

The segmentation process starts with an initial contour obtained automatically or with the help of user interaction; this contour is evolved toward object boundaries under the action of forces derived from the energy, and energy minimization is often accomplished by gradient descent.

The first active-contour methods were edge-based. They used functionals that depended on the response of the image to an edge filter [1]–[6] such that the magnitude of motion forces derived from these functionals is small when the strength of the edge is large. Segmentation with an edge-based active contour is affected by problems of edge detectors. Weak or undetected perceptual edges cause an active contour to pass over real boundaries, while undesired strong edges stop the contour. Consequently, a classical drawback of edge-based active contours is their small range of capture, which requires the initial contour to be placed in the close vicinity of objects to be segmented.

Intensity information gained from image regions delimited by a contour can be added to an edge-based energy functional in order to make active contours more robust [7], [8]; further results for energies based on a linear combination of edge and region terms can be found in [9] and [10]. Zhu and Yuille [11] assumed it to be sufficient to consider only the region information and designed a more general energy functional for the purpose of segmenting an image in  $N$  regions.

For region-based energies, it is crucial to capture the information that best distinguishes between different objects. The information about a region is named *region descriptor* (in similarity with [12] and [13]). An obvious region descriptor is one that characterizes the variation of color (intensity) within a region.

Recently, many other types of information have been included in the energy functional: a vector field, as the optical flow field [14], [15], motion detection [16], [17], texture filters [17]–[19], image description by vectors of features [20], measures of shape similarity [21], [22], or the geometry of the active contour [23].

Image features that describe a region usually vary within this region and the goal of a region descriptor is to formalize and measure the variation. This is commonly achieved by interpreting the varying values of a feature as realizations of a random variable with a probability density function (pdf) that needs to be determined. Some methods approximate region pdfs before the active contour segmentation is started: in [7], [9], and [24], image intensities are modeled as a Gaussian mixture and its parameters are learned beforehand with an expectation maximization (EM) algorithm; in [25], the user selects image samples, thus making it possible to compute a mean and a variance for each region. Other methods [11], [26]–[28] approximate the parameters of the pdf at each evolution step, and for each region.

Arbitrary densities of real world objects can be approximated by nonparametric kernel density estimators (KDEs) [20], [29]. With an appropriate choice of kernel window width, nonparametric KDEs can describe the data closely, but because of this, new data points not present in the learning set may have low probabilities. Two regions described nonparametrically may be compared in order to decide if they belong to the same object by computing a statistical divergence, like the Kullback–Leibler divergence or the Battacharya distance [30].

Generally, a region descriptor is estimated from all samples within the region delimited by the active contour (e.g., [17], [24], [26]), i.e., they are global region descriptors. Active contours based on global region descriptors are negatively affected when the support of distributions of different regions overlap. In this situation, values which fall on the tail of the distribution might be classified based on the maximum likelihood criterion into the wrong region. This is an omnipresent problem in classification tasks; however, an overview of possible solutions is beyond the scope of this paper. In our work, we use the idea that the overlap is likely to decrease if only samples from around the contour are the basis for density estimation.

In this paper, regions are described only locally, due to the fact that local processing may minimize the overlap of pdfs describing regions. For this purpose, the concept of local region descriptors (LRDs) computed from samples within windows centered on the active contour is introduced along with a formalism for constructing energy functionals from LRDs. These functionals have many local minima. This issue is addressed in our work by changing the function that takes LRD quantities as argument, such that the number of local minima is reduced. First, a balloon force is added to a region competition equation defined via the function “logarithm-of-pixel-probability” with pdf determined from a LRD. Second, we introduce two novel functions based on a Gaussian similarity measure and on local modeling of the image as a Markov Random Field (MRF). We name all these functions segmenter functions. LRD-based active contours can be used for object segmentation, for example, for interactive object selection for enhancement and photographic processing of digital images; the method could also be used for medical image segmentation. In our ongoing work, it is used to track the contours of surgeons’ hands in video streams recorded while they accomplish a simple suturing task.

In the remainder of the paper, a short review of region-based functionals and their properties is presented for completeness in Section II. Novel LRD-based active contours and segmenter functions are introduced and their properties explored in Section III. Their implementation is accomplished with the fast level-set method [31]; low computation times are important, especially for tracking applications. The proposed active contours are tested on various synthetic and real, gray, and color images by placing initial contours inside the objects to be segmented and letting them grow toward object boundaries. In experiments, objects with visual characteristics that create the impression of an approximately uniform appearance can be segmented correctly and rapidly, even when they are neighbored by objects similar in appearance. The paper concludes with a discussion and comments on future work.

## II. ACTIVE CONTOURS EVOLUTION BASED ON REGION DESCRIPTORS

Active contour methods have been studied intensely in recent years; we present a short overview of these methods as to allow for the placement of our method in the active contours framework.

The appearance of many objects in the world can be described by their color (or intensity), while their shapes are mostly composed of smooth surfaces rather than of randomly connected points. An image of an arrangement of such objects is often the union of homogeneous regions delineated by smooth boundaries. As early as 1985, Mumford and Shah have formulated this observation mathematically. In their work [32], a region in an image  $I$  may be approximated by a piecewise smooth function  $u(\mathbf{x}) = u(x, y)$  defined on the domain  $R \subset R^2$  and discontinuous at the set of boundary points  $C$ ; the boundary is considered to be parametrized by the arc length  $s$ ; the best-approximating piecewise smooth function is determined by minimizing the functional

$$E(u, C) = \int_R |u - I|^2 d\mathbf{x} + \mu \int_{R-C} |\nabla u|^2 d\mathbf{x} + \nu \int_C ds. \quad (1)$$

The parameters  $\mu$  and  $\nu$  weigh the contributions of smoothness of the approximating function and the smoothness of the boundary set, respectively. Textures, however, cannot be described by smooth functions; instead, their visual characteristics may be modeled as random variables. One may consider that the value of a feature, e.g., the image intensity  $I(\mathbf{x})$ , at an image pixel  $\mathbf{x} = (x, y)$  in a region  $R_i$  is drawn from a pdf  $p(I(x, y)|\theta_i) = p_i(\mathbf{x})$ , where  $\theta_i$  denotes the parameters of the probability density in region  $R_i$ . Assuming  $N$  regions and that the pixels in each region are independent random variables, Zhu and Yuille introduce a functional that depends on the probabilities of image pixels and the image’s division into regions

$$E(R_i, \theta_i, N) = \sum_{i=1}^N \left\{ \left( \frac{\mu}{2} \int_C ds \right) - \int_{R_i} \int \log p_i(\mathbf{x}) d\mathbf{x} + \beta \right\}. \quad (2)$$

In order to minimize this energy, an initial boundary  $C_{i,j}$  is evolved between two regions  $R_i$  and  $R_j$  and its motion is derived using Green’s theorem and the Euler–Lagrange equations [11]

$$\frac{\partial C_{i,j}}{\partial t} = -\mu k_i \vec{n}_i + (\log p_i(I(\mathbf{x})) - \log p_j(I(\mathbf{x}))) \vec{n}_i \quad (3)$$

with  $k_i$  the curvature of the curve  $C_{i,j}$  and  $\vec{n}_i$  its normal pointing in the direction of the region  $R_j$ . The first half of the second term may be interpreted as a force, with the magnitude depending on the region descriptors  $R_i$ , pushing the curve outward or inward.

Finding the minimum of the energy involves the unknown parameters  $\theta_i$ . They are determined according to the EM principle [33]. First, the regions  $R_i$  are considered to be fixed, and the parameters of their probability densities  $p_i$  are computed. Second, the previously computed probability densities are considered constant, time is introduced as an artificial parameter  $t$ , and the equation of motion is determined by computing the

gradient of the energy with respect to the contour's variation in time.

The algorithmic implementation of the above equation of motion may rely on parametric curves known as snakes [1], [11], [34], [35]. Handling snake points may be a daunting task, especially when more image objects are to be detected or when the topology of the contour changes. The level-set method introduced by Osher and Sethian [36] is an alternative that can deal with these issues easily. The level-set method has thus become a very popular method for numerically evolving curves in the direction of their normal. To this end, an evolving curve  $C(t)$ , can be embedded as the zero-level set of a function  $\Phi$

$$C(t) = \{(x, y) | \Phi(x, y, t) = 0\} \text{ with } \Phi(x, y, 0) = C_0 \quad (4)$$

where  $C_0$  is the initial curve.  $\Phi$  is very often a signed distance function taken to be positive on the inside and negative on the outside of the curve. The outward normal  $\vec{n}$  of  $C$  and its curvature  $k$  can be expressed with the gradient  $\nabla\Phi$  of the new function. An equation of motion of the form  $C_t = k\vec{n}$  can be rewritten as  $\Phi_t = \text{div}(\nabla\Phi/|\nabla\Phi|)|\nabla\Phi|$ , with  $\Phi_t$  the time derivative of function  $\Phi$ .

Level sets were introduced into the variational approach to image segmentation with edge-based functionals [5], [6], [37], where  $\Phi$  is the signed distance transform of the active contour. For region-based contours, a contour's energy and motion were formulated with the help of level sets in [26] and [27]

$$E(\Phi, p_1, p_2) = - \int \int_R H(\Phi) \log p_1 + (1 - H(\Phi)) \log p_2 - \mu |\nabla H(\phi)| dx dy \quad (5)$$

$$\frac{\partial\Phi}{\partial t} = H'(\Phi) (\log p_1 - \log p_2 + \mu \left( \text{div} \left( \frac{\nabla\Phi}{|\nabla\Phi|} \right) \right)). \quad (6)$$

The domain of the entire image is denoted here with  $R$ . The energy equation obtained with this formulation is equivalent to (2) for the special case of two regions. The function  $H$  is the Heaviside function

$$H(x) = \begin{cases} 1, & x \geq 0, \\ 0 & x < 0 \end{cases} \quad (7)$$

Since this function is discontinuous, its derivative  $H'(x)$  needs to be defined; commonly, it is defined as the Dirac delta distribution:  $H'(x) = \delta(x)$ . Alternatively, to avoid introducing an infinite term at  $\Phi(x, y) = 0$ , one can use a regularized versions  $H_\epsilon$  of the Heaviside function and obtain its derivative  $H'_\epsilon = \delta_\epsilon$ , similar to [26], with a small  $\epsilon > 0$ . The Dirac delta is then obtained for  $\epsilon \rightarrow 0$ .

Recently, some authors have been concerned with level-set based functionals that must be optimized over multiple regions, i.e., when more than two objects are to be segmented in an image [12], [13], [24], [29], [38], and [39]. If each of  $N$  regions is represented by its own level set  $\Phi_i$ , the functional in (5) and its

corresponding equation of motion can be written by analogy as follows [29]:

$$E(\Phi_i, p_i, N) = \sum_{i=1}^N \left( - \int \int_R H(\Phi_i) \log p_i - \frac{\mu}{2} |\nabla H(\Phi_i)| dx \right) \quad (8)$$

$$\frac{\partial\Phi_i}{\partial t} = H'(\Phi_i) \times \left( \log p_i - \max_{j \neq i, H(\Phi_j) > 0} \log p_j + \mu k_i \right) \quad (9)$$

where we have denoted  $k_i = (\text{div}(\nabla\Phi_i/|\nabla\Phi_i|))$ . In (9), the statistics force generated by a region  $R_i$  at a contour point competes with the strongest of the statistic forces generated by all other regions neighboring the point.

In [12] and [13], shape derivatives are introduced. With shape derivatives, region descriptors are variables that depend on  $\Phi_i$  and thus introduce additional terms to the evolution equation. These terms may improve the segmentation, increasing at the same time the complexity of the implementation and its computational cost. We prefer a fast method and rely on LRDs for good segmentation results, and so growing contours draw upon (8) and (9), as will be seen later in the next section.

### III. LOCAL REGION DESCRIPTORS FOR ACTIVE CONTOURS

Probability densities that strongly overlap usually lead to poor segmentation results. The extreme example of an image with two regions having normal distributions with the same mean but different variances, as shown in Fig. 1(a), has been discussed in [11]. The authors correct the evolution equation by analyzing two sets of parameters in order to solve the problem. One set of parameters is computed for each region from all pixels in this region. The other set is computed from the pixels within a window  $W(x, y)$  centered on each pixel on the evolving curve. The probability of a boundary pixel is then replaced with the probability of the window considered to have  $m$  independent pixels in the energy functional (2). The motion force then includes a term that compares the mean and variance over samples in the window with the global region mean and variance in a statistic-test-like manner.

Global region descriptors decouple pixel intensities from their spatial positions. Visually, more elements form a group not only if they look alike, but also if they are in close proximity of each other, whereas similar elements which are further away will not belong to the group. For a very large number of objects, their visual properties change in a relatively slow manner; a comparatively sudden change in color and texture is very often accounted for by the presence of a boundary between objects. For our method, we assume that this situation exists in the image to be segmented.

Global descriptors may hamper an active contour. In the example in Fig. 1(a), regions are well characterized by their respective variances, but this need not always be the case when their distributions overlap. Consider the synthetic image in Fig. 2(a).

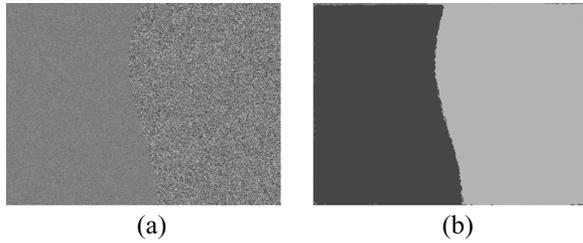


Fig. 1. Example of segmentation of a synthetic image. (a) Synthetic image showing two regions with mean 128 and variances 10 and 35 and (b) its segmentation by (12) (using the segmenter function  $g_{\text{mrf}}$  of LRDs). The segmentation is initialization invariant, as long as an initial contours is placed inside each region.

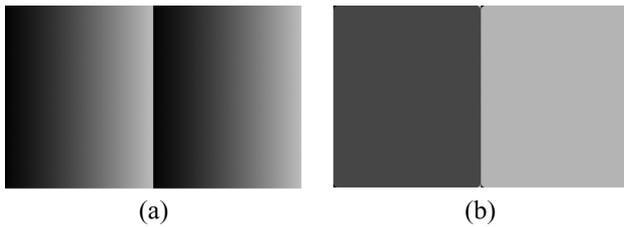


Fig. 2. Example of segmentation of a synthetic image. (a) Synthetic image showing two regions with the same discrete uniform distributions and (b) segmentation obtained with LRDs plus balloon force according to (12). The segmentation is initialization invariant, as long as an initial contours is placed inside each region.

It shows two rectangles filled with the same color gradient that clearly are two separate regions. The intensities in the two rectangles not only overlap, they even follow the same discrete uniform distribution. However, we see two rectangles because the intensity varies slowly within a rectangle and the sudden change responsible for boundary perception occurs between rectangles.

Describing each region only locally is more appropriate for this image. In the following, we show how both synthetic and natural images can be correctly segmented by minimizing energies given by different segmenter functions that depend on LRDs only. First, LRDs are defined, then the concept of segmenter functions is introduced along with different segmenter functions.

### A. Local Region Descriptors

A region descriptor is defined by choosing or computing features that quantify the visual characteristics of a region and by choosing a probability model to express the variation of those features within the region. Features may be chosen from intensity, color and quantities that measure texture properties, e.g., values obtained with Gabor filters. Probability distributions on features may be modeled parametrically or nonparametrically.

*Local region descriptors* are region descriptors with probability distributions computed from feature samples *within regions that lie inside windows centered on an active contour*. At each pixel on the contour, statistics that describe a region are computed only from samples in this region. The LRD is defined by specifying the size of the window, the features that describe a region and the type of pdf underlying the feature samples in the window. Features may be computed by either taking into account the position of the contour or not. Choosing the window's

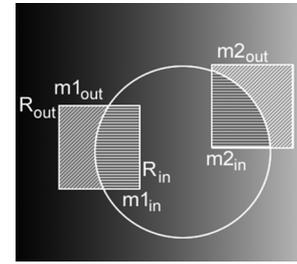


Fig. 3. Initial contour separating the image in an inside region  $R_{\text{in}}$  and an outside region  $R_{\text{out}}$ , and two windows  $W_1(x, y)$  and  $W_2(x, y)$  centered on contour pixels. Region patches included in each window are highlighted. Each includes the pixels for computing the LRDs  $m_{1,\text{in}}$ ,  $m_{1,\text{out}}$ ,  $m_{2,\text{in}}$ , and  $m_{2,\text{out}}$ .

shape is also part of an LRD's definition. Most obvious choices for the shape of a window with the purpose of describing the image locally are squares (as in [11] and [40]) or circles (as in [11] and [41]).

Two square-shaped windows used for computing LRD values are depicted in Fig. 3. The initial curve of circular shape divides the image into two regions:  $R_{\text{in}}$  and  $R_{\text{out}}$ . Each window includes patches from both regions. We observe that a LRD value describing  $R_{\text{in}}$ , for example the mean pixel intensity, can vary significantly along the contour; the same is true for an LRD describing  $R_{\text{out}}$ . For example, the inside mean  $m_{1,\text{in}}$  in the window on the left of the image is closer in value to the outside mean  $m_{1,\text{out}}$  in the same window than to inside mean  $m_{2,\text{in}}$  in the window on the right.

We were inspired by Pappas' adaptive clustering algorithm [40] in our decision to determine the statistics of each region separately. That algorithm estimates the local pdf of a class of pixels only from pixels that already belong to this class with very promising results.

Samples within a window are assumed to follow a Gaussian distribution in [11], but the position of the contour is not taken into account when computing statistics. Unlike in [11], pdfs for LRDs *are computed for each region in the window separately*; another difference to [11] is that here global descriptors are not computed. The position of the true boundary is considered uncertain to a degree depending on the window size [11]; in other words, processing within local windows over real boundaries blurs those boundaries. Windows too small do not include enough samples to reliably compute statistic forces. Windows too large are associated with large uncertainty about real boundary positioning. With LRDs the boundary is generally accurate *when it is found*. The window size influences the result minimally, as long as it contains enough samples from each region and the assumptions about the local pdfs are true; this can be seen in the results of experiments with different window sizes (Figs. 8 and 12). These results are due to the fact that, as the contour approaches the boundary, only samples from one image regions are used to compute the statistics for this region. In [34], the possibility of separately describing patches of regions is theoretically mentioned, but experimentally tested is a method that compares all samples in a window centered on a snake point with region patches outside and inside the point.

A method that involves local region processing and takes into account the contour's position has been presented very recently

in [42]; an independent development, the method is based on a special type of LRD. The feature used to describe the region is the local weighted average of the intensity inside the region. This feature is obtained at every pixel by convolving the image with a Gaussian kernel that is strictly positive only inside the region containing the pixel. The window  $W$  is reduced to a pixel and there is no probability distribution model involved.

In [43], it has been demonstrated that the Mumford–Shah energy based on piecewise smooth approximations of an image is equivalent to a likelihood-based energy, like the one in (8), formulated with LRDs. With LRD, the authors refer to is the intensity feature with normal pdf with constant  $\sigma = \sqrt{0.5}$ , and a given window size  $W$ . This is an alternative interpretation of the LRD in [42]. Another LRD for which the feature is pixel intensity is used in [41] to formulate a hybrid region-edge-based active contour; the local statistics computed for this feature are the local region sample mean and variance. For the images segmented in this paper, we choose simple region descriptors that rely on intensity or color and we assume the distributions of these features are locally Gaussian; intensity is modeled with univariate normal distributions and color with multivariate ones. We make this choices because we want to emphasize the advantages that come from local modeling.

#### IV. SEGMENTER FUNCTIONS

The majority of energies associated with active contours have either a form similar to (1) or to (8); (1) expresses the assumption that the image can be approximated with piecewise smooth functions, while for (8) it is assumed that each pixel is assigned a region label and the energy represents the logarithm of the joint pdfs of pixel features conditioned on their region label. Both energy types thus include a double integral over the image domain of functions of values depending on pixel features. This suggests that we can introduce general-form energies based on LRDs by allowing a flexible form for the function under the double integral.

Formally, consider a LRD composed of a feature vector  $f(x, y) = (f_{ilt1}, \dots, f_{iltM})$  and a pdf model with a different set of parameters  $\theta^{W(x,y) \cap R_i}$ , shortly  $\theta_i^W$ , in each window  $W(x, y)$  centered on a contour pixel  $(x, y)$ . Also consider a real-valued function  $g$ . Each region  $R_i$  is represented by a level-set function  $\Phi_i$ . We propose the minimization of the following energy functional, considering the number  $N$  of regions known

$$E = \sum_{i=1}^N - \int \int_R \left( H(\Phi_i) g(f_i(x, y), \theta_i^W) - \frac{\mu}{2} |\nabla H(\Phi_i)| \right) dx dy. \quad (10)$$

The equation of motion for the level set function  $\Phi_i$  can be derived from the associated Euler–Lagrange equation (as shown in the Appendix)

$$\frac{\partial \Phi_i}{\partial t} = H'(\Phi_i) \left( g(f_i, \theta_i^W) - g(f_j, \theta_j^W) + \mu k_i \right). \quad (11)$$

We denote function  $g$  as segmenter function.

Note that the Mumford–Shah functional in (1) cannot be expressed, in general, as a particular case of this energy, since the

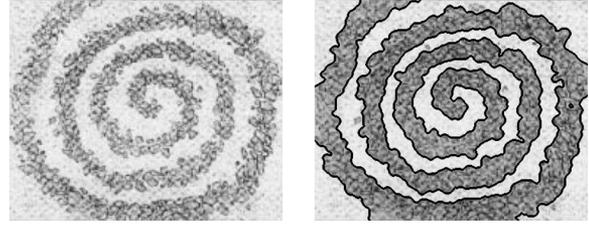


Fig. 4. Spiral image presented in [27] (left) segmented with LRDs and segmenter function  $g = \log p_{\mathcal{N}}$  (right).

function smoothness term  $\int_{R-C} |\nabla u| dx$  is missing; an exception is presented in [43] (discussion will follow shortly). To obtain an energy similar to the energy in (8), we have to set the segmenter function to  $g = \log p$ . The difference is that the parameters of  $p$  are not computed from all values in a region  $R_i$ , but as LRDs.

Very often, the result of applying a filter  $f_i(x, y)$  to an image region is considered to be normally distributed, and, thus,  $g$  is the logarithm of a Gaussian pdf of a random vector (with more components when more filters are employed), i.e.,  $g = \log p_{\mathcal{N}}$  with  $p_{\mathcal{N}} \sim \mathcal{N}(\mathbf{m}_i^W, \Sigma_i^W)$  and where  $\mathbf{m}_i^W$  is the mean and  $\Sigma_i^W$  is the covariance matrix. The image of a spiral (the one presented in [27]) is initialized with a grid of circles. It is segmented by a contour moving according to (11) with the segmenter function  $g = \log p_{\mathcal{N}}$ . The LRD is composed of the feature pixel-intensity for which the probability model is a local normal distribution; the result is shown in Fig. 4.

As in [11], contours are best initialized within real objects. If a real boundary cuts through an initial patch, it still may be possible to obtain a correct segmentation by evolving the contour according to the region competition equation (11); it is necessary for this purpose that part of the object to be segmented occupies a larger area in the initial patches. This situation occurs for the spiral in Fig. 4.

Energies based on LRDs are prone to have more local minima than those based on global descriptors (depending on the degree of separability between descriptors), especially when an initial contour lies within a real object. Since LRDs characterize the image locally, they are short sighted. They tend to be equal on both sides of a contour making the motion force approximately zero. Not being able to use information from image parts outside the windows, the contour cannot escape a local minimum.

One alternative is to rely on the curvature term to evolve the contour when it lies in a homogeneous region, as in [42]. This alternative requires balancing the influence of the data term and the curvature term in the energy functional: the less homogeneous the regions, the larger  $\mu$  needs to be; however, this is known to be an unreliable solution. Because of the large number of local minima, the method in [42] is very sensitive to initialization and so is the method in [41]. For both methods, local image data must have unimodal distributions that are well characterized by the local mean. The variance of the data is not taken into consideration, but it can often be an important source of information (e.g., in the image in Fig. 1). Such probability models are often not suitable for descriptors based on more than one feature.

In the previous section, we have noted that special cases of LRDs are used to construct the energies in [41] and [42]; we may now show the segmenter functions used there. We have discussed in the previous section that the LRD in [42] can be interpreted in two different ways. If the interpretation is that the LRD relies on the feature intensity, the segmenter function that needs to be used is  $g = \log p_{\mathcal{N}}$  with  $p_{\mathcal{N}} \sim \mathcal{N}(m_i^W, \sqrt{0.5})$ . If the LRD is considered to be based on the local weighted mean intensity inside a region, denoted here by  $f(\mathbf{x})$ , the energy proposed by the authors is obtained with the segmenter function  $g = (I(\mathbf{x}) - f(\mathbf{x}))^2$ ;  $I(\mathbf{x})$  denotes image intensity at pixel  $\mathbf{x}$ . In this case,  $g$  resembles the term under the integral in the Mumford–Shah functional from (1). The segmenter function in [41] also resembles this term, since  $g = (I(\mathbf{x}) - m_i^W(\mathbf{x}))^2$ , where  $m_i^W(\mathbf{x})$  is computed as the local mean intensity inside region  $R_i$ . We note here that the function in [34] may also be seen as segmenter function. It is based on the Ward distance and its role is to decide whether a small rectangular patch around a snake point should be merged with the foreground or the background.

Another known solution to the local minima problem involves adding a small constant force term  $\lambda$  to the motion of the level set  $\Phi_i$

$$\frac{\partial \Phi_i}{\partial t} = H'(\Phi_i) (g(f_i, \theta_i^W) - g(f_j, \theta_j^W) + \lambda + \mu k_i). \quad (12)$$

The parameter  $\lambda$  can be seen as a maximal area constraint [26] or plays the role of a balloon force [3]. In the context of LRD-based contour evolution, it can be interpreted as follows:  $\lambda$  indirectly gives a measure of the minimal difference that needs to exist between two local image regions in order to consider that they describe the sides of a real image boundary. As such, it may reflect the *a priori* knowledge about the smallest local image variations created by real edges. Equation (12) describes a *local* region competition equation with an added balloon force that yields a method for segmenting images with strongly overlapping distributions for foreground and background objects, as will be discussed in Section VII and exemplified in Fig. 5.

For the moment, we turn to the synthetic example in Fig. 2(a): the segmentation in Fig. 2(b) is obtained according to (12). The feature for the LRD is pixel intensity, which is assumed to be normally distributed in local regions, and  $g = \log p_{\mathcal{N}}$ ; this particular form of (12) will be referred to as LRDs plus balloon force. A circular initial contour is placed inside each region. While the computed means and variances for the outside and inside of the contour in each window are minimally different, balloon forces drive each contour to grow. When the contour reaches the bright-to-dark boundary, the local regions are different enough for the contour to stop. We may observe that at the boundary, distributions of LRDs do not overlap. Since intensity in a region varies linearly, the initial contours may be placed anywhere inside the respective regions. However, if global descriptors are computed, the result of the segmentation will depend on the initialization: for example, if the initial circles are centered within each region and include no dark pixels, the dark pixels will remain in the background since their initial probability is larger in the background, a fact that does not change when region histograms are reestimated at each iteration.

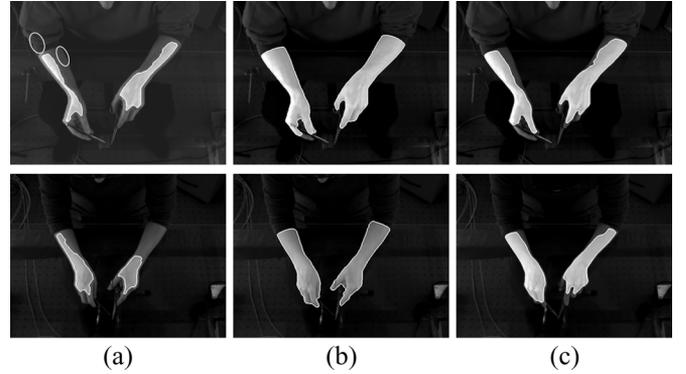


Fig. 5. Segmentation of two images of hands. (a) Initial contours. In the top image two regions with largely overlapping pdfs are marked with ellipses. (b) Segmentation with LRDS and the constant speed  $\lambda$ . (c) Segmentation with the method described in [11].

## V. NOVEL SEGMENTER FUNCTIONS

The search for other solutions to avoid local minima led us to two novel segmenter functions as will be shown in the following.

Consider an initial contour included in an image object. Intuitively, it should grow as long as pixels just outside the contour look similar to pixels just inside the contour; then, instead of describing both sides of the contour for region competition, only the objects need to be described, while a similarity test checks whether outside pixels match the description. In order to describe the intuition formally, we assume that the LRD has a normal distribution and we define the segmenter function  $g$  to be the confidence that a sample was drawn from this distribution. For a 1-D feature we know the percentage  $\alpha$  of values that fall within an interval of length  $c_\alpha$  around the mean  $m^W$ , measured in standard deviations  $s^W$ . Chosen a confidence level  $\alpha$ , the function  $g$  will be written

$$g_{\text{sim}}(c_\alpha, f_1(\mathbf{x}), \theta^W) = c_\alpha \cdot s^W - |f_1(\mathbf{x}) - m^W| \quad (13)$$

where  $|\cdot|$  denotes the absolute value; we remind here that  $\theta^W = (m^W, s^W)$  and  $W = W(x, y) \cap R_{\text{in}}$ . The smaller the difference between the feature value and the window mean the higher is the confidence that the pixel at the center of the window belongs to the inside of the contour. For  $\alpha = 0.68$ , the values accepted as similar fall within one standard deviation from the mean ( $c_\alpha = 1$ ). This means that if the difference is larger than one standard deviation, the pixel is considered to belong to the outside.

If more features are observed,  $g$  can be formulated using the value of the  $\chi_d^2$  distribution, with  $d$  degrees of freedom. For a chosen confidence  $1 - \alpha$ , values similar to the curve's inside will fall within a given interval of size  $\chi_d^2(\alpha)$  around the mean. Let  $f(\mathbf{x}) \in R^d$  be the feature vector,  $\mathbf{m}^W$  its mean, and  $\Sigma^W$  its covariance matrix.  $g_{\text{sim}}$  is then written as

$$g_{\text{sim}}(\alpha, f, \theta^W) = \chi_p^2(\alpha) - (f - \mathbf{m}^W)'(\Sigma^W)^{-1}(f - \mathbf{m}^W) \quad (14)$$

and it can be interpreted similarly to the 1-D case. Only LRDs which describe the inside of the object are computed, and letting all the variables describe the object only, the energy to be minimized is changed to

$$E(\Phi, p) = - \int \int_R \left( H(\Phi) g_{\text{sim}}(f(\mathbf{x}), \theta^W) - \frac{\mu}{2} |\nabla H(\Phi)| \right) d\mathbf{x}. \quad (15)$$

The energy is minimal when  $g_{\text{sim}}(\mathbf{x}) > 0$  for each  $\mathbf{x}$  with  $H(\Phi(\mathbf{x})) = 1$ , i.e., each pixel in the segmented object must be similar to its neighbors in the object, with similarity measured by  $g_{\text{sim}}$ . The associated Euler–Lagrange equation for the level-set function  $\Phi$  can be derived (see the Appendix), leading to the following equation:

$$\frac{\partial \Phi}{\partial t} = H'(\Phi) \left( g_{\text{sim}}(f, \theta^W) + \frac{\mu}{2} k \right). \quad (16)$$

This equation can be interpreted as follows: the level set will move to include a contour pixel  $\mathbf{x}$  if  $g_{\text{sim}}(\mathbf{x}) > 0$  and the curve stays smooth; it will do the opposite if  $g_{\text{sim}} < 0$ . Since only the object is described, the background can be complex and have any sort of empirical distribution as long as it does not match the local description of the object to be segmented. It is thus still possible to segment the object with  $g_{\text{sim}}$  even if object edges are weak (e.g., Figs. 7 and 9). This is not possible with local region competition plus balloon force that needs minimally strong edges that are stronger than edges inside the object to be segmented. Also, this method has difficulties when one of the objects separated by the boundary has a local probability distribution with multiple modes [see the incorrect segmentation of the camera in Fig. 10(c)].

The similarity-based function  $g_{\text{sim}}$  has difficulties to segment objects that are very noisy (e.g., Figs. 1 and 15) or when a small percentage of the pixels in a local window does not match the local Gaussian description (e.g., the background of Fig. 7 is slightly textured and there are some highlights in the arm and racket in Fig. 14). Methods that deal with this type of problems are often based on Markov Random Fields (MRF) and we will now show how a more robust segmenter function can be obtained by using MRF modeling.

A MRF models the intuitive idea that the intensity of a pixel depends on the intensities of neighbored pixels. Let the current segmentation of image  $I$  be  $C$ , i.e.,  $C$  is the characteristic function of the current local segmentation. If this segmentation is considered to be the realization of a MRF, the pdf for a pixel's segmentation label can be approximated by a Gibbs distribution as follows (for an introduction on MRF, see, for example, [40])

$$P(C(\mathbf{x}) = c) = \frac{1}{Z} \exp \left( -\frac{1}{T} \sum_{cl \in Cl(\mathbf{x})} (V_{cl}(c)) \right) \quad (17)$$

with  $Cl(\mathbf{x})$  the set of pairwise cliques that include pixel  $\mathbf{x}$ , and  $V_{cl}(c)$  the potential of a clique  $cl$  of two 8-connected pixels  $\mathbf{x}$  and  $\mathbf{y}$

$$V_{cl}(c) = \begin{cases} -\beta, & \text{if } c(\mathbf{x}) = c(\mathbf{y}) \text{ and } \mathbf{x}, \mathbf{y} \in Cl(\mathbf{x}) \\ \beta, & \text{if } c(\mathbf{x}) \neq c(\mathbf{y}) \text{ and } \mathbf{x}, \mathbf{y} \in Cl(\mathbf{x}). \end{cases} \quad (18)$$

$T$  is a parameter considered here constant. The original image is assumed to be a noisy version of the segmented image. The goal is to maximize the conditional pdf  $p(I(x, y) | C = c)$  assuming that it is normally distributed with mean  $m_i^W$  and standard deviation  $s_i^W$ . This is equivalent to minimizing the cost

$$\text{Cost}(\mathbf{x} \in R_i) = \frac{(I(\mathbf{x}) - m_i^W)^2}{2(s_i^W)^2} + \frac{1}{T} \sum_{cl \in Cl(\mathbf{x})} V_{cl}(c). \quad (19)$$

It may be observed that minimizing (19) is equivalent to maximizing (13) if the potential term is ignored. One can conclude

that it is possible to write a regularized version of (13) by taking into account the potential term

$$g_{\text{mrf}}(c_\alpha, f_1, \theta^W, C) = c_\alpha \cdot s^W - |f_1(\mathbf{x}) - m^W| - \nu \sum_{cl \in Cl(\mathbf{x})} V_{cl}(c). \quad (20)$$

Parameter  $\nu$  controls the influence of the regularization term. The purpose of this regularization is to ensure a smooth segmentation; from this point of view, it may be seen as a binarized version of the smoothness constraint in the Mumford–Shah functional in (1).

The MRF cost function  $g_{\text{mrf}}$  can replace  $g_{\text{sim}}$  in the energy of a growing curve (15) and in the corresponding equation of motion (16). It has been employed to segment the synthetic example in Fig. 1(a). The image was first filtered to replace each pixel's intensity with the value of the standard deviation of pixels in the 8-neighborhood; subsequently a Gaussian filter was applied and the resulting image was segmented employing the segmenter function  $g_{\text{mrf}}$ . The result is shown in Fig. 1(b). An initial circular contour was placed inside each image region and the two contours evolved independent of one another; one can thus observe for each region a stable segmentation result, independent of the other region.

Knowledge about the image can influence not only the choice of the pdf model for computing LRDs, but the choice for the segmenter function  $g$  as well. A more detailed discussion follows in Section VII, after mentioning one possible implementation for the method presented in this paper.

## VI. LOCAL REGION DESCRIPTORS FOR ACTIVE CONTOURS AND THE FAST LEVEL-SET IMPLEMENTATION

The level-set method is a very powerful tool that offers an implicit representation of image regions and is able to deal automatically with topology changes. However, numerical schemes for implementing level sets are computationally expensive; a fair amount of research has been done to improve upon the issue, for example, in [16] and [44].

The fast level-set method [31] has been shown to be two orders of magnitude faster than a sparse field numeric algorithm. It reduces the segmentation problem to classifying pixels to either the inside or outside of the zero level set representing a contour. The evolution algorithm can thus be greatly simplified [31], [45]. The level-set function is integer valued. At all pixels inside the curve the value is 3, except at border pixels where the value is 1. At all pixels outside the curve the value is  $-3$ , except at border pixels where the value is  $-1$ . Boundary smoothness is incorporated into the implementation by applying a Gaussian filter to the level-set function; the amount of smoothness is controlled by filter size and the number of filtering operations.

Speed forces are computed only at pixels on the inner and outer borders, and the level-set function is accordingly updated to include or exclude the pixel in question. The speed of the contour is always 1 pixel per time unit in this implementation. Thus, to evolve the contour at a pixel, we only need to compute the sign of the segmenter function; in fact, with this implementation, it suffices to have a segmenter function with codomain  $\{-1, 0, 1\}$ . The advantage is that many computations can be approximated by integer computations and speed is increased. For example, for  $g_{\text{mrf}}$  computations are reduced to comparing the

number of cliques assigned a positive  $\beta$  when the pixel belongs to the contour's inside with a threshold. Even with such a strong approximation, segmentation results are very good as will be seen in the next section.

In order to reduce the number of iterations needed for convergence, several initial patches can be placed inside one object, since level set methods can deal with topology changes automatically. Finally, LRDs can be computed every few steps during curve evolution, instead of every step; their values then determine motion forces not only for the center of the window, but also for pixels in its immediate neighborhood. The algorithm may be summarized as follows.

- Initialize contour  $C$ , level set  $\Phi$ , and select  $g$  and  $W$
- For each pixel  $\mathbf{x}$  on  $C$ :
  - compute  $\theta_i^W$ , the LRD for each neighboring region  $i$ ;
  - for each  $\mathbf{y} \in N_s(\mathbf{x})$ , compute the sign of  $g(\mathbf{y}, \theta_i^W)$ .
- Evolve  $C$  according to steps 2 and 3 of Table 1 in [31].
- Stop if convergence or maximum number of iterations reached.

In these expressions,  $N_s$  is a neighborhood of range  $s$ ; when  $s = 0$  the computation is carried out only for the contour pixel. With this algorithm, short times for the segmentation of large images have been observed, as will be shown in the following section.

## VII. RESULTS AND DISCUSSION

In this section, we present results for the proposed method and discuss the effects of different choices for parameters and window sizes. The method was implemented in C++ and all computations have been reduced to integer arithmetic. Contour initialization is started by the user, by selecting a few pixels in each object to be segmented and a circular contour is generated around each selected pixel.

### A. Experiment Settings

In our experiments, natural gray-level and color images were segmented. Features for LRDs are pixel intensity or color unless otherwise specified and their local pdfs are assumed to be Gaussian. We have experimented with LRDs,  $g = \log p_N$  and balloon forces according to (12), and with the two novel segmenter functions. The properties of evolution based on LRDs plus balloon forces are discussed together with the influence of the magnitude of the balloon force in Section VII-C. For the novel segmenter functions,  $c_\alpha$  was set to 1 for all images. Experimentally it was found that a smaller value of 0.75 will often create holes in the segmentation, and a larger value of 1.5 or 2 will cause leakage, especially for regions with large variances. Both  $g_{\text{sim}}$  and  $g_{\text{mrf}}$  use only information about the object. Due to this fact, the boundary between background and silhouette in Fig. 7 can be segmented although the hair and shirt contain the intensities present in the background because these regions do not fit the local normal distribution for the background close to the boundary. Starting from the silhouette, with an initialization that is just-as-far away from the boundary, we could not achieve this segmentation because there are too many edges present.

With a few exceptions, boundary smoothness was implemented by filtering the level-set function with a Gaussian filter

of  $5 \times 5$  pixels (standard deviation of 1.5), because a larger filter can better prevent leakage; it also prevents the formation of small holes in the segmentation (e.g., of noisy images). However, a larger filter does not allow for an accurate segmentation of corners and for such regions it is better to employ a smaller filter. For example, due to sharp corners in the woman's shirt and the table, a  $3 \times 3$  pixels filter has been employed to segment images in Figs. 14 and Fig. 15.

The size of the window  $W$  and the magnitude of the balloon force were varied in the experiments; segmentations change gradually as these parameters are varied; in some situations, these changes are minimal. The parameter  $\beta$  introduced to compute  $g_{\text{mrf}}$  has also been varied. It was found that it has a much larger influence on the segmentation result, since it is very coarse.

### B. Comparison With the Method From [11]

LRDs plus balloon force (with window side length of 11 px and  $\lambda = 5$ ) were used to segment and track the hands shown in Fig. 5; the difficulty of these images lies in the overlapping support of the histograms for the hands and the background (intensities between 50 and 75). For illustration purposes, two such regions are marked with ellipses in Fig. 5. Edge detectors, like Canny and Sobel yield edge maps that lead to leakage or false boundaries. The lightest patches in the hands, as shown in Fig. 5, are detected with an adaptive-threshold segmentation of the background-subtracted image. The binary image is repeatedly eroded to ensure that the initial contour is well inside the hands; using connected components, only the largest two patches from the eroded image are kept and used to initialize the contour. Subsequent frames are initialized by combining the information from the background-subtracted current frame and optical flow computed for pixels on the final contour in the previous frame.

The images of hands in Fig. 5 have also been segmented with a fast level set implementation of the method introduced in [11], starting with the same initial contours. Since neighbored regions on the arms and sleeves are not well characterized by their variances, the method [11] cannot achieve an accurate segmentation of the hands, as shown in Fig. 5(c). Dark pixels are mostly in the background and such pixels on the hand also get assigned to the background.

With the method in [11], further segmentation problems created by overlapping global distributions can be seen in Fig. 6. For the image in Fig. 6(e), the initial contour grows to include all dark objects, instead of stopping at the silhouette. All buildings are included in the background of Fig. 6(f). One can observe that many contours in segmentations in Fig. 6(g) and (h) are correct; however, the segmentation has leaked in Fig. 6(g), and it stops before reaching the boundary in Fig. 6(h). In this last image, pixels in the face region distort the PDF of part of the background region. Combined with the intensity gradient in the background itself and the noise in the image, this creates the conditions for the contour to stop prematurely. These problems are not present in the segmentations obtained with the method presented in this paper, as can be seen in Fig. 6(i)–(l).

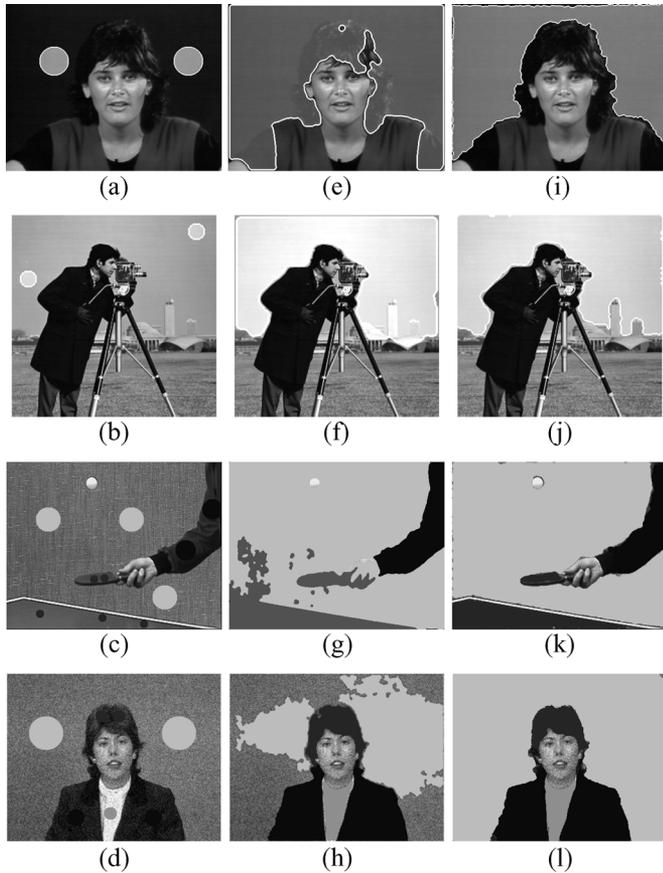


Fig. 6. Comparison of two segmentation methods. (a)–(d) Initial contours for both segmentation methods. (e)–(h) Segmentations obtained with the method in [11] (superimposed on initial images). (i)–(l) Segmentations obtained with the proposed method (superimposed on initial images).

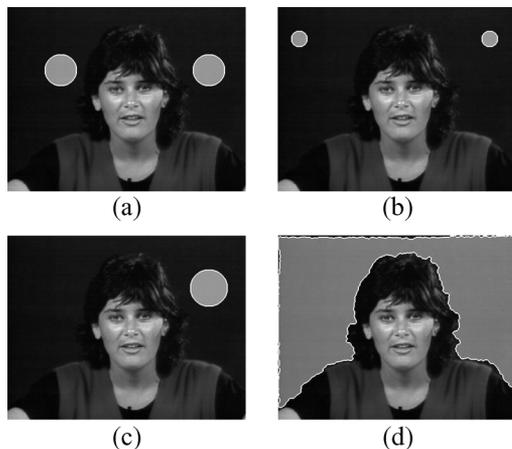


Fig. 7. Segmentations obtained with different initializations. (a)–(c) Different positions of initial contours for a popular image. (d) Segmentation with the function  $g_{\text{mrf}}(\beta = 5)$  for initialization (a) and  $W = 21$  px; the other initializations result in contours within a few pixels difference from the one in (d).

### C. Influence of Parameter Choices and Initialization

In Fig. 7, we show the influence of different initializations for the Miss America image. We employ the segmenter function  $g_{\text{mrf}}$  because the background is slightly textured;<sup>1</sup> the lighter

<sup>1</sup>Details can be seen by zooming in the electronic version.



Fig. 8. Segmentations obtained with  $g_{\text{mrf}}$  and  $\beta = 5$  and different window side lengths: (a) 11 px, (b) 15 px, (c) 25 px, (d) 31 px. Initialization as in Fig. 7(a).

pixels are outliers to a local normal distribution with very small variance. The segmentation in Fig. 7(d) has been obtained by setting the initial contour as depicted in Fig. 7(a) and using a window side length of size  $21 \times 21$  pixels,  $W = 21$  px. Setting the contour as depicted in Fig. 7(b) and (c) yields segmentations that are within few pixels from the contour presented in Fig. 7(d).

For the same image, the influence of the window size is shown by varying the length of the window side from 11 px to 31 px. The corresponding segmentations are shown in Fig. 8. Here, one can observe that larger windows are more prone to forming stable holes. Altogether, we may conclude that initialization and window size have little influence on the final segmentation. Inaccuracies at the image boundaries are partly due to pixels that do not match the interior LRD, e.g., pixels become lighter at the top of the image, and partly due to boundary effects. Critical pixels in boundary regions have less neighbors in their corresponding windows than those in the interior. The latter problem can be solved when the image is extended by repeating border pixels prior to segmentation and, after segmentation, cropping the result to the original size.

We now compare segmentations with  $g_{\text{mrf}}$  and LRDs plus balloon force, still looking at the Miss America image. The background in Fig. 7 has a homogeneous texture; the lighter pixels in the texture can be included in the curve due to  $g_{\text{mrf}}$ . The background, shirt and arm are all dark; these regions have quite similar values, but locally they are slightly different. The differences can be detected with  $g_{\text{mrf}}$ , as shown in Fig. 7(d). LRDs plus balloon force in (12) do not rely as much on representing the region locally with the correct model, but on what we perceive as strong edges; this motion cannot detect the difference between the two regions, as can be observed in Fig. 9(b) [initialization as in Fig. 7(a)].

In Fig. 10, we show the result for segmenting the cameraman image with  $g_{\text{sim}}$  and LRDs plus balloon force, respectively, starting from two different initializations. All segmentations were obtained with  $W = 11$  px; the balloon force was  $\lambda = 5$ . The segmentations with  $g_{\text{sim}}$  are practically the same, while results for the two initializations for LRDs plus balloon force differ very little.



Fig. 9. Intermediary step (a) and segmentation obtained with LRDs plus balloon force,  $W = 11$  px and  $\lambda = 5$  (b) [initialization as in Fig. 7(a)].

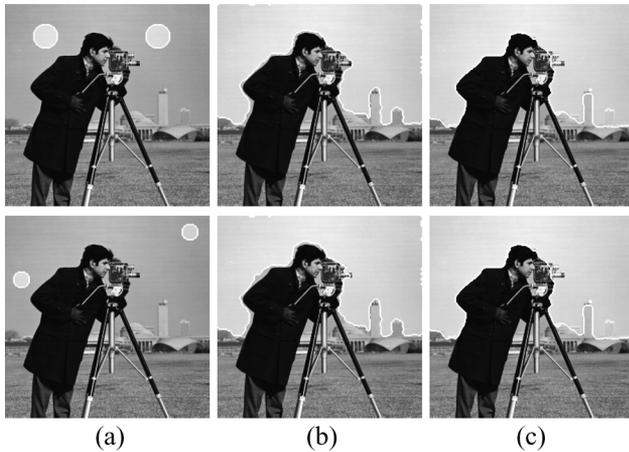


Fig. 10. (a) Two different initializations. (b) Corresponding segmentations obtained with  $g_{sim}$  and  $W = 11$  px. (c) Corresponding segmentations obtained with LRDs plus balloon force and  $W = 11$  px and  $\lambda = 5$ .

Looking at the results in Fig. 10, one may compare the properties of  $g_{sim}$  and LRDs plus balloon force.  $g_{sim}$  is able to detect the faint edges between the sky and the buildings because the function is sensitive to subtle image changes in regions with small local variances; this also explains the hollow around the camera man. LRDs plus balloon force ( $\lambda = 5$ ) is less sensitive, but more accurate in finding boundaries: the contour around the camera man is very accurate, but part of the camera was segmented into the background and some of the building edges are not detected [Fig. 10(c)].

The number of detected building edges depends on the value of  $\lambda$ . The influence of this parameter is shown in Fig. 11. The window width was  $W = 11$  px and  $\lambda$  was varied from 2 to 10 [initialization as in Fig. 10(a) top]. Fig. 11(a) shows segmentation with  $\lambda = 2$ ; observe that more building edges are found. As  $\lambda$  increases, more of the buildings are segmented into the background. For  $\lambda = 8, 9$  the segmentation is very similar to the one for  $\lambda = 10$  shown in Fig. 11(d).

Fig. 12 shows the influence of window size on segmenting the background in the cameraman image. The top row shows segmentations with  $g_{sim}$  and window side lengths of 13 px, 17 px, and 21 px. The source of the differences in segmentation is the color gradient in the sky. For a larger window, the differences between bright outside pixels and the LRD are larger than for a smaller window. The part of the luminous sky center not included in the final contour grows with the window size.

The bottom row in Fig. 12 shows segmentations with LRDs plus balloon force with  $\lambda = 5$  and window side lengths of 7 px, 13 px, and 17 px. In this situation, the larger the window the

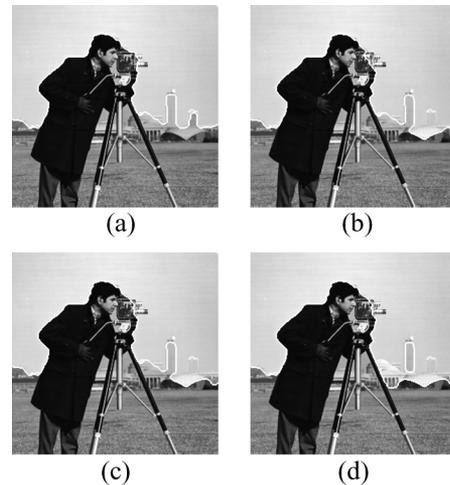


Fig. 11. Segmentation results for different values of  $\lambda$ : (a)  $\lambda = 2$ , (b)  $\lambda = 6$ , (c)  $\lambda = 7$ , (d)  $\lambda = 10$  [with  $W = 11$  px and initialization as in Fig. 10(a) top].



Fig. 12. Top row: Segmentation with window side lengths of 13 px, 17 px, and 21 px and  $g_{sim}$ . Bottom row: Segmentation with window side lengths of 7 px, 13 px, and 17 px and LRDs plus balloon force [initializations as in Fig. 10(a)].

more inaccurate the segmentation and more of the buildings are assigned to the background, because parts of different buildings are represented with the same region descriptor. This decreases the sensitivity to small changes, like the ones between the sky and the buildings. Since the difference between grass and buildings is large enough, that the boundary can still be found.

Fig. 13 shows the influence of the threshold  $\beta$  for  $g_{mrf}$ . This threshold is set to 4 for most images: if 4 or more cliques have positive  $\beta$  for the inner region, the speed is positive. Setting this threshold to 3 usually causes leakage, as can be seen for the background in Fig. 13(a) and for the racket in Fig. 13(c). Setting it to 5 usually prevents the contour in reaching object boundaries, as can be seen looking at the arm shown in Fig. 13(c). However, there are exceptions. Setting this threshold to 5 may prevent leakage as in Fig. 7(d); this is due to the fact that dark pixels in the hair, with intensities similar to those in the background, can be excluded from the segmentation since many have 5 or more neighbors that have intensities that do not match the background. In some situations, it may be useful to set the threshold to 3; highlights on the arm initialized as in Fig. 13(d) are better segmented with  $\beta = 3$  than 4

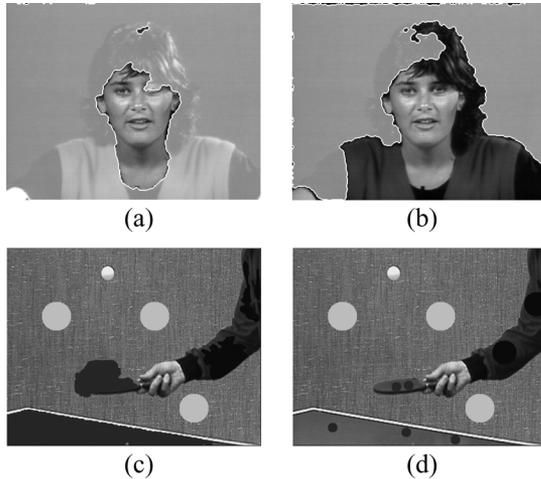


Fig. 13. Influence of varying the threshold value for  $g_{\text{mrf}}$ : (a)  $\beta = 3$ , (b)  $\beta = 4$ , and (c)  $\beta = 3$ , for the racket and table, and  $\beta = 5$  for the arm; the background contours were not evolved so that the over- and under-segmentations can be observed. Initializations: for (a) and (b) as in Fig. 7(a), for (c) as in (d). For parameters leading to correct segmentations see the other figures.

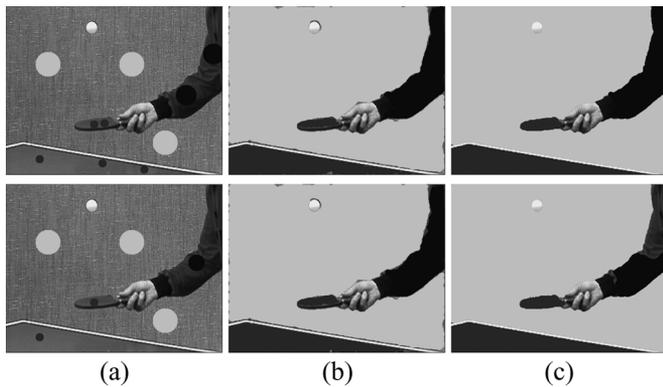


Fig. 14. Segmentations of a table tennis scene. (a) Two initializations for the table tennis scene. (b) Corresponding segmentations with  $g_{\text{mrf}}$  for the arm, racket and table;  $g_{\text{mrf}}$  was chosen to deal with the highlights on these objects;  $g_{\text{sim}}$  was sufficient in dealing with the background. (See text for parameters). (c) Corresponding segmentations with LRDs plus balloon force ( $W = 11$  px,  $\lambda = 5$ ).

(segmentation in Fig. 14(b) bottom). In Fig. 14, when using LRDs plus balloon force, two different initializations result in different segmentations for the arm, but not for other objects. This is due to the lighter crease in the shirt not included in the initial contour. This crease has a smooth shape and forms an edge—pixels on the inside of the final contour are darker, pixels on the outside lighter. This fits the definition of a real boundary, and, thus, the final contour stops at this crease. When the novel segmenter functions are used, the different initializations lead to very similar segmentations.

From our first experiments, we conclude that, for LRDs plus balloon force, a window side length of 11 px and  $\lambda = 5$  are good candidates for starting the segmentation of a gray image. Segmentations in Figs. 14(c) and 15(c) were obtained with these values. For the novel segmenter functions, a good candidate for the window side length is 21 px. Objects in Figs. 14(b) and 15(b) were segmented with this window size. An exception was the racket; since this object is small, a smaller window had to be



Fig. 15. Segmentations of a noisy image. (a) Initialization for a noisy image called Claire. (b) Its segmentation with the segmenter function  $g_{\text{mrf}}$ ,  $W = 21$  px and  $\beta = 4$ . (c) Its segmentation with LRDs plus balloon force,  $W = 11$  px and  $\lambda = 5$ .



Fig. 16. Frame from the color sequence Akyio segmented with LRDs plus balloon force ( $W = 11$  px and  $\lambda = 350$ ) taking the RGB vector as feature. Initialization (left) and final segmentation (right).



Fig. 17. Frame from the color sequence Erik segmented with LRDs plus balloon force ( $W = 11$  px and  $\lambda = 350$ ) taking the RGB vector as feature. Initialization (left) and final segmentation (right).

chosen in order to keep the size of the window comparable to the initial patch. The visual characteristics of the background in Fig. 14(b) change minimally such that the pdf remains constant over large image patches; the segmentation of the background with  $W = 21$  px is thus very similar to the segmentation with  $W = 41$  px.

A good candidate threshold for  $g_{\text{mrf}}$  is  $\beta = 4$ . This threshold has been used to obtain the segmentations for the arm, racket and table in Fig. 14(b) top and the racket and table in Fig. 14(b) bottom; for the arm in this last image  $\beta = 3$ . For Fig. 15(b),  $\beta$  was also 4.  $g_{\text{mrf}}$  was chosen for the arm racket and table in Fig. 14(b) in order to deal with the highlights on these objects;  $g_{\text{sim}}$  was sufficient in dealing with the background.

#### D. Examples of Color Image Segmentation

We have also tested the proposed method on color images. Figs. 16 and 17 present the initial contour and the result of the segmentation of frames from two color video sequences, commonly referred to as Akyio and Erik. For these images, the LRD is considered to be a normally distributed RGB color vector. Both segmentations were achieved by LRDs plus balloon force, with a  $5 \times 5$  px smoothing filter, the size of the window  $W$  was 11 px and  $\lambda = 350$ . These images cannot be segmented correctly with the function  $g_{\text{sim}}$  or  $g_{\text{mrf}}$  because the

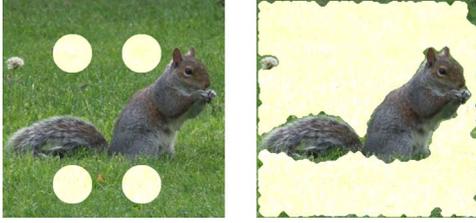


Fig. 18. Image of a squirrel segmented with  $g_{sim}$  and  $g_{mrf}$  taking the RGB vector as feature. Initialization (left) and final segmentation (right).

local regions that include the eyes and eyebrows have two modes with different basins of attraction (one represents skin, the other eyebrows and eyelashes and iris). This type of distribution is badly approximated by a normal distribution. The eye regions are close to the real object boundaries, and the large variances of the Gaussians fitted to those regions cause background pixels to be segmented into the foreground. The contour leaks around the eyes.

The multivariate version of  $g_{mrf}$  was used to segment the grass in Fig. 18(a); a similar result can be obtained with  $g_{sim}$ , with the difference that more small holes are formed in the segmentation. The image was first smoothed to decrease the variance in the grass. Otherwise, the variance of the window sample is too large and causes the contour to leak into the region occupied by the squirrel. In the regions around the squirrel's bottom and tail, grass overlaps fur and there is not a clear smooth boundary. Both local regions include many light pixels. The variance in the local region in the fur is larger than in the grass, and, thus, if we start the segmentation from the squirrel, it will leak into the grass at the bottom.

### E. Computation Time

Our implementation is fast: the  $397 \times 499$  noisy Claire image and the  $409 \times 518$  table tennis scene have been segmented in under 5 s to obtain each of the two results in Figs. 14(b) and (c) and 15(b) and (c), respectively. The faces in the  $409 \times 500$  and  $327 \times 400$  color images in Figs. 16 and 17 were both segmented in under 0.5 s. The number of computations for the algorithm depends the number of evolution steps, the number of contour pixels at each step and the square of the window side length. The first two variables make up for the largest amount of computation time. The Cameraman image is segmented in 0.61 s with  $W = 11$  px and in 0.71 s with  $W = 21$  px, with the initialization from Fig. 10(a), top. The image in Fig. 15 is segmented in 5 s with  $W = 11$  px and in 7 s with  $W = 21$  px, with the initialization from Fig. 15(a). However, if the background of this image is initialized with 4 instead of 2 circles, the computation time can be reduced to under 1 s, because the number of evolution steps decreases considerably. The C++ application runs single threaded (it uses only one of the processor cores) on an Intel(r) Core 2 Duo E6600.

## VIII. CONCLUSIONS AND FUTURE WORK

In this paper, we have concentrated on region-based active contours and region descriptors. Specifically, we have suggested that overlap of pdfs for different regions is problematic, and it

can be reduced if pdfs are computed locally. We have thus proposed to replace global region descriptors with local ones in the framework of active-contours segmentation. For this purpose, we have defined LRDs by choosing representative region features and selecting a probability model to represent their variation. Parameters of the local pdf are computed from samples that lie in a region within a window centered on the active contour. We have proposed to associate energies based on LRDs to active contours by defining segmenter functions for a general-form energy. This procedure has been exemplified with three segmenter functions. We have obtained a region competition equation and an object-oriented approach that proved to have different properties in our experiments. The segmentation results proved to be accurate in situations where the global distributions of foreground and background overlap, and other methods fail. Results could be obtained very fast by reducing calculations to integer arithmetics and choosing a fast level-set algorithm.

Formulating the joint motion of multiple growing contours guided by LRDs gives hope for a more robust method. Robustness could also be improved with the help of texture, diffusion and other image filters, since in our experiments we have only used the pixel intensity/color as feature. Filters cannot help when an object is occluded by other objects; shape priors are needed in this situation and some of our efforts are channeled in this direction. The speed of the algorithm can also be improved; at present only one processor core is used, but since memory requirements are modest, LRDs can be computed in parallel for different windows; we would expect the speed to increase with the number of cores almost linearly.

## APPENDIX

In this appendix, we give a solution for finding the minimum of the following energy:

$$E(\Phi, p) = - \int \int_R (H(\Phi)g(f, \theta^W) - \mu |\nabla H(\Phi)|) dx dy. \quad (21)$$

To find the optimum  $\Phi$ , the Euler-Lagrange equations for the level set function must be derived. For this purpose, it is common to assume that the parameters of  $g$  do not depend on  $\Phi$ . To derive the variation of  $\Phi$ , consider replacing  $\Phi$  by  $\Phi + \epsilon\psi$  where  $\epsilon$  is a very small number. Since  $E$  is minimized by  $\Phi$ ,  $\partial E(\Phi + \epsilon\psi)/\partial \epsilon = 0$  for  $\epsilon = 0$ . Because of  $H'(x) = \delta(x)$ , we have  $|\nabla H(\Phi)| = \delta(\Phi)|\nabla \Phi|$ . According to the chain rule, this partial derivative can be written by simultaneously substituting  $\epsilon = 0$  (and thus obtaining  $\partial E/\partial \Phi$ )

$$\left( \frac{\partial E}{\partial \Phi}, \psi \right) = - \int \int_R g\delta(\Phi)\psi - \mu \left( \delta'(\Phi)|\nabla \Phi| \psi + \delta(\Phi) \frac{\partial |\nabla(\Phi + \epsilon\psi)|}{\partial \epsilon} \right) dx dy. \quad (22)$$

The last partial derivative is written by substituting  $\epsilon = 0$

$$\frac{\partial |\nabla(\Phi + \epsilon\psi)|}{\partial \epsilon} = \frac{1}{2|\nabla(\Phi + \epsilon\psi)|} + \frac{\partial}{\partial \epsilon} ((\Phi_x + \epsilon\psi_x)^2 + (\Phi_y + \epsilon\psi_y)^2) \quad (23)$$

$$= \frac{1}{2|\nabla\Phi|} 2(\Phi_x\psi_x + \Phi_y\psi_y) = \frac{\nabla\Phi\nabla\psi}{|\nabla\Phi|}. \quad (24)$$

By plugging this term into (22) and integrating it by parts, we obtain

$$\begin{aligned} \left(\frac{\partial E}{\partial\Phi}, \psi\right) &= - \iint_R g\delta(\Phi)\psi \\ &\quad - \mu \left( \delta'(\Phi)|\nabla\Phi|\psi + \delta(\Phi) \frac{\nabla\Phi\nabla\psi}{|\nabla\Phi|} \right) dx dy \\ &= - \iint_R g\delta(\Phi)\psi \\ &\quad - \mu \left( \delta'(\Phi)|\nabla\Phi|\psi - \operatorname{div} \left( \delta(\Phi) \frac{\nabla\Phi}{|\nabla\Phi|} \right) \psi \right) \\ &\quad + \int_{\partial R} \frac{\delta(\Phi)}{|\nabla\Phi|} \nabla\Phi n \psi ds \end{aligned} \quad (25)$$

where  $n$  is the outward normal vector to  $\partial R$ . We can write

$$\begin{aligned} \operatorname{div} \left( \delta(\Phi) \frac{\nabla\phi}{|\nabla\phi|} \right) &= \frac{\partial}{\partial x} \left( \delta(\Phi) \frac{\Phi_x}{|\nabla\Phi|} \right) + \frac{\partial}{\partial y} \left( \delta(\Phi) \frac{\Phi_y}{|\nabla\Phi|} \right) \\ &= \delta'(\Phi) \frac{\Phi_x^2}{|\nabla\Phi|} + \delta(\Phi) \frac{\partial}{\partial x} \left( \frac{\nabla\Phi}{|\nabla\Phi|} \right) \\ &\quad + \delta'(\Phi) \frac{\Phi_y^2}{|\nabla\Phi|} + \delta(\Phi) \frac{\partial}{\partial y} \left( \frac{\nabla\Phi}{|\nabla\Phi|} \right) \\ &= \delta'(\Phi)|\nabla\Phi| + \delta(\Phi) \operatorname{div} \left( \frac{\nabla\Phi}{|\nabla\Phi|} \right). \end{aligned} \quad (26)$$

By substituting (26) in (25)

$$\begin{aligned} \left(\frac{\partial E}{\partial\Phi}, \psi\right) &= - \iint_R \left( g\delta(\Phi) + \mu\delta(\Phi) \operatorname{div} \left( \frac{\nabla\Phi}{|\nabla\Phi|} \right) \right) \psi dx dy \\ &\quad + \mu \int_{\partial R} \frac{\delta(\Phi)}{|\nabla\Phi|} \frac{\partial\Phi}{\partial n} \psi ds. \end{aligned} \quad (27)$$

This partial derivative must be zero for all  $\psi$ , and, thus, we must have  $\delta(\Phi)(g + \mu k) = 0$ , where we denote by  $k$  the term  $\operatorname{div}(\nabla\Phi/|\nabla\Phi|)$ . The gradient descent in time for  $\Phi$  can then be written

$$\Phi_t = H'(\Phi)(g + \mu k) \quad (28)$$

with  $\Phi(x, y, 0) = \Phi_0(x, y)$ ,  $(x, y) \in R$  and boundary conditions  $(\delta(\Phi)/\nabla\Phi)(\partial\Phi/\partial n) = 0$  on  $\partial R$ .

#### ACKNOWLEDGMENT

The authors would like to thank the reviewers for valuable comments that helped improve the paper's technical contents and its presentation.

#### REFERENCES

[1] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, January 1988.

[2] C. Xu and J. L. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 359–369, Mar. 1998.

[3] L. D. Cohen, "On active contour models and balloons," *Comput. Vis., Graph., Image Process., Image Understand.* vol. 53, no. 2, pp. 211–218, 1991 [Online]. Available: <http://citeseer.ist.psu.edu/cohen91active.html>

[4] S. Kichenassamy, A. Kumar, P. J. Olver, A. Tannenbaum, and A. J. Yezzi, "Gradient flows and geometric active contour models," in *Proc. Int. Conf. Computer Vision*, 1995, pp. 810–815.

[5] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: A level set approach," *IEEE Trans. Pattern Anal. Mach. Intell.* vol. 17, no. 2, pp. 158–175, Feb. 1995.

[6] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *Int. J. Comput. Vis.*, vol. 22, no. 1, pp. 61–79, 1997.

[7] N. Paragios and R. Deriche, "Unifying boundary and region-based information for geodesic active tracking," presented at the IEEE Computer Soc. Conf. Computer Vision and Pattern Recognition, 1999.

[8] L. Cohen, E. Bardinet, and N. Ayache, "Surface reconstruction using active contour models," presented at the SPIE Conf. Geometric Methods in Computer Vision, 1993.

[9] Ecabert and O. Thiran, "Variational image segmentation by unifying region and boundary information," presented at the 16th Int. Conf. Pattern Recognition, 2002.

[10] S. J. Besson, M. Barlaud, and G. Aubert, "Detection and tracking of moving objects using a new level set based method," presented at the Int. Conf. Pattern Recognition, 2000.

[11] S. C. Zhu and A. Yuille, "Unifying snake/balloon, region growing and bayes/mdl/energy for multi-band image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 9, pp. 884–900, Sep. 1996.

[12] G. Aubert, M. Barlaud, O. Faugeras, and S. Jehan-Besson, "Image segmentation using active contours: Calculus of variations or shape gradients?," *SIAM J. Appl. Math.*, vol. 63, no. 6, pp. 2128–2154, Jun. 2003.

[13] S. Jehan-Besson and M. Barlaud, "Dream2s: Deformable regions driven by an eulerian accurate minimization method for image and video segmentation," *Int. J. Comput. Vis.*, vol. 53, no. 1, pp. 45–70, 2003.

[14] T. Roy, É. Debreuve, M. Barlaud, and G. Aubert, "Segmentation of a vector field: Dominant parameter and shape optimization," *J. Math. Imag. Vis.*, vol. 24, no. 2, pp. 259–276, 2006.

[15] T. Roy, M. Barlaud, É. Debreuve, and G. Aubert, "Vector field segmentation using active contours: Regions of vectors with the same direction," presented at the Workshop on Variational, Geometric, and Level Set Methods in Computer Vision (VLSM), 2003.

[16] N. Paragios and R. Deriche, "Geodesic active contours and level sets for the detection and tracking of moving objects," *IEEE Trans. Pattern Anal. Mach. Intell.* vol. 22, no. 3, pp. 266–280, Mar. 2000.

[17] T. Brox, M. Rousson, R. Deriche, and J. Weickert, "Unsupervised segmentation incorporating colour, texture and motion," presented at the Computer Analysis of Images and Patterns 2003, Lecture Notes Comput. Sci.

[18] N. Paragios and R. Deriche, "Geodesic active contours for supervised texture segmentation," in *Proc. IEEE Computer Soc. Conf. Computer Vision and Pattern Recognition*, Los Alamitos, CA, 1999, vol. 2, p. 2422.

[19] T. Brox and J. Weickert, "A tv flow based local scale measure for texture discrimination," presented at the 8th Eur. Conf. Computer Vision, 2004.

[20] M. Rousson, T. Brox, and R. Deriche, "Active unsupervised texture segmentation on a diffusion based feature space," presented at the IEEE Conf. Computer Vision and Pattern Recognition, 2003.

[21] D. Cremers, F. Tischhäuser, J. Weickert, and C. Schnörr, "Diffusion snakes: Introducing statistical shape knowledge into the mumford-shah functional," *Int. J. Comput. Vis.*, vol. 50, no. 3, pp. 295–313, 2002.

[22] M. Gastaud, M. Barlaud, and G. Aubert, "Combining shape prior and statistical features for active contour segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 5, pp. 726–734, May 2004.

[23] M. Rochery, I. H. Jermyn, and J. Zerubia, "Higher order active contours," *Int. J. Comput. Vis.* vol. 69, no. 1, pp. 27–42, 2006.

[24] N. Paragios and R. Deriche, "Geodesic active regions: A new paradigm to deal with frame partition problems in computer vision," *J. Vis. Commun. Image Represent.*, vol. 13, no. 1/2, pp. 249–268, 2002.

[25] A. Chakraborty, L. H. Staib, and J. S. Duncan, "Deformable boundary finding in medical images by integrating gradient and region information," *IEEE Trans. Med. Imag.*, vol. 15, no. 6, pp. 859–870, Jun. 1996.

- [26] T. Chan and L. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, pp. 266–277, 2001.
- [27] T. Chan and L. Vese, "An active contour model without edges," in *Proc. Scale-Space Theories in Computer Vision*, 1999, pp. 141–151.
- [28] C. Chesnaud, P. Refregier, and V. Boulet, "Statistical region snake-based segmentation adapted to different physical noise models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1145–1157, Nov. 1999.
- [29] T. Brox and J. Weickert, "Level set based image segmentation with multiple regions," in *Pattern Recognition*, C.-E. Rasmussen, H. Bulthoff, M. Giese, and B. Scholkopf, Eds., 2004 [Online]. Available: <http://www.mia.uni-saarland.de/Publications/brox-dagm04.pdf>, Springer LNCS 3175
- [30] D. Freedman and T. Zhang, "Active contours for tracking distributions," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 518–526, Apr. 2004.
- [31] Y. Shi and W. Karl, "A fast level set method without solving pdes," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, 2005 [Online]. Available: <http://www.iss.bu.edu/students/yshi/Publications/ShiKarlICASSP05.pdf>
- [32] D. Mumford and J. Shah, "Boundary detection by minimizing functionals," presented at the IEEE Conf. Computer Vision and Pattern Recognition 1985 [Online]. Available: <http://www.math.neu.edu/~shah/papers/MS85.pdf>
- [33] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *J. Roy. Statist. Soc.*, vol. 39, p. 138, 1977.
- [34] R. Ronfard, "Region-based strategies for active contour models," *Int. J. Comput. Vis.*, vol. 13, no. 2, pp. 229–251, 1994.
- [35] P. Brigger, J. Hoeg, and M. Unser, "B-Spline snakes: A flexible tool for parametric contour detection," *IEEE Trans. Image Process.*, vol. 9, no. 9, pp. 1484–1496, Sep. 2000.
- [36] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations," *J. Comput. Phys.* vol. 79, pp. 12–49, 1988 [Online]. Available: <http://www.math.berkeley.edu/~sethian/2006/Papers/sethian.osher.88.pdf>
- [37] V. Caselles, F. Catté, T. Coll, and F. Dibos, "A geometric model for active contours in image processing," *Numer. Math.*, vol. 66, no. 1, pp. 1–31, 1993.
- [38] C. Samson, L. Blanc-Feraud, G. Aubert, and J. Zerubia, "A level set model for image classification," in *Proc. Scale-Space Theories in Computer Vision*, 1999, pp. 306–317 [Online]. Available: <http://citeseer.ist.psu.edu/samson99level.html>
- [39] L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the mumford and shah model," *Int. J. Comput. Vis.*, vol. 50, no. 3, pp. 271–293, 2002.
- [40] T. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Trans. Signal Process.*, vol. 40, no. 4, pp. 901–914, Apr. 1992.
- [41] S. Lankton, D. Nain, A. Yezzi, and A. Tannenbaum, "Hybrid geodesic region-based curve evolutions for image segmentation," presented at the Proc. SPIE Medical Imaging, 2007.
- [42] J. Piovano, M. Rousson, and T. Papadopoulos, "Efficient segmentation of piecewise smooth images," in *Proc. Scale Space and Variational Methods in Computer Vision*, 2007, vol. LNCS 4485, pp. 709–720.
- [43] T. Brox and D. Cremers, "Implicit active contours driven by local binary fitting energy," presented at the Int. Conf. Scale Space and Variational Methods in Computer Vision, 2007.
- [44] J. Sethian, *Level Set Methods and Fast Marching Methods. Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science.* Cambridge, U.K.: Cambridge Univ. Press, 1999.
- [45] Y. Shi and W. C. Karl, "Real-time tracking using level sets," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, 2005, pp. 34–41.



**Cristina Darolti** received the diploma in economic informatics from the University of Economic Studies, Bucharest, Romania, finishing among the top five percent of her year. She is currently pursuing the Ph.D. degree at the Institute for Signal Processing, Lübeck, Germany.

Her research interest include image processing, with a focus on image segmentation and image understanding, computer vision, and machine learning. Her work is currently applied in medical applications.



**Alfred Mertins** (M'96–SM'08) received the Dipl.-Ing. degree from the University of Paderborn, Germany, in 1984, the Dr.-Ing. degree in electrical engineering and the Dr.-Ing. habil. degree in telecommunications from the Hamburg University of Technology, Germany, in 1991 and 1994, respectively.

From 1986 to 1991, he was a Research Assistant at the Hamburg University of Technology and from 1991 to 1995, he was a Senior Scientist at the Microelectronics Applications Center, Hamburg. From 1996 to 1997, he was with the University of Kiel, Germany, and from 1997 to 1998, with the University of Western Australia. In 1998, he joined the University of Wollongong, where he was at last an Associate Professor of Electrical Engineering. From 2003 to 2006, he was a Professor in the Faculty of Mathematics and Science, University of Oldenburg, Germany. In November 2006, he joined the University of Lübeck, Germany, where he is a Professor and Director of the Institute for Signal Processing. His research interests include speech, audio, and image processing, wavelets and filter banks, pattern recognition, and digital communications.



**Christoph Bodensteiner** (M'07) received the M.S. degree in computer science from the Technical University Munich, Germany. He is currently pursuing the Ph.D. degree at the Institute for Robotics at the University of Lübeck, Lübeck, Germany.

He was a Software Engineer developing systems for surgical navigation. His research interests include medical applications for mobile C-arms, nonrigid 2-D/3-D registration, and reconstruction in medical imaging.



**Ulrich G. Hofmann** received the Diploma in physics and the Ph.D. degree in biophysics from the Technical University of Munich, Munich, Germany, in 1993 and 1996, respectively.

He was an EU-Fellow (HCM) at the Abo Akademi, Turku, Finland, in 1996, and a Feodor-Lynen-Fellow at the California Institute of Technology, Pasadena, from 1997–1998. Since 1999, he has been a Senior Researcher and Group Leader in neuroengineering at the Institute for Signal Processing, University of Lübeck, Lübeck, Germany. He teaches biomedical engineering and computational neurosciences. His aim is to improve invasive brain-machine interfaces.