Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition Letters 28 (2007) 1946–1956
Pattern Recognition Letters
www.elsevier.com/locate/patrec

Globally adaptive region information for automatic color–texture image segmentation

Mohand Saïd Allili *, Djemel Ziou

Department of Computer Science, Faculty of Science, University of Sherbrooke, Sherbrooke, Quebec, Canada J1K 2R1

Received 28 March 2006; received in revised form 5 April 2007
Available online 27 June 2007

Communicated by S. Ablameyko

Abstract

In this paper, we propose an automatic segmentation of color–texture images with arbitrary numbers of regions. The approach combines region and boundary information and uses active contours to build a partition of the image. The segmentation algorithm is initialized automatically by using homogeneous region seeds on the image domain. The partition of the image is formed by evolving the region contours and adaptively updating the region information formulated using a mixture of pdfs. We show the performance of the proposed method on examples of color–texture image segmentation, with comparison to two state-of-the-art methods.

Keywords: Color; Texture; Polarity; Level sets; Automatic segmentation

1. Introduction

In the last decades, image segmentation has been the subject of active research in computer vision and image processing. While most of the past work has focused on developing algorithms based on either color (Allili and Ziou, 2005; Ben Ayed et al., 2005; Sifakis et al., 2002; Zhu and Yuille, 1996) or texture features (Jain and Farrokhnia, 1991; Liapis et al., 2004; Paragios and Deriche, 2002b), some attempts have been made to combine texture and color features to build a unified segmentation framework (Carson et al., 2002; Freixenet et al., 2004; Rousson et al., 2003; Sagiv et al., 2006). Indeed, real-world images generally contain a combination of color and texture and, therefore, combining color and texture features would be of significant benefit in distinguishing between regions having the same color but different textures and vice versa (Allili and Ziou, 2005; Freixenet et al., 2004; Malik et al., 2001). Also, the ability to perform the segmentation in unsupervised fashion is desirable for several applications, but remains a great challenge. Indeed, texture segmentation methods in their own right require the number and type of textures to be given *a priori* (Chan and Vese, 2001; Paragios and Deriche, 2002b), and hence the segmentation is not fully automatic. Finally, to the best of our knowledge, no work in the past attempted to exploit together region and boundary information for unsupervised color–texture image segmentation.

Recently, there has been considerable investigation into the use of active contours for color and texture segmentation. In this approach, the segmentation problem is formulated by minimizing an energy functional according to a single or multiple region contour(s) to build a partition of the image into uniform regions (Chakraborty and Duncan, 1999; Chan and Vese, 2001; Paragios and Deriche, 2002b; Rousson et al., 2003; Zhu and Yuille, 1996). The advent of level set formalism has permitted for handling automatically topology changes for the contours and developing stable numerical schemes for the segmentation

* Corresponding author. Tel.: +1 819 821 8000x63247,62859; fax: +1 819 821 8200.
E-mail addresses: ms.allili@usherbrooke.ca (M.S. Allili), d.ziou@usherbrooke.ca (D. Ziou).

0167-8655/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2007.05.002

methods (Osher and Sethian, 1998). Among the recent methods, some use boundary information to guide the contour evolution (Caselles et al., 1997; Cohen, 1991; Yezzi et al., 1997), while others are guided by the statistics of the regions enclosed by the contours (Ben Ayed et al., 2005; Chan and Vese, 2001; Mansoursi et al., 2006; Rousson et al., 2003; Yezzi et al., 2002). In the latter group, two different approaches have been investigated for extracting and modeling region information. In the first approach, the region information (the number and statistics of the regions) is calculated prior to curve evolution, using the whole image data. Region contours are then evolved to build a partition of the image that conforms to the *a priori* – extracted information (Paragios and Deriche, 2002a,b; Sifakis et al., 2002). In the second approach, the segmentation is data-driven; that is, the region information is extracted adaptively to the region contour evolution. In general, however, the approach assumes the number of regions is fixed *a priori* by the user, which means the segmentation is not fully unsupervised (Ben Ayed et al., 2005; Chan and Vese, 2001; Rousson et al., 2003; Yezzi et al., 2002).

In this paper, we propose an automatic method for color–texture segmentation based on active contours. The segmentation is steered by the combination of region and boundary information. The region information is based on mixture modeling of the combined color and texture features, while the boundary information is modeled using the polarity information. Based on an automatic region initialization that we proposed in (Allili and Ziou, 2005), the segmentation evolves the initial regions and updates the mixture parameters adaptively to the data. The algorithm starts with an initial number of regions that constitutes the number of mixture components. As the region contours evolve, all the mixture parameters, with the number of components, are adaptively updated to the image data. The implementation of our method is based on the level set approach. We show real-world image segmentation examples to demonstrate the performance of the proposed method in achieving a fully automatic segmentation of color–texture images with an arbitrary number of regions.

This paper is organized as follows: In Section 2, some work related to the proposed method is presented. In Section 3, we present the proposed variational model for segmentation. Finally, in Section 4, some experimental results are given, followed by a conclusion.

2. Related work

Most of the segmentation methods in the past considered only color or texture features to build a partition of the image into homogeneous regions. A great number of methods has been proposed for color segmentation (Lucchese and Mitra, 2001), and texture segmentation in its own right has been the focus of several studies. The methods explored for texture segmentation fall into two main groups. Supervised texture segmentation rests on a prior knowledge about the type and number of textures to be segmented (Chan and Vese, 2001; Gimel'farb, 2001; Paragios and Deriche, 2002b; Talbar et al., 1998). In contrast, unsupervised segmentation does not use any prior knowledge. Most often, clustering techniques are used to build an image partition composed of regions with uniform textures.

Among the previous studies dealing with unsupervised texture segmentation, we find those that consider the texture image as arising from probability distributions on Markov random fields (MRFs) (Hofmann et al., 1998; Kervrann and Heitz, 1995; Manjunath and Chellappa, 1991). Furthermore, knowing the parameters of the underlying distribution, such methods are able to synthesize textures by using sampling techniques. The major drawback of MRF-based segmentation, however, is that it is computationally prohibitive. The authors (Deng and Manjunath, 2001; Laine and Fan, 1993) propose unsupervised texture segmentations based, respectively, on the response of filter-banks of Gabor functions and wavelet coefficients as texture features. Although the approaches do not use prior knowledge about the type of texture to be segmented, they require the number of regions to be known. A remark should be made on the above approaches: they are generally dedicated to intensity image segmentation.

Recently, a model has been proposed to unify the above unsupervised approaches in a probabilistic framework (Zhu and Yuille, 1996). The authors of that work refer to their model as "*region competition*". The contour evolution was implemented using the snake model, via the minimization of an energy functional. Since then, variants of this approach have been proposed. The authors (Sagiv et al., 2006; Unal et al., 2005) propose approaches for bimodal image segmentation by deformation of a single active contour. Although these approaches perform well for segmenting a single object lying on a background, generalization to an arbitrary number of regions remains complex for two reasons. The segmented regions are assumed to be connected in the work (Unal et al., 2005), which is not allways the case in real-world images. In the work (Sagiv et al., 2006), the deformation of the contour is guided using texture boundary information. The model allows for segmenting a textured object lying on a uniform background. However, it does not permit discrimination of multiple textures in the image. In the works (Chan and Vese, 2001; Rousson et al., 2003), an explicit representation of the image regions is used for segmentation, by means of multiple active contours and their intersections. This approach yields a partition of the image into an arbitrary number of regions. However, the number of regions must be a power of 2. Recently, an approach to build a partition of an image, by simultaneously evolving multiple active contours and controlling their intersection was developed (Ben Ayed et al., 2005; Mansoursi et al., 2006). The approach has been applied successfully to color and texture SAR image segmentation with arbitrary numbers of regions. However, the number of regions is assumed to be known.

3. Description of the segmentation model

3.1. Region initialization

To automatically initialize the region contours, we proceed by the algorithm that we proposed recently in (Allili and Ziou, 2005). In this algorithm, an adaptive scale is used first to smooth texture regions while roughly preserving their boundaries. To detect whether a pixel lies on a texture, the method looks at the structure of the color gradient vectors in the pixel neighborhood, which is formulated using the *polarity information*. Let $\vec{v} = (v_x, v_y)$ be the vector pointing to the strongest first-order directional derivative of the color image, as proposed in (Drewniok, 1994). A structure matrix $\mathscr{S}$ for the pixel $\boldsymbol{x} = (x, y)$ is defined by

$$\mathscr{S}(\boldsymbol{x}) = G_\sigma \star (\vec{v}^{\mathrm{T}}\vec{v}) = G_\sigma \star \begin{pmatrix} v_x v_x & v_x v_y \\ v_x v_y & v_y v_y \end{pmatrix} \tag{1}$$

where $\vec{v}^{\mathrm{T}}$ denotes the transpose of the vector $\vec{v}$ and G_σ is a Gaussian kernel with a scale σ that smoothes each element of the matrix $\mathscr{S}$ in a neighborhood of the pixel, $\mathscr{W}(\boldsymbol{x})$, by using the convolution operation $\star$. Assume now that v_1 and v_2 are the eigenvalues of $\mathscr{S}$, where $v_1 > v_2$. When $v_1 \gg v_2$, the neighborhood $\mathscr{W}(\boldsymbol{x})$ has a dominant orientation in the direction of the eigenvector that corresponds to v_1. Let us denote the normalized vector in this direction by $\vec{\eta}$. We express the polarity information $P(\boldsymbol{x})$, that measures the extent to which the strongest first-order directional derivative vectors in the neighborhood of $\boldsymbol{x}$ are oriented in the same direction, by the following function:

$$P(\boldsymbol{x}) = \sum_{(u,v)\in\mathscr{W}(\boldsymbol{x})} G_\sigma * \langle \vec{v}(u,v), \vec{\eta} \rangle \tag{2}$$

where $\langle\,\rangle$ denotes the vector scalar product. The smoothing scale for the pixel $\boldsymbol{x}$ is chosen by looking at the behavior of the polarity to the changing of the value of σ. In a typical image region, homogeneous in color or texture, region edges will be located where the polarity maintains values near 1 for all the considered scale values; whereas the polarity vanishes inside a color/texture region when the value of the scale is increased (since multiple vector directions will be included in the sum (2)). By varying the scale σ from 1 to 6, we choose the smoothing scale beyond which the polarity does not vary more than a fixed threshold τ.

In a second phase for region initialization, regularly-spaced seeds are initialized on the smoothed image domain and only homogeneous seeds are retained. A seed is considered homogeneous if the polarity of all the pixels inside it is close to 0. The homogeneous seeds are used subsequently to initialize the region contours in the original image. Fig. 1 shows two examples of polarity calculation and homogeneous seeds initialization. To display the boundary plausibility and the polarity images, we have changed their dynamic range. We can note in the figure that the polarity information more accurately distinguishes the boundaries of the salient regions.

The final step of our region initialization algorithm consists of grouping the seeds into regions. Here, we use a combination of color and texture features to calculate a mixture of pdfs that models the distribution of these features. We use the CIE-$L^*a^*b^*$ color space, which is perceptually uniform. For texture, we use features calculated from the correlogram of the pixel neighborhood (Huang et al., 1997). An element of the correlogram matrix $C^{\upsilon,\theta}(c_i, c_j)$ should give the probability that, given a pixel $\boldsymbol{x}_1$ of color c_i, a pixel $\boldsymbol{x}_2$ at distance υ and orientation θ from $\boldsymbol{x}_1$ is of color c_j. We calculate the correlogram for four orientations, $\theta \in \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$, and three displacements $\upsilon \in \{1, 2, 3\}$. We derive from each correlogram four typical features, namely: energy (EN), entropy (ET), inverse-difference-moment (idm) and correlation (C), where we average each feature over the number of orientations and displacements (see Appendix for the formulation of these features). By combining the color and texture features, we end up with a seven-dimensional region feature vector.

3.1.1. Fitting a mixture model to the regions

To model the distribution of the features, we use a mixture of general Gaussian distributions (GGD) as in (Allili and Ziou, 2005). The GGD formalism generally yields a

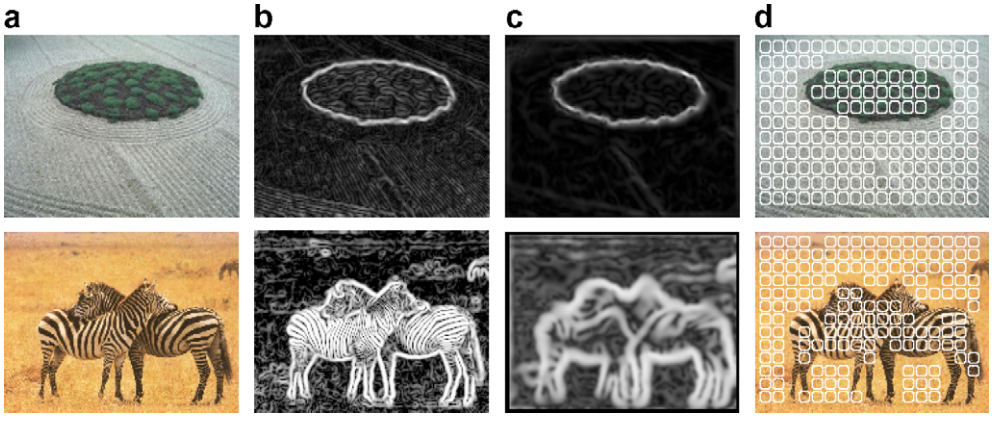

Fig. 1. An example of polarity calculation: (a) represents the original image, (b) and (c) represent the image of the boundary plausibility and the polarity information. Finally, (d) represents the initialization of the homogenous seeds.

good compromise, fitting the image data while not over-fitting the real number of components in the mixture (Allili et al., 2007). The GGD model is formulated as follows. Let $\vec{U}(x) = (u_1(x), \ldots, u_d(x))$ be a d-dimensional feature vector; the probability of the vector according to the mixture is given by

$$p(\vec{U}(x)|\theta_k) = \prod_{i=1}^{d}\left[\frac{\varrho_{ki}}{2\sigma_{ki}} \cdot \exp\left(-\psi_{ki}\left|\frac{u_i(x)-\mu_{ki}}{\sigma_{ki}}\right|^{\lambda_{ki}}\right)\right] \quad (3)$$

where the coefficients ϱ and ψ are given by $\varrho_{ki} = \frac{\lambda_{ki}\sqrt{\frac{\Gamma(3/\lambda_{ki})}{\Gamma(1/\lambda_{ki})}}}{\Gamma(1/\lambda_{ki})}$ and $\psi_{ki} = [\frac{\Gamma(3/\lambda_{ki})}{\Gamma(1/\lambda_{ki})}]^{\frac{\lambda_{ki}}{2}}$. We denote by $\Gamma(m)$ the gamma function that is defined by the integral: $\Gamma(m) = \int_0^\infty z^{m-1}e^m\,dz$ where m and z are real variables. In function (3), μ_{ki} and σ_{ki} are the pdf location and standard deviation parameters in the ith dimension of the feature vector $\vec{U}(x)$. In the same dimension, the parameter $\lambda_{ki} \geqslant 1$ controls the degree to which the tails of the distribution are peaked or flat (as shown in Fig. 2 for the case of two dimensions).

By assuming M regions, a mixture of M GGDs is calculated for the seed data, using the maximum likelihood estimation (Allili and Ziou, 2005). In order to estimate the number of mixture components, M, note that several criteria have been proposed in the past for the Gaussian mixture model, such as Akaike (AIC) (Akaike, 1974), minimum description length (MDL) (Rissanen, 1978) and minimum message length MML (Baxter and Olivier, 2000). However, none of these criteria has been developed for the general Gaussian mixture. For simplicity, we use the AIC criterion which is easy to calculate and, in general, gives a good performance, compared to the other criteria (see McLachlan and Peel, 2000). The general AIC formula is given as follows:

$$\mathrm{AIC} = -\log(\mathscr{L}(\Theta)) + 2\xi \quad (4)$$

where $\mathscr{L}(\Theta) = (\prod_{x\in\mathrm{Seeds}}\sum_{k=1}^{M}\pi_k p(\vec{U}(x)|\theta_k))$ is the likelihood of the data, and ξ is the number of parameters of the mixture. The log-likelihood gives the degree of mixture fit to the data (smaller values indicate a worse fit). The second term of the AIC is a regularization that penalizes overfitting the number of components in the mixture. In a final step of region initialization, we group the seeds into regions by maximizing for each seed the membership probability of its features vectors, given by the following function:

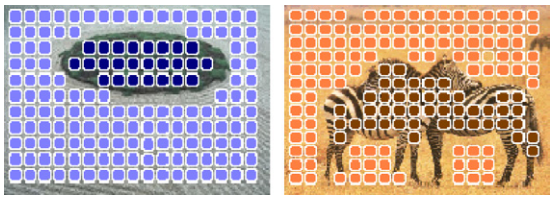

Fig. 3. Grouping of the homogeneous seeds generated for the examples in Fig. 1 into regions.

$$\mathrm{argmax}_k\left(\prod_{x\in\mathrm{Seed}}\left(\pi_k p(\vec{U}(x)|\theta_k)\right)\right) \quad (5)$$

where π_k, $k \in \{1, \ldots, M\}$, designates the *a priori* probability of the kth mixture component. Fig. 3 shows the result of seed grouping for the images shown in Fig. 1, where each image contains two regions.

3.2. Adaptive texture–color segmentation

In the following, we use the notation Ω_k and $\partial\Omega_k$ to designate, respectively, a region and its boundaries. The objective of the segmentation is to create a partition of the image composed of M regions, $P = \{\Omega_1, \ldots, \Omega_M\}$, where $\bigcup_{i=1}^{M}\Omega_k = \Omega$ and the formed regions are considered to be homogeneous with respect to variation of color and texture characteristics. We formulate the segmentation using a variational model, as we recently proposed in (Allili and Ziou, 2005). In the model, the region information, based on a mixture of general Gaussians, is calculated adaptively to the data as the region contours evolve. The objective function underlying the model is formulated using the following energy functional:

$$E(\partial\Omega, \Theta) = \sum_{k=1}^{M}\left[\alpha\oint_{\partial\Omega_k} g(P(s))\,ds + \beta\int\int_{\Omega_k} -\log(p(\theta_k|\vec{U}(x)))\,dx\right] \quad (6)$$

where $\partial\Omega = \bigcup_{k=1}^{M}\partial\Omega_k(s)$, and Θ designates the mixture parameters that include the parameters of each pdf of the mixture θ_k and the *a priori* probabilities π_k, $k \in \{1, \ldots, M\}$. The boundary information is added in the first term of

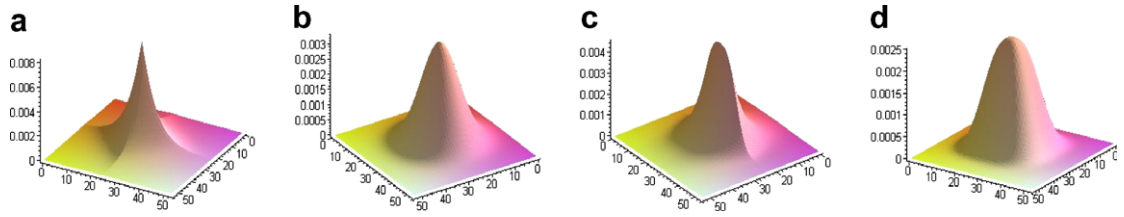

Fig. 2. Different representations of a two-dimensional general Gaussian pdf according to the parameter $\vec{\lambda}$. $\vec{\mu}$ and $\vec{\sigma}$ are fixed at (23,24) and (7,7), respectively. (a) $\vec{\lambda} = (1.1, 1.1)$, (b) $\vec{\lambda} = (2, 2)$, (c) $\vec{\lambda} = (1.1, 2.8)$ and (d) $\vec{\lambda} = (2.8, 2.8)$.

functional (6) using the geodesic active contours (GAC) formalism (Caselles et al., 1997), where the color/texture boundary plausibility corresponds to the absolute value of the polarity information $|P(s)|$. Here, g is a strictly decreasing function of the polarity. The second term of functional (6) represents the region information. This term is minimal when each pixel is classified into the region that has the maximum posterior probability $p(\theta_k|\vec{\boldsymbol{U}}(\boldsymbol{x})) \propto \pi_k p(\vec{\boldsymbol{U}}(\boldsymbol{x})|\theta_k)$. Functional (6) is minimized alternately according to the region contours $\partial\Omega_k$ and the mixture parameters Θ. To minimize the energy according to the region contours, we calculate the Euler–Lagrange equations. After introducing the level set formalism for the contours (Osher and Sethian, 1998), we obtain the following motion equation for each region contour:

$$\frac{\mathrm{d}\Phi_k}{\mathrm{d}t} = (\alpha\mathscr{V}_{\mathrm{b}}(\Phi_k) - \beta\mathscr{V}_{\mathrm{r}}(\Phi_k))\|\nabla\Phi_k\| \tag{7}$$

where

$$\mathscr{V}_{\mathrm{b}}(\Phi_k) = g(P(\Phi_k))\kappa + \nabla g(P(\Phi_k)) \cdot \frac{\nabla\Phi_k}{\|\nabla\Phi_k\|}$$

$$\mathscr{V}_{\mathrm{r}}(\Phi_k) = \log(\pi_k p(\vec{\boldsymbol{U}}(\Phi_k)|\theta_k)) - \log(\pi_h p(\vec{\boldsymbol{U}}(\Phi_k)|\theta_h))$$

Here, $\Phi_k : \Re^2 \to \Re$ is a level set function and the contour $\partial\Omega_k$ is represented by its zero level set. The symbol κ stands for the curvature of the zero level set. The term $\mathscr{V}_{\mathrm{b}}$ represents the boundary velocity that regularizes the curve and aligns it with the region boundaries. On the other hand, the term $\mathscr{V}_{\mathrm{r}}$ represents the region velocity. In the interior of a region, the boundary term vanishes and the contour is driven only by this velocity in the direction that minimizes the Bayes error classification for the region features (Duda et al., 2000). Here, with the objective of obtaining the best classification of pixels, the region term is formulated as a competition for a given pixel between the current region Ω_k and the region $\Omega_h \neq \Omega_k$ that has the maximum posterior probability for the pixel region feature vector $\vec{\boldsymbol{U}}(\boldsymbol{x})$.

For the minimization of functional (6) according to the mixture parameters, we used the same approach that we proposed in (Allili and Ziou, 2005). This minimization has the objective of fitting the mixture parameters to the region data enclosed by the contours after evolution. The updating equations for the parameters are given, as follows: $\forall k \in \{1,\ldots,M\}$, $\forall i \in \{1,\ldots,d\}$

$$\mu_{ki}^{(\ell+1)} = \frac{\int\int_{\Omega_k}(|u_i(\boldsymbol{x}) - \mu_{ki}^{(\ell)}|)^{\lambda_{ki}-2}u_i(\boldsymbol{x})\mathrm{d}\boldsymbol{x}}{\int\int_{\Omega_k}(|u_i(\boldsymbol{x}) - \mu_{ki}^{(\ell)}|)^{\lambda_{ki}-2}\mathrm{d}\boldsymbol{x}} \tag{8}$$

$$\sigma_{ki}^{(\ell+1)} = \left(\frac{\lambda_{ki}\psi_{ki}}{\int\int_{\Omega_k}\mathrm{d}\boldsymbol{x}}\int\int_{\Omega_k}(|u_i(\boldsymbol{x}) - \mu_{ki}^{(\ell+1)}|)^{\lambda_{ki}}\mathrm{d}\boldsymbol{x}\right)^{\frac{1}{\lambda_{ki}}} \tag{9}$$

$$\pi_k = \left(\frac{\int\int_{\Omega_k}\mathrm{d}\boldsymbol{x}}{\sum_{j=1}^{M}\int\int_{\Omega_j}\mathrm{d}\boldsymbol{x}}\right) \tag{10}$$

The parameter λ that determines the shape of the distributions is updated using the Newton–Raphson method (Allili and Ziou, 2005; Allili et al., 2007). The updating equation is given by the following formula:

$$\lambda_{jk}^{(\ell+1)} \simeq \lambda_{jk}^{(\ell)} - \left(\frac{\partial^2\log(\mathscr{L}(\Theta))}{\partial(\lambda_{jk}^{(\ell)})^2}\right)^{-1}\frac{\partial\log(\mathscr{L}(\Theta))}{\partial\lambda_{jk}^{(\ell)}} \tag{11}$$

where $\mathscr{L}(\Theta) = \left(\prod_{\boldsymbol{x}\in\bigcup_{k=1}^{M}\Omega_k}\sum_{k=1}^{M}\pi_k p(\vec{\boldsymbol{U}}(\boldsymbol{x})|\theta_k)\right)$.

3.3. Adaptive number of regions

So far, the number of regions used for curve evolution is the one initially calculated by the AIC criterion, using the homogeneous seed data (i.e., this number represents the number of components of the mixture of pdfs fitting the seed data). However, with the curve evolution, the mixture parameters are updated by the data newly added to regions. With the new set of region data, it is natural to formulate the number of regions, also, as a free variable to be determined adaptively by the segmentation model.

To update the number of regions and make the segmentation fully unsupervised, an explicit criterion is used in our work, which is based on the minimization of functional (6). After evolving the level set functions by using a fixed number of regions, we check whether the fusion (i.e., $M \leftarrow M-1$) of two contacting regions minimizes the energy functional. Let $\partial\Omega'$ and Θ' be the resultant region contours and mixture parameters after a potential region fusion. The fusion is operated if the energy functional (6) decreases; that is,

$$E(\partial\Omega', \Theta') < E(\partial\Omega, \Theta) \tag{12}$$

Using the energy functional (6) explicitly in determining the number of regions has the following advantages:

- In past work, when the number of regions is unknown it is calculated in a preliminary stage for segmentation, using a mixture estimate of the image histogram (Allili and Ziou, 2005; Paragios and Deriche, 2002a). By using the energy functional (6) for updating the number of regions, the contribution of the boundary information is also included as a regularization which penalizes the segmentation for having region contours lying on homogeneous parts of the image, thereby reducing the risk of image over-segmentation.
- Since the initial seeds capture the homogeneous parts of the image, isolated seeds may create separate regions that could over-segment the image. This situation may arise in a region that contains small areas different in color from that of the main region. These areas could arise from specularities and shadows, for example. In this case, seeds lying on those areas may create their own isolated regions that could over-segment the image. The updating of the number of regions by the

fusing step, using Eq. (12), corrects such situations, by adding the small regions to their neighboring regions.

3.4. Summary of the algorithm

The following script summarizes the steps in segmentation algorithm:

Algorithm:
Input: Color–texture image,
Output: Segmented image.
1. Calculate the polarity of pixels and smooth the image.
2. Initialize the mixture parameters and the region contours.
3. **While** (The level sets have not converged) **do** {
   - Propagate the level set functions using Eq. (7).
   - Update the mixture parameters using Eqs. (8),(9),(10) and (11).

   }**End while**.
4. **While**(There are 2 regions to fuse) **do** {
   - $M \leftarrow M-1$ and update the mixture parameters.
   - Re-initialize the level set functions.
   - **While** (The level sets have not converged) **do** {
     - Propagate the level set functions using Eq. (7).
     - Update the mixture parameters using Eqs. (8),(9),(10) and (11).

}

End while
}**End while**.

The implementation of the curve evolution is performed using the narrow-band algorithm (Adalsteinsson and Sethian, 1995). In this algorithm, the level set distance function is calculated only on a band surrounding the region contour, which is re-initialized whenever the zero level set reaches the limit of the band. In our algorithm, the re-initialization of the level set functions is carried out along with the updating of the mixture parameters since the new delineation of the region contours is known at this step (i.e., after narrow-band re-initialization). The convergence of the level set functions is detected in the algorithm when the contours cease to evolve by Eq. (7). Convergence of the statistical parameters is detected when the distance between the parameters resulting from two successive updating iterations ℓ and $\ell+1$ is less than a certain threshold ϵ, i.e., $\|\Theta^{(\ell)}-\Theta^{(\ell+1)}\|<\epsilon$, or a maximum number of iterations is reached.

4. Experiments

In our experiments, we tested the performance of the proposed segmentation method using real-world images containing texture regions. For all the segmentations, the size of the seeds in the region initialization was fixed to (9×9) pixels and the inter-seed distance was set to 3 pixels. Moreover, for all the segmentation examples, we set $\alpha=\beta=0.5$. It is to note, however, that for the seed size, for example, it is desirable to set as much as possible of seeds in order to cover all the homogenous parts of the image. The setting of α and β has been obtained experimentally, using several images with different texture content. These values yielded, in general, a good compromise between region homogeneity, formulated by minimizing the Bayes error classification, and good boundary localization, formulated using the polarity information. Finally, we note that texture features and polarity information for each image are calculated in a pre-processing step, prior to curve evolution. The pixel neighborhood size used for calculating the correlogram texture features was, also, set experimentally to 32×32 pixels. Fig. 4 shows a typical example of region initialization and curve evolution using the above parameter setting.

In Fig. 5, we show some examples of image segmentation obtained using the proposed method. For each example, we show the result of region initialization, and the results of curve evolution using the initial number of regions and after region fusion (i.e., after adapting the number of regions). Below each resultant segmentation, we show number of regions with which it was obtained. We can note that updating the number of regions did in fact enhance the accuracy of segmentation in almost all the examples. See for instance the first and last examples, where the bodies of the animals were initially over-segmented, and then the segmentation was corrected after the number of regions was updated. The over-segmentation in the first example was due to the non-uniform color of the hair on the left of the animal body; whereas in the last example, it was due to the self-shadowing on the right of the animal body.

To show the performance of the proposed approach, we also ran the segmentation using two recent methods proposed in (Carson et al., 2002; Mansoursi et al., 2006). Color–texture images are segmented into regions (blobs) in an unsupervised fashion in (Carson et al., 2002), by first smoothing texture regions, then fitting a mixture to the smoothed image histogram by means of maximum likelihood estimation. An effective partition of the image is built in (Mansoursi et al., 2006) using active contours, where an explicit correspondence is established between the contours and the regions of the image. Furthermore, the region data are described by a multivariate Gaussian distributions. We note, however, that in both of the methods in (Carson et al., 2002; Mansoursi et al., 2006), only region information is used to steer the segmentation algorithms.

Fig. 6 shows the segmentation of the images in Fig. 5 using the above-mentioned methods; we also show the

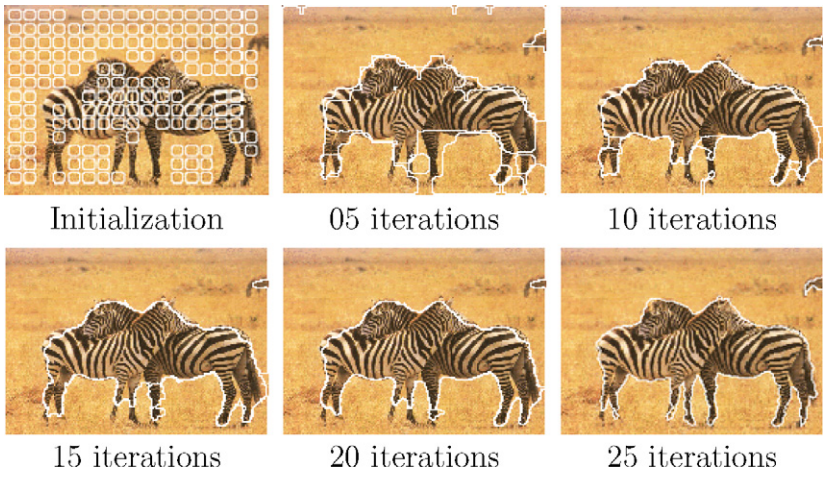

Fig. 4. Example of curve evolution using Eqs. (7)–(10) for updating the mixture parameters.

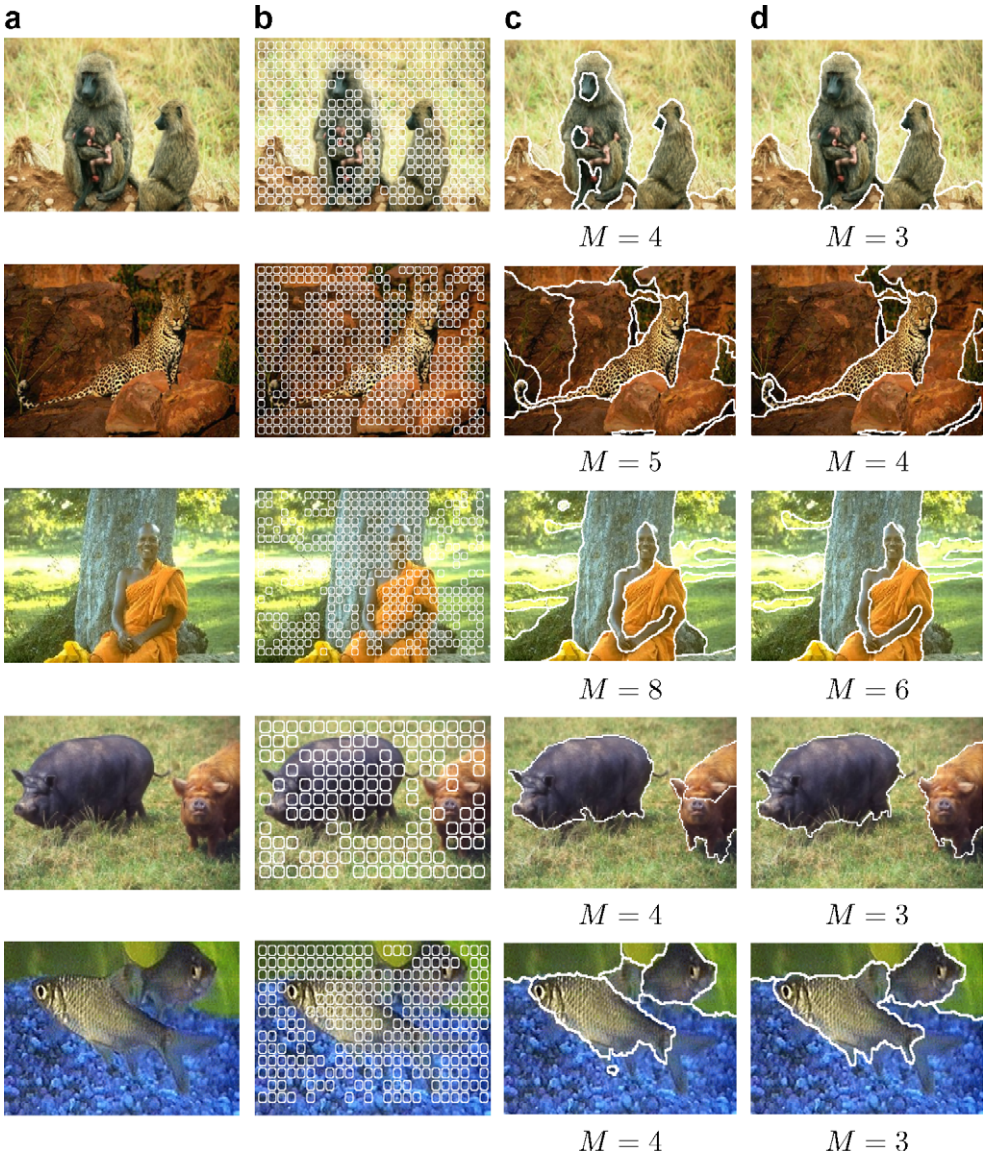

Fig. 5. Examples of image segmentation using the proposed method: (a) represents the original image, (b) and (c) represent, respectively, the region initialization and the result of curve evolution using the initial number of regions. Finally, (d) represents the final segmentation after region fusion.

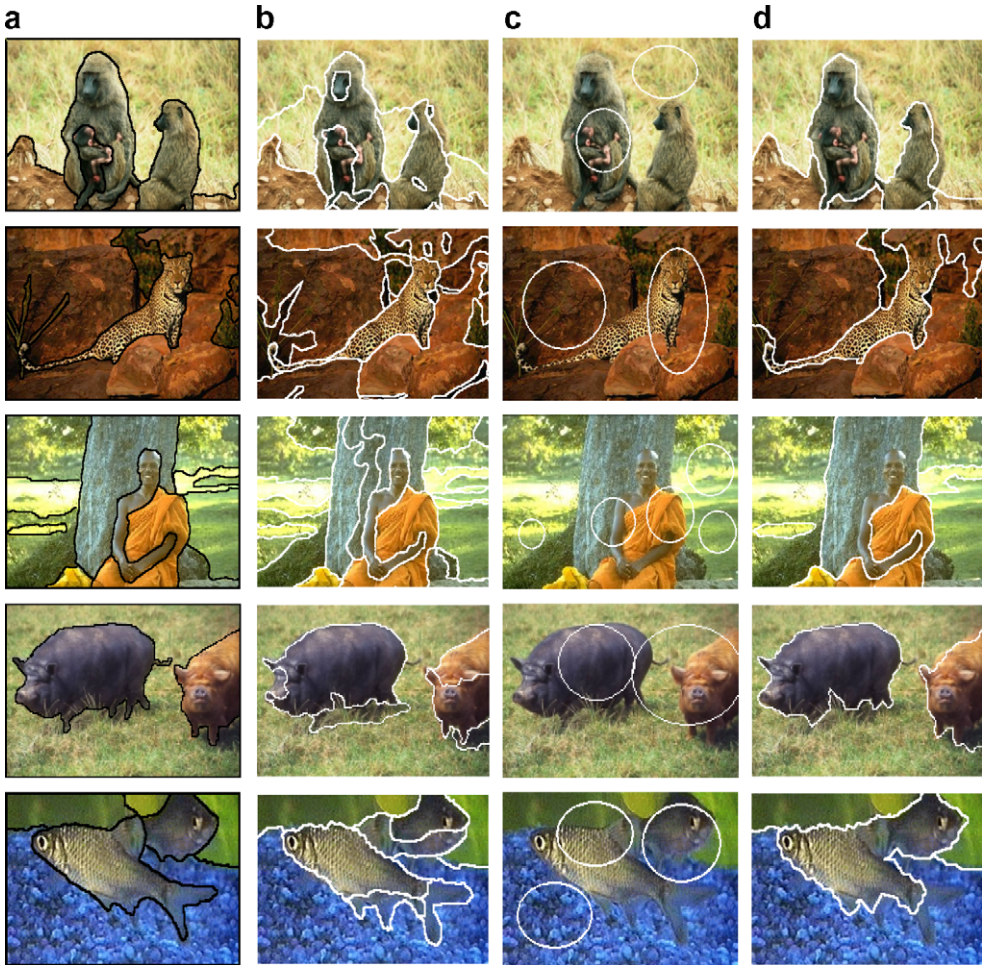

Fig. 6. Performance comparison of the proposed segmentation with recent methods: (a) represents the ground truth, (b) represents the segmentation obtained by the method in (Carson et al., 2002), (c) and (d) represent, respectively, the initialization and the final segmentation obtained by the method in (Mansoursi et al., 2006).

ground-truth, which is a segmentation performed manually on the images. We note that for the method in (Mansoursi et al., 2006), we used the same region features as our approach and we set the number of regions equal to the one of the ground truth. We perform the initialization of that method using $M-1$ contours (see Fig. 6), which correspond to $M-1$ regions plus the background. In Fig. 7, we show the graphs of evolution of energy functional (6) according to the number of iterations. The graphs are generated for all the images in Fig. 5, using our algorithm, run without changing the number of regions and with the change, and the method in (Mansoursi et al., 2006). Two main observations can be made regarding the graphs of the figure. First, for all the examples, running our method with dynamic number of regions decreased the energy functional below the value achieved without using the dynamic number of regions or by the method in (Mansoursi et al., 2006). This generally yielded optimal segmentations with regard to boundary and region information. Second, in comparison to Mansoursi et al. (2006), our method converges faster, since we use a region initialization that is nearer the the optimal solution. The same argument holds for the initial energy value, which is greater for Mansoursi et al. (2006) than for our method.

To measure the accuracy of segmentation of our method, two objective criteria are used. Before defining these criteria, let us first denote by S_i, $i=1,\ldots,K$ and S'_j, $j=1,\ldots,K'$ the sets of segments composing the segmentation of a tested method and the ground truth. A segment, here, is a connected set of pixels delineated by a closed contour. The first objective criterion measures the accuracy of boundary localization and is defined as follows (Martin et al., 2001): Let $S_i(\boldsymbol{x})$ and $S'_j(\boldsymbol{x})$ be the segments that contain the pixel $\boldsymbol{x}$ in a tested segmentation (TS) and in the ground truth (GT), respectively. The error is given by

$$\epsilon_1 = \frac{1}{L}\sum_{x} \min[\epsilon_{\boldsymbol{x}}(\mathrm{TS},\mathrm{GT}), \epsilon_{\boldsymbol{x}}(\mathrm{GT},\mathrm{TS})] \tag{13}$$

with $\epsilon_{\boldsymbol{x}}(\mathrm{TS},\mathrm{GT}) = \frac{|S_i - S'_j|}{|S_i|}$, $\epsilon_{\boldsymbol{x}}(\mathrm{GT},\mathrm{TS}) = \frac{|S'_j - S_i|}{|S_j|}$, where the symbols "$|\cdot|$" and "$|-|$" denote, respectively, the set cardinality and set intersection. Finally, L is the number of

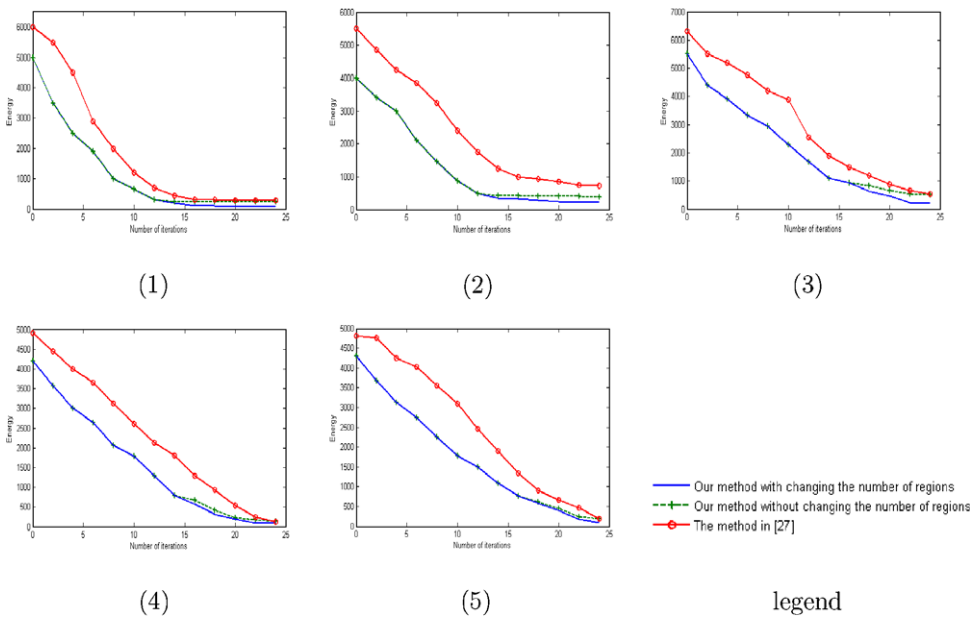

Fig. 7. Graphs of energy evolution for the segmentation of the images in Fig. 5 using our method, run with and without changing the number of regions, and the method in (Mansoursi et al., 2006).

pixels in the image. The second criterion measures the over-segmentation of an image. A set of segments in the TS image $S_{m1},\ldots,S_{m\ell}$, where $2 \leqslant \ell \leqslant K$, are considered as an instance of over-segmentation of a segment S'_m in the GT *iff* (Hoover et al., 1996): $\forall i \in \{1,\ldots,\ell\} : |S_{mi} \cap S'_m| \geqslant k|S_{mi}|$ and $\sum_{i=1}^{x} |S_{mi} \cap S'_m| \geqslant k|R'_m|$, where k is a threshold that we set here to 0.75 as suggested in (Martin et al., 2001). We define the error ϵ_2 as the number of segments in the GT over-segmented in the TS, and vice versa.

In Table 1, we show the values of the above criteria for the examples presented in Figs. 5 and 6. For the accuracy of boundary localization (i.e., error ϵ_1), of the three tested methods our approach yielded the least error for all the examples. This performance can be justified by two main factors. The first factor concerns the algorithm of segmentation itself, whose initialization step captures the main salient regions of the image (i.e., the homogeneous seeds yield a good approximate for the optimal solution). This makes a big difference with the method in (Mansoursi et al., 2006), which aims to obtain a partition of the image, but with less regard for the alignment of the final contours with the real region boundaries. Further, the updating of the number of regions allows for refining the segmentation by explicitly using the value of the energy functional as a criterion. Finally, since the functional contains boundary information, it penalizes contours not aligned with pixels having high boundary plausibility (see especially the examples in rows 2, 4 and 5 in Fig. 5 where these contours were eliminated). For the over-segmentation error, our method and the method in (Mansoursi et al., 2006) yielded smaller values than the one in (Carson et al., 2002). Indeed, since the last method performs the segmentation directly on the smoothed image, a lot of blobs with homogeneous color were created inside the salient regions and their boundaries, which over-segmented the image. We should note, however, that among the tested methods, the one in (Carson et al., 2002) is the fastest, since it does't make use of curve evolution. For the shown examples, the execution time varied between 4 to 7 seconds for our method, 5–9 s for the method in (Mansoursi et al., 2006), and only 1–3 s for the method in (Carson et al., 2002).

Table 1
Values of the different criteria measuring the performance of segmentation where M1, M2 and M3 designate, respectively, our method, the method in (Carson et al., 2002) and the method in (Mansoursi et al., 2006)

Image	ϵ_1			ϵ_2		
	M1	M2	M3	M1	M2	M3
1 (179 × 268)	0.10	0.25	0.20	1	4	1
2 (196 × 300)	0.11	0.19	0.24	3	6	4
3 (200 × 300)	0.15	0.20	0.32	2	5	3
4 (120 × 183)	0.09	0.18	0.11	0	3	0
5 (142 × 244)	0.13	0.45	0.27	1	4	2

5. Conclusions

We have proposed a new framework for unsupervised color–texture segmentation using active contours. The method permits for segmentation of images containing color–texture with an arbitrary number of regions, in a fully automatic fashion. It combines boundary and adaptive region information to steer the curve evolution toward the real region boundaries. The performance of the method was demonstrated on real-world image segmentation, with comparison to two recent methods. In the future, we intend to investigate applying this method to the segmentation of large collections of natural images.

Acknowledgements

The completion of this research was made possible thanks to the Natural Sciences and Engineering Research Council of Canada (NSERC) and Bell Canada's support through its Bell University Laboratories R&D programs. We thank the reviewers for their helpful comments.

Appendix

In what follows, we give the formulas used to calculate the correlogram features. Consider a correlogram matrix obtained for a displacement υ and orientation ϕ. The energy (EN), entropy (ET), inverse-difference-moment (IDM) and correlation (C) of the matrix are calculated as follows:

$$\mathrm{EN}(\upsilon,\phi)=\sum_{c_i,c_j}\left(C^{\upsilon,\phi}(c_i;c_j)\right)^2 \tag{14}$$

$$\mathrm{ET}(\upsilon,\phi)=\sum_{c_i,c_j}-C^{\upsilon,\phi}(c_i;c_j)\log\left(C^{\upsilon,\phi}(c_i;c_j)\right) \tag{15}$$

$$\mathrm{IDM}(\upsilon,\phi)=\sum_{c_i,c_j}\frac{1}{\left(1+\|c_i-c_j\|^2\right)}C^{\upsilon,\phi}(c_i;c_j) \tag{16}$$

$$C(\upsilon,\phi)=\sum_{c_i,c_j}\frac{(c_i-M_x)(c_j-M_y)^{\mathrm{T}}}{|\Sigma_x||\Sigma_y|}C^{\upsilon,\phi}(c_i;c_j) \tag{17}$$

where we have

$$M_x(\upsilon,\phi)=\sum_{c_i}c_i\sum_{c_j}C^{\upsilon,\phi}(c_i;c_j),$$

$$M_y(\upsilon,\phi)=\sum_{c_j}c_j\sum_{c_i}C^{\upsilon,\phi}(c_i;c_j),$$

$$\Sigma_x(\upsilon,\phi)=\sum_{c_i}(c_i-M_x)^T(c_i-M_x)\sum_{c_j}C^{\upsilon,\phi}(c_i;c_j)$$

$$\text{and}\quad \Sigma_y(\upsilon,\phi)=\sum_{c_j}(c_j-M_y)^{\mathrm{T}}(c_j-M_y)\sum_{c_i}C^{\upsilon,\phi}(c_i;c_j)$$

with $|\Sigma_x|$ (respectively $|\Sigma_y|$) denotes the determinant of the matrix Σ_x (respectively Σ_y). Roughly speaking, EN describes the uniformity of a texture. ET measures the randomness of the elements of the correlogram. A homogeneous texture will have a low entropy and vice versa. IDM has relatively a high value when the high values of the correlogram are near the diagonal. Thus, IDM increases when there are small regions in the texture with the same color, which could be taken as a measure of the coarseness of the texture. Finally, C measures the correlation between the different elements of the correlogram. Thus, C increases when the texture is more complex.

References

Adalsteinsson, D., Sethian, J., 1995. A fast level set method for propagating surfaces. J. Comput. Phys. 118 (2), 269–277.

Akaike, H., 1974. A new look at the statistical model identification. IEEE Trans. Automat. Contr. 19 (6), 716–723.

Allili, M.S., Ziou, D., 2005. An automatic segmentation combining mixture analysis and adaptive region information: A level set approach. IEEE Can. Conf. Comput. Robot Vis., 73–80.

Allili, M.S., Bouguila, N., Ziou, D., 2007. Finite generalized Gaussian mixture modeling and applications to image and video foreground segmentation. IEEE Can. Conf. Comput. Robot Vis., 183–190.

Baxter, R.A., Olivier, J.J., 2000. Finding overlapping components with MML. Statist. Comput. 10 (1), 5–16.

Ben Ayed, I., Mitiche, A., Belhadj, Z., 2005. Multiregion level set partitioning on synthetic aperture radar images. IEEE Trans. Pattern Anal. Mach. Intell. 27 (5), 793–800.

Carson, C., Belongie, S., Greenspan, H., Malik, J., 2002. Blobworld: Image segmentation using expectation–maximization and its application to image querying. IEEE Trans. Pattern Anal. Mach. Intell. 24 (8), 1026–1038.

Caselles, V., Kimmel, R., Sapiro, G., 1997. Geodesic active contours. Internat. J. Comput. Vis. 22 (1), 61–79.

Chakraborty, A., Duncan, J.S., 1999. Game-theoretic integration for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 21 (1), 12–30.

Chan, T., Vese, L., 2001. Active contours without edges. IEEE Trans. Image Process. 10 (2), 266–277.

Cohen, L.D., 1991. On active contour models and balloons. CVGIP: Image Understand. 53 (2), 211–218.

Deng, Y., Manjunath, B.S., 2001. Unsupervised segmentation of color–texture regions in images and video. IEEE Trans. Pattern Anal. Mach. Intell. 23 (8), 800–810.

Drewniok, C., 1994. Multispectral edge detection: some experiments on data from landsat-TM. Internat. J. Remote Sens. 15 (18), 3743–3765.

Duda, R.O., Hart, P.E., Stork, D.G., 2000. Pattern Classification. John Wiley.

Freixenet, J., Muñoz, X., Martí, J., Lladó, X., 2004. Colour texture segmentation by region-boundary cooperation. Euro. Conf. Comput. Vis. LNCS 3022, 250–261.

Gimel'farb, G., 2001. Supervised texture segmentation by maximising conditional likelihood. Energy Min. Meth. Comput. Vis. Pattern Recogn. LNCS 2134, 169–184.

Hofmann, T., Puzicha, J., Buhmann, J.M., 1998. Unsupervised texture segmentation in a deterministic annealing framework. IEEE Trans. Pattern Anal. Mach. Intell. 20 (8), 803–818.

Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P.J., Bunke, H., Goldgof, D.B., Bowyer, K., Eggert, D.W., Fitzgibbon, A., Fisher, R.B., 1996. An experimental comparison of range image segmentation algorithms. IEEE Trans. Pattern Anal. Mach. Intell. 18 (7), 673–689.

Huang, J., Kumar, S.R., Mitra, M., Zhu, W.-J., Zabih, R., 1997. Image indexing using color correlograms. IEEE Conf. Comput. Vis. Pattern Recogn., 762–768.

Jain, A., Farrokhnia, F., 1991. Unsupervised texture segmentation using Gabor filters. Pattern Recogn. 24 (12), 1167–1186.

Kervrann, C., Heitz, F., 1995. A markov random field model-based approach to unsupervised texture segmentation using local and global spatial statistics. IEEE Trans. Image Process. 4 (6), 856–862.

Laine, A., Fan, J., 1993. Texture classification by Wavelet Packet signatures. IEEE Trans. Pattern Anal. Mach. Intell. 15 (11), 1186–1191.

Liapis, S., Sifakis, E., Tziritas, G., 2004. Color and texture segmentation using wavelet frame analysis deterministic relaxation and fast marching algorithms. J. Vis. Commun. Image Rep. 15 (1), 1–26.

Lucchese, L., Mitra, S.K., 2001. Color image segmentation: A state-of-the-art survey. Image Process. Vis. Pattern Recog. Proc. Indian Nat. Sci. Acad. (INSA-A) 67 (2), 207–221.

Malik, J., Belongie, S., Leung, T., Shi, J., 2001. Contour and texture analysis for image segmentation. Internat. J. Comput. Vis. 43 (1), 7–27.

Manjunath, B.S., Chellappa, R., 1991. Unsupervised texture segmentation using Markov random field models. IEEE Trans. Pattern Anal. Mach. Intell. 13 (5), 478–482.

Mansoursi, A., Mitiche, A., Vásquez, C., 2006. Multiregion competition: A level set extension of region competition to multiple region image partitioning. Comput. Vis. Image Understand. 101 (3), 137–150.

Martin, D., Fowlkes, C., Tal, D., Malik, J., 2001. A database of human segmented natural Images and its application to evaluating segmentation algorithms and measuring ecological statistics. IEEE Internat. Conf. Comput. Vis. II, 416–423.

McLachlan, G., Peel, D., 2000. Finite mixture models. Wiley Series in Probability and Statistics.

Osher, S., Sethian, J., 1998. Fronts propagating with curvature dependant speed: Algorithms based on Hamilton–Jacobi formulation. J. Comput. Phys. 79 (1), 12–49.

Paragios, N., Deriche, R., 2002a. Geodesic active regions: A new paradigm to deal with frame partition problems in computer vision. J. Vis. Commun. Image Rep. 13 (1–2), 249–268.

Paragios, N., Deriche, R., 2002b. Geodesic active regions and level set methods for supervised texture segmentation. Internat. J. Comput. Vis. 46 (3), 223–247.

Rissanen, J., 1978. Modeling by shortest data description. Automatica 14 (5), 465–471.

Rousson, M., Brox, T., Deriche, R., 2003. Active unsupervised texture segmentation on a diffusion based feature space. IEEE Conf. Comput. Vis. Pattern Recogn. II, 699–704.

Sagiv, C., Sochen, N.A., Zeevi, Y.Y., 2006. Integrated active contours for texture segmentation. IEEE Trans. Image Process. 15 (6), 1633–1646.

Sifakis, E., Garcia, C., Tziritas, G., 2002. Bayesian level sets for image segmentation. J. Vis. Commun. Image Rep. 13 (1), 44–64.

Talbar, S.N., Holambe, R.S., Sontakke, T.R., 1998. Supervised texture segmentation using wavelet frames. Internat. Conf. Signal Process. II, 1177–1180.

Unal, G., Yezzi, A., Krim, H., 2005. Information-theoretic active polygons for unsupervised texture segmentation. Internat. J. Comput. Vis. 62 (3), 199–220.

Yezzi, A.J., Kichenassamy, S., Kumar, A., Olver, P., Tannenbaum, A., 1997. A Geometric snake model for segmentation of medical imagery. IEEE Trans. Med. Imag. 16 (2), 199–209.

Yezzi, A., Tsai, A., Willsky, A.S., 2002. A fully global approach to image segmentation via couples curve evolution equations. J. Vis. Commun. Image Rep. 13 (1–2), 195–216.

Zhu, S., Yuille, A., 1996. Region competition: Unifying snakes region growing and Bayes/MDL for multiband image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 18 (9), 884–900.