A unifying approach and interface for spline-based snakes

Mathews Jacob, Thierry Blu and Michael Unser

Biomedical Imaging Group, DMT/IOA
Swiss Federal Institute of Technology, Lausanne
CH-1015 Lausanne EPFL, SWITZERLAND
Mathews.Jacob, Thierry.Blu, Michael.Unser@epfl.ch

ABSTRACT

In this paper, we present different solutions for improving spline-based snakes. First, we demonstrate their minimum curvature interpolation property, and use it as an argument to get rid of the explicit smoothness constraint. We also propose a new external energy obtained by integrating a non-linearly pre-processed image in the closed region bounded by the curve. We show that this energy, besides being efficiently computable, is sufficiently general to include the widely used gradient-based schemes, Bayesian schemes, their combinations and discriminant-based approaches. We also introduce two initialization modes and the appropriate constraint energies. We use these ideas to develop a general snake algorithm to track boundaries of closed objects, with a user-friendly interface.

Keywords: - snake, active-contours, green's theorem, interface

1. INTRODUCTION

The estimation of boundaries of objects in images is a key problem in medical imaging applications. After the introduction of the snake algorithm by Kass et al,[1] a lot of research was focussed on solving the boundary detection problem using this approach. Snakes or active contour models, as conceived in[1] are ordered collections of points (represented in a piecewise constant spline basis) that have an explicit internal energy — a term that forces the curve to be continuos and smooth. However, the curve can be represented more efficiently in other bases (like finite elements, Fourier basis,[2,3] B-splines,[4,5] etc.) Such a representation, besides being more compact, brings in an implicit smoothness to the curve.

In this paper, we focus on spline snakes. We will provide new arguments that support the choice of cubic B-splines as a natural basis. We will also propose enhancements to make the snake algorithm more efficient. As the B-spline curve is the minimum curvature curve interpolating the knot points, we eliminate the explicit internal energy. Hence the dynamical problem is converted into a simpler parameter estimation problem.

In the past few years, many researchers have proposed different image energies that bring the snake close to the desired contour. We will draw on many of these ideas and propose a unifying framework for the image energy. As a reasonable compromise, we choose a combination of gradient and region terms to yield a snake that is precise and less sensitive to noise and to starting conditions.

We choose two different external constraint energies, depending of two different user input schemes. The first one is a shape input mode where the external constraint energy restricts the permissible shapes. It will force the snake to be close to some reference template. In the second mode the user may place some points. Here the external constraint energy acts like a spring that pulls the snake towards the points that are specified.

The paper is organized as follows. In the next section, we give the preliminaries such as parametric curve representation and Green's theorem. In section 3, we explain how we get rid of the internal energy by representing the curve in a B-spline basis. In section 4, we concentrate on the external energy. Here we introduce a simple cost function, which accommodates a wide class of external energies used in practice. We also give two external constraint energies adapted to the corresponding initialization modes. In section 5, we deal with the optimization algorithm.

Medical Imaging 2001: Image Processing, Milan Sonka, Kenneth M. Hanson, Editors,
Proceedings of SPIE Vol. 4322 (2001) © 2001 SPIE · 1605-7422/01/$15.00

2. PRELIMINARIES

2.1. Parametric curve representation

A curve in the $x - y$ plane can be represented in terms of an arbitrary parameter t as $\mathbf{r}(t) = (x(t), y(t))$. If the curve is closed, the function vector $\mathbf{r}(t)$ is periodic.

When the curve $\mathcal{C}$ is represented as above, the function $\mathbf{r}(t)$ can be approximated efficiently as linear combinations of some basis functions, which makes the representation compact and easy to handle. Among such approximations, the scaling function representation is the most widely used; it ranges from Fourier curves to description as a polygon. The scaling function representation of a curve is given by

$$\mathbf{r}(t) = \sum_{k=-\infty}^{\infty} \mathbf{c}_k \varphi(t-k) \tag{1}$$

Here $\mathbf{c}_k$ denotes the coefficient vector. If the period, M, is an integer, we have $\mathbf{c}_k = \mathbf{c}_{k+M}$. This reduces the infinite summations to

$$\mathbf{r}(t) = \sum_{k=0}^{M-1} \mathbf{c}_k \varphi_p(t-k), \tag{2}$$

where

$$\varphi_p(t) = \sum_{k=-\infty}^{\infty} \varphi(t - k \cdot M) \tag{3}$$

Here, we consider the parametric curve representation where φ is a B-spline as in Brigger et al.[4]

2.2. Green's theorem

Green's theorem relates the volume integral of the divergence of a vector field in a closed region to the surface integral of the field; its restriction to two dimensions can be written as

$$\int_{\mathcal{S}} \left(\frac{\partial \mathbf{F}_x}{\partial x} + \frac{\partial \mathbf{F}_y}{\partial y}\right) dx\,dy = \oint_{\mathcal{C}} (\mathbf{F}_y dx - \mathbf{F}_x dy) \tag{4}$$

The first integral is evaluated over the area $\mathcal{S}$ enclosed by the curve and the second one along the curve $\mathcal{C}$. Using this theorem, surface integrals can be computed efficiently as curve integrals. For example, the integral

$$\int_{\mathcal{S}} f(x,y)\,dx\,dy = -\oint_{\mathcal{C}} g(x,y)\,dy, \tag{5}$$

where $g(x,y) = \int_{-\infty}^{x} f(\tau, y)\,d\tau$.

3. INTERNAL ENERGY

The internal energy enforces the smoothness and continuity of the conventional snakes, where the curve is an ordered collection of points.[1] However, if the curve is represented in a cubic B-spline basis, this term is no more required. This is due to the minimum curvature interpolation property of the cubic B-spline curves, which we discuss in the next subsection. In addition to this property, the B-splines have compact support which enables the local control of the contour. Moreover, the cubic B-spline is known to have good approximation properties and there exist efficient algorithms for their processing.[6]

3.1. Minimum curvature interpolation

The minimum curvature curve satisfying the interpolation constraints (enforced by the user input points) is the cubic B-spline interpolant, provided it is described in the curvilinear abscissa. For the proof, see Appendix A.

The smoothness of the snake is dependent on the number of knot points used in its representation. This property enables us to get rid of the explicit smoothness constraint (internal energy term), which makes the optimization simpler. Here we make the assumption that the snake does not deviate a lot from the curvilinear abscissa during the optimization process.

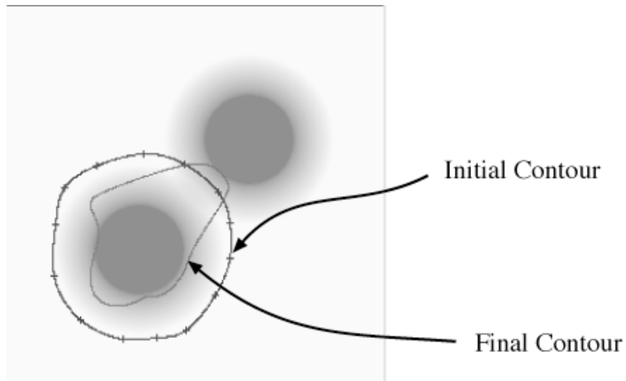


Figure 1. Example of gradient based snake getting misguided by a nearby object.

4. EXTERNAL ENERGY

As we got rid of the internal energy, fitting the snake is simplified; it can be reformulated as a parameter estimation problem. We may also use the well known statistical formulation[3] and estimate the snake parameters by maximizing the corresponding likelihood function. Under the assumption that all images are equally probable, we rewrite the expression for the likelihood function using the Bayes rule as

$$M(\mathcal{C}|f) = \log\left(P\left(\mathcal{C}|f\right)\right) \quad = \quad \underbrace{\log\left(P\left(f|\mathcal{C}\right)\right)}_{E_{\text{image}}} + \underbrace{\log\left(P\left(\mathcal{C}\right)\right)}_{E_{\text{constraint}}}, \tag{6}$$

where $\mathcal{C}$ denotes the snake curve and f denotes the image. The first term on the right hand side corresponds to the image energy and is responsible for guiding the snake to the desired image features. It is dependent only on the image data and is denoted by E_{image}. The second term is the external constraint energy and represent the prior knowledge of the shape. It will be denoted as $E_{\text{constraint}}$. We will discuss each of them in detail.

4.1. Image energy.

We introduce the generic form of the energy function

$$E^{c}_{\text{image}} = \int_{\mathcal{S}} g(x,y)\,dxdy, \tag{7}$$

where $g = Tf$ is a transformed version of the image f and $\mathcal{S}$ is the closed region bounded by the curve. We will see that this simple cost function can accommodate a large class of widely used image energies including gradient schemes,[1,7] Bayesian schemes[8,9] and their combinations.[3] We now show how the conventional schemes fall into this generalized framework with an appropriate choice of the operator T.

4.1.1. Gradient-based image energy

The most widely used image energy is gradient based and is given by[1]

$$E^{g}_{\text{image}} = \oint_{\mathcal{C}} |\nabla f(\mathbf{r})|^2 dr \tag{8}$$

Here f is the intensity of the image and ∇ denotes the gradient operator. But as this energy takes advantage of the magnitude of the gradient only, it can misguide the snake to nearby object as shown in Fig.1.

This problem can be eliminated by using the fact that we are trying to track a closed region. When the bounding curve is scanned clockwise, the desired region is always on the right. Hence we expect the gradient to be oriented towards the right of the curve (assuming that we are tracking a brighter region). An image energy that makes use of this property is

$$E^{g}_{\text{image}} = \oint_{\mathcal{C}} \mathbf{k}\cdot\left(\nabla f(\mathbf{r}) \times d\mathbf{r}\right), \tag{9}$$

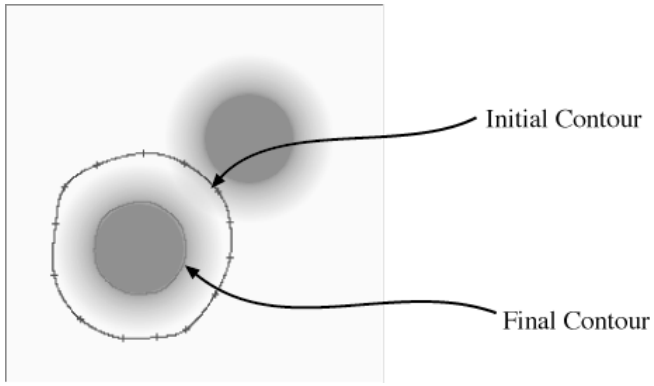


Figure 2. How the use of the directed gradient energy (9) instead of (8) avoids the pitfall in Fig. 1.

where $\mathbf{k}$ denotes the unit vector perpendicular to the $x - y$ plane. It is seen from Fig.2 that this energy guides the snake close to the desired object. A scheme similar to this one, where the snake is also penalized for having an orientation inconsistent with the gradient, was proposed by Fok et. al.[7]

E^g_{image} can be also rewritten as a surface integral (cf. Appendix B) as

$$E^g_{\text{image}} = \oint_{\mathcal{C}} \mathbf{k} \cdot (\nabla f(\mathbf{r}) \times d\mathbf{r}) = -\int_{\mathcal{S}} \underbrace{\nabla^2 f(\mathbf{s})}_{T_g(f)} d\mathcal{S}, \tag{10}$$

where ∇^2 denotes the laplacian operator and $\mathcal{S}$ denotes the area bounded by the curve $\mathcal{C}$. Using this result, this gradient energy is written in the form of (7) with $g = -\nabla^2 f$.

As this term has sharp maxima at the gradient boundaries, the contour estimated using this energy will be precise. The main drawback of gradient-based schemes is their inability to lock on contours that are too far from the snake's initial position. This is simply because the gradients vanish as we get farther from the contour. Moreover, the snake has difficulty moving into concavities and is sensitive to noise.

4.1.2. Region-based image energy

Using region-based image energies can eliminate this problem of lack of convergence away from the contour. Such cost functions are less sensitive to noise and can track concavities better. These properties are due to the fact that a region based criterion considers the whole image f, as compared to the gradient energy which uses only the values of f close to the snake.

Assuming the image pixels to be independent random variables, we use the statistical formulation of Staib and Duncan[3] to specify the region likelihood function:

$$E^r_{\text{region}} = \int_{\mathcal{S}} \log\left(P\left(f(\mathbf{s})\,|\,\mathbf{s} \in \mathcal{S}\right)\right) d\mathbf{s} \; + \; \int_{\mathcal{S}'} \log\left(P\left(f(\mathbf{s})\,|\,\mathbf{s} \in \mathcal{S}'\right)\right) d\mathbf{s}, \tag{11}$$

where $\mathcal{S}$ and $\mathcal{S}'$ are the regions inside and outside the snake respectively. We transform this equation to the form of (7) by rewriting it as

$$E^r_{\text{region}} = \int_{\mathcal{S}} \log\left(P\left(f(\mathbf{s})\,|\,\mathbf{s} \in \mathcal{S}\right)\right) d\mathbf{s} \; + \; \left(C - \int_{\mathcal{S}} \log\left(P\left(f(\mathbf{s})\,|\,\mathbf{s} \in \mathcal{S}'\right)\right) d\mathbf{s}\right), \tag{12}$$

where the constant $C = \int_{\mathcal{S} \cup \mathcal{S}'} \log\left(P\left(f(\mathbf{s})\,|\,\mathbf{s} \in \mathcal{S}'\right)\right) d\mathbf{s}$. As C does not depend on the position of the snake, we take it out of the cost function.

$$E^r_{\text{image}} = \int_{\mathcal{S}} \underbrace{\log\left(\frac{P\left(f(\mathbf{r})|\mathbf{s} \in \mathcal{S}\right)}{P\left(f(\mathbf{s})|\mathbf{s} \in \mathcal{S}'\right)}\right)}_{T_r(f)} d\mathbf{s} \tag{13}$$

The region based term has the ability to bring the snake close to the actual contour when it is initially far from it. Hence the final snake is less dependent on the initialization. Moreover, it also guides the snake into boundary concavities and hence is an alternative to the gradient vector flow approach.[10] As we are using the whole function f rather than its values close to the boundaries, this term is more robust to noise. However, it is less precise than the gradient based scheme, because the peak of the cost function near the actual contour is not as sharp (due to the averaging effect).

4.1.3. Combined image energy

Combining what we have seen so far, we consider the generalized energy function (7), with $g = \underbrace{(\alpha T_g + \beta T_r)}_{T_c}(f)$. As we combine the two types of cost functions, we get a powerful class of image energies, which inherits the advantages of both schemes. The addition of the laplacian enhances the edges and hence improves the precision of the estimated contour. This also justifies the use of edge enhancement as a preprocessing step for segmentation.

4.2. External constraint energy.

As mentioned above, the external constraint energy is dependent on the apriori knowledge of the shape (parameters). This information may be obtained from the user input or from the previous knowledge of the shape. We choose two different external constraint energies depending on the initialization mode.

4.2.1. Initialization

The user has the choice between two initialization modes, whichever is most appropriate for his application.

1. Shape Input

   This procedure is valid as long as the shape of an object is relatively well preserved from one instance to the other. In this case, we keep a reference template of the shape which is parametrized as well. The user adjusts the template by translating and resizing the initial shape so that it fits approximately the object of interest.

   In such applications, the common shape transformation is affine in nature. Hence, we can penalize the snake for not being an affine transformation of the reference template. This external constraint energy term can be written as

$$E_{\text{constraint}} = \min_{\mathbf{A},b} \frac{1}{\sigma^2} \sum_{k=0}^{N-1} \left|\mathbf{c}_k - \mathbf{A}\mathbf{c}_{\text{ref},k} - \mathbf{b}\right|^2, \qquad (14)$$

   where $\mathbf{A}$ is a 2×2 affine matrix and $\mathbf{b}$ is the translation vector. Here $\mathbf{c}$ is the sequence of B-spline coefficient vectors of the snake and $\mathbf{c}_{\text{ref}}$ is the coefficient vector of the template. Our criterion (14) is essentially equivalent to measuring the L_2 distance between the snake and the reference curve because the B-spline representation is affine invariant and also because the B-splines form a Riesz basis. This shape constraint is a cheaper alternative to Cootes and Taylor method of eigen-shapes,[11] which demands intensive training.

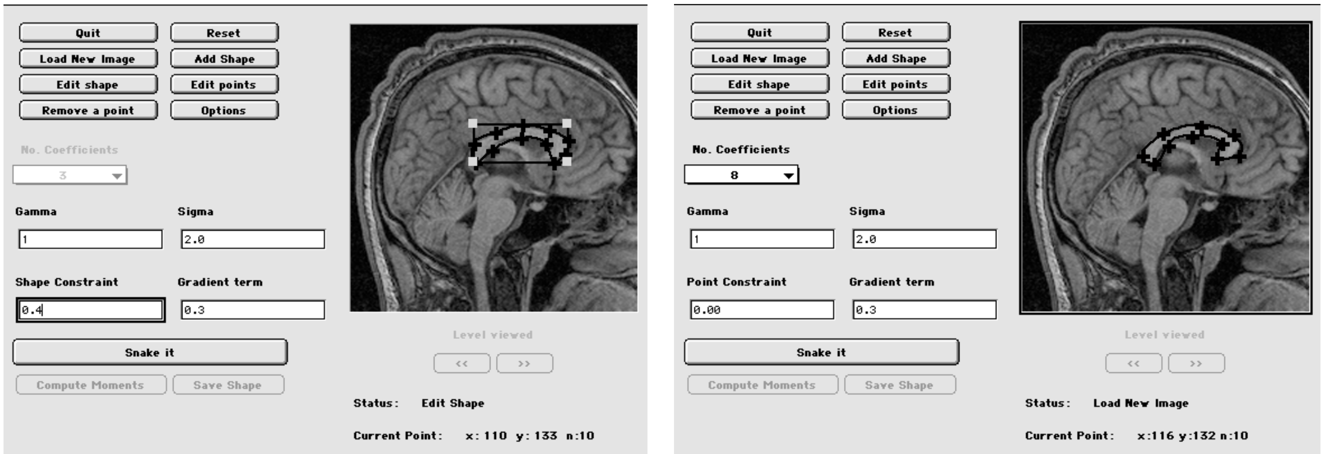
   In the iteration loop, the affine parameters are also to be optimized. We resort to a two step scheme; the affine parameters are assumed to be a constant and the coefficients are optimized in the initial stage. In the second step, we solve for the optimal affine parameters that map $\mathbf{c}$ to $\mathbf{c}_{\text{ref}}$.

2. Point Input

   The user inputs some points on the boundaries of the object. We fit a spline with the user specified number of knot points to the sampled data. The curve is then reparametrized to be closer to the curvilinear abscissa and the fit is performed again. This procedure is iterated until we get a fit reasonably close to the curvilinear abscissa. By specifying the number of knot points, the user can control the smoothness of the curve. The number of knot points can also be chosen using the minimum description length criterion.[12]

   After the reparametrization, the i^{th} input point correspond to some value of the parameter t_i. We assume that the curve samples $\mathbf{r}_i = \mathbf{r}(t_i)$ are randomly distributed with the mean as the i^{th} input point. If the distribution is Gaussian with a uniform variance of σ, the external constraint energy is written as

$$E_{\text{constraint}} = \frac{1}{\sigma^2} \sum_{k=0}^{N-1} \left|\mathbf{r}_k - \mathbf{r}_{\text{init},k}\right|^2$$



1) Shape input mode. 2) Point Input mode

Figure 3. In the shape input mode, the user adjusts the bounding box of the reference shape to fit the object approximately. In the point input mode, a curve with a specified number of points are fit to the user placed points.

Here, $\mathbf{r}_{\text{init}}$ and $\mathbf{r}$ are the input point vectors and the corresponding curve sample vectors respectively. In this case the curve is penalized for not passing through the input points. The force acts like a spring that pulls the snake towards the desired points.

5. OPTIMIZATION ALGORITHM

The external energy (7) and its derivatives are computed efficiently as a line integral using Green's theorem as (4). We use a conjugate-gradient algorithm for optimization. To speed up the method and to make it more robust, we perform the iteration in a multiresolution framework. The algorithm gives an approximate fit to a low resolution image. This fit is then used as the initialization for the next finer resolution image. The process is repeated until one reaches the finest level. The number of levels are chosen depending on the relative size of the initialized region. This scheme makes the algorithm less sensitive to noise and starting conditions.

Our general version of the spline-based snake algorithm has been implemented in C with both types of initialization modes. It is currently being applied to different types of biomedical images and the results are quite encouraging.

6. CONCLUSION

We have presented the minimum curvature interpolation property of the cubic splines as a strong argument in favor of the B-spline snakes. The main point is that the spline curves are smooth by construction and there is no need to include the traditional elasticity terms as the snake's internal energy. Hence, we obtain a natural family of parametric snakes which are unambiguously defined in terms of their B-spline coefficients.

Thanks to this representation, we can formulate the snake algorithm as a parameter estimation problem. We have introduced a general, yet simple form of external energy function and have shown that it can accommodate a large class of cost functions. As a good compromise, we have proposed a cost function that is a combination of region and gradient based energies. We have also shown how to compute and optimize it efficiently — the key idea is to use Green's theorem to convert a surface integral into a curve integral. Finally, we have introduced a simple way of constraining the snake to be close to some reference shape.

By combining these various techniques, we are able to improve the traditional snake algorithms. The method that we propose is quite robust and relatively insensitive to starting conditions.

Acknowledgements

This work was supported by the Swiss National Science Foundation under grant 2100-053540.

Appendix - A: Minimum curvature interpolation

Consider the curvature of the curve at a point $(x(t), y(t))$ which is given as.

$$C(x,y) = \frac{x'y'' - x''y'}{\left(x'^2 + y'^2\right)^{\frac{3}{2}}}$$

If the parameter t is the curvilinear abscissa, ${x'}^2 + {y'}^2$ is a constant for all t. Hence the square of the curvature can be written as

$$\begin{aligned} |C(x,y)|^2 &= K_1|x'y'' - x''y'|^2 \\ &= K_1(\mathbf{r}' \times \mathbf{r}'') \cdot (\mathbf{r}' \times \mathbf{r}''), \end{aligned}$$

where $\mathbf{r} = (x, y)$. Making use of the vector identity $\mathbf{a}.(b \times c) = c.(a \times b)$, this can be rewritten as

$$\begin{aligned} (\mathbf{r}' \times \mathbf{r}'') \cdot (\mathbf{r}' \times \mathbf{r}'') &= \mathbf{r}'' \cdot (\mathbf{r}' \times \mathbf{r}'' \times \mathbf{r}'') \\ &= \mathbf{r}'' \cdot (\mathbf{r}''(\mathbf{r}' \cdot \mathbf{r}') - \mathbf{r}'(\mathbf{r}' \cdot \mathbf{r}'')) \\ &= |\mathbf{r}''|^2|\mathbf{r}'|^2 - |\underbrace{\mathbf{r}'' \cdot \mathbf{r}'}_{d(\mathbf{r}'^2)=0}|^2 \end{aligned}$$

For the second step, we make use of the identity $\mathbf{a} \times b \times c = (a \cdot c)b - (b \cdot c)a$. The second term in the last expression is zero as it is the derivative of ${\mathbf{r}'}^2$, which is a constant when the curve is described in the curvilinear abscissa. So the expression for the curvature can be written as

$$\begin{aligned} |C(\mathbf{r})|^2 &= K|\mathbf{r}''|^2 \\ &= K\left[{x''}^2 + {y''}^2\right], \end{aligned}$$

where K is some constant. Minimization of this condition along with the interpolation constraints is a well known problem and the solution is the cubic B-spline interpolation.

Appendix - B: Expressing the gradient-based energy as a surface integral

We consider the energy term

$$E^g_{\text{image}} = \oint_{\mathcal{C}} \mathbf{k} \cdot (\nabla f(\mathbf{r}) \times d\mathbf{r}) = -\oint_{\mathcal{C}} d\mathbf{r} \cdot (\mathbf{k} \times \nabla f(\mathbf{r})) \quad (15)$$

Using Green's theorem (4), we transform this line integral into a surface integral as

$$E^g_{\text{image}} = -\int_{\mathcal{S}} \nabla \times (\mathbf{k} \times \nabla f(\mathbf{r})) \cdot d\mathbf{s}, \quad (16)$$

where $\mathcal{S}$ is the region bounded by the curve $\mathcal{C}$ and $d\mathbf{s}$ is the elemental area vector oriented in the direction of $\mathbf{k}$. Using the standard vector identities, the above equation is reduced to

$$E^g_{\text{image}} = -\int_{\mathcal{S}} (\nabla^2 f)\, \mathbf{k} \cdot d\mathbf{s} = -\int_{\mathcal{S}} (\nabla^2 f)\, d\mathcal{S}, \quad (17)$$

where $d\mathcal{S}$ denotes the magnitude of $d\mathbf{s}$

REFERENCES

1. M. Kass, A. Witkin, and D. Terzopoulos, "Snakes:active contour models," *International Journal of Computer Vision*, vol. 1, pp. 321–331, 1988.
2. A. Chakraborthy, 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, pp. 859–870, 1996.

3. L.H. Staib and J.S. Duncan, "Boundary finding with parametrically deformable models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, pp. 1061–1075, 1992.
4. P. Brigger, J. Hoeg, and M. Unser, "B-spline snakes: a flexible tool for parametric contour detection," *IEEE Trans. Image Process.*, vol. 9, pp. 1484–1496, September 2000.
5. M. Flickner, H. Sawhney, D. Pryor, and J. Lotspeich, "Intelligent interactive image outlining using spline snakes," in *28th Asilomar Conference on Signals, Systems and Computers*, 1994.
6. M. Unser, "Splines: a perfect fit for signal and image processing," *IEEE Signal Processing Magazine*, vol. 16, pp. 22–38, 1999.
7. Y.L.Fok, J.C.K.Chan, and R.T.Chin, "Automated analysis of nerve-cell images using active contour models," *IEEE Trans. Med. Imag.*, vol. 15, pp. 353–367, 1996.
8. Geir Storvik, "A Bayesian approach to dynamic contours through schocastic sampling and simulated annealing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, pp. 976–986, October 1994.
9. C. Chesnaud, P. Refregier, and V. Booulet, "Statistical region snake-based segmentation adapted to different physical noise models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, pp. 1145–1157, November 1999.
10. C. Xu and J.L. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Trans. Image Process.*, vol. 7, pp. 359–369, 1998.
11. T.F.Cootes, C.J.Taylor, and J.Haslam, "Use of active shape models for locating structures in medical images," *Image and vision computing*, vol. 12, July/August 1994.
12. M.A.T. Figueiredo, J.M.N. Leitao, and A.K.Jain, "Unsupervised contour representation and estimation using B-splines and a minimum description length criterion," *IEEE Trans. Image Process.*, vol. 9, pp. 1075–1087, 2000.