

An Application of Active Contour Models to Head Boundary Location

J B Waite and W J Welsh

Image Processing Research Group
British Telecom Research Laboratories
Martlesham Heath
Ipswich IP5 7RE

Locating the boundary of a head in a head-and-shoulders image is an important problem in model based coding. An approach to this problem using adaptive contour models or 'snakes' is presented. The paper provides a tutorial introduction to the theory of snakes and shows in some detail how they may be implemented using a finite difference method. Some experimental results are presented showing snakes locating the head in a set of test images.

Edge Detection

A classical problem in image processing is the detection, location and description of the edges or boundaries of objects in an image. Classical edge detection algorithms provide information on edges in the form of an 'edge image' [1]. In general they work by setting the grey level of each pixel in the 'edge image' to a value that is dependent on the magnitude of the gradient of the grey level at the corresponding point in the original image. The processing from this class of edge detector is purely local. Such an edge detector has no concept of an edge (Eg that edges are continuous, they tend to be smooth almost everywhere) and just flags points where the gradient is high, be they edges or noise.

The concept of an edge is far more than the presence of a high gradient at a particular location; whether an edge is present or not depends on the spatial distribution of the gradient highs and lows, and this is information that the edge detector does not explicitly possess. However, when an edge-detected image is viewed the edges are clearly visible. This is because our visual system is able to 'post-process' the image and provide us with the continuity information that is inherent in edges. Note that this information is present in the edge image even though the detector does not extract it.

A model-based coding technique has been developed which requires conformation of a wire-frame model to the head of a subject in a head and shoulders image [2], hence location of the head boundary is of considerable importance. Traditional

edge detectors in general will not form an edge that is completely closed (ie forms a loop around the head) but will instead create a number of edge segments that taken together outline the boundary of the head. A method is required that processes the edge segments and generates a smooth extension that describes the head boundary. Snakes appear to show promise in this direction. The remainder of the paper describes the theory of snakes and shows in detail how they may be implemented. Section 7 shows some results of applying snakes to the problem of finding head boundaries.

Snakes

Introduced by Kass et al [3], snakes are a method of attempting to provide some of the post-processing that our own visual system performs. A snake has built into it various properties that are associated with both edges and the human visual system (Eg continuity, smoothness and to some extent the capability to fill in sections of an edge that have been occluded).

A snake is a continuous curve (possibly closed) that attempts to dynamically* position itself from a given starting position in such a way that it 'clings' to edges in the image. The form of snake that will be considered here consists of curves that are piecewise polynomial. That is, the curve is in general constructed from N segments $\{x_i(s), y_i(s)\} i = 1, \dots, N$ where each of the $x_i(s)$ and $y_i(s)$ are polynomials in the parameter s . As the parameter s is varied a curve is traced out.

Snake Properties

From now on snakes will be referred to as the parametric curve $\underline{u}(s) = (x(s), y(s))$ where s is assumed to vary between 0 and 1. What properties should an 'edge hugging' snake have?

* The use of the term 'dynamically' is strictly speaking incorrect. The way that snakes have been programmed in the past make them appear as if they are moving in time, but this is just a product of the implementation. Snakes are solutions to static problems; time does not enter into the formulation.

- (a) The snake must be 'driven' by the image. That is, it must be able to detect an edge in the image and align itself with the edge. One way of achieving this is to try to position the snake such that the average 'edge strength' (however that may be measured) along the length of the snake is maximised. If the measure of edge strength is $F(x, y) \geq 0$ at the image point (x, y) then this amounts to saying that the snake $\underline{u}(s)$ is to be chosen in such a way that the functional

$$\int_{s=0}^{s=1} F(x(s), y(s)) ds \quad \dots(1)$$

is maximised. This will ensure that the snake will tend to mould itself to edges in the image if it finds them, but does not guarantee that it will find them in the first place. Given an image the functional may have many local minima (a static problem); finding them is where the 'dynamics' arises. An edge detector applied to an image will tend to produce an edge map consisting of mainly thin edges. This means that the edge strength function tends to be zero at most places in the image, apart from on a few lines. As a consequence a snake placed some distance from an edge may not be attracted towards the edge because the edge strength is effectively zero at the snakes initial position. To help the snake come under the influence of an edge the edge image is blurred to broaden the width of the edges.

- (b) If an elastic band were held around a convex object and then let go, the band would contract until the object prevented it from doing so further. At this point the band would be moulded to the object, thus describing the boundary. Two forces are at work here; firstly that providing the natural tendency of the band to contract, and secondly the opposing force provided by the object. The band contracts because it tries to minimise its elastic energy due to stretching. If the band were described by the parametric curve $\underline{u}(s) = (x(s), y(s))$ then the elastic energy at any point s is proportional to

$$\left(\frac{d\underline{u}}{ds} \Big|_s \right)^2 = \left(\frac{dx}{ds} \Big|_s \right)^2 + \left(\frac{dy}{ds} \Big|_s \right)^2.$$

That is, the energy is proportional to the square of how much the curve is being stretched at that point. The elastic band will take up a configuration so that the elastic energy along its entire length, given the constraint of the object, is minimised. Hence the elastic band assumes the shape of the curve $\underline{u}(s) = (x(s), y(s))$ where $\underline{u}(s)$ minimises the functional

$$\int_{s=0}^{s=1} \left\{ \left(\frac{dx}{ds} \right)^2 + \left(\frac{dy}{ds} \right)^2 \right\} ds \quad \dots(2)$$

subject to the constraints of the object. We would like closed snakes to have analogous behaviour. That is, to

have a tendency to contract, but to be prevented from doing so by the objects in an image. To model this behaviour the parametric curve for the snake is chosen so that the functional (2) tends to be minimised. If in addition the forcing term (1) were included then the snake would be prevented from contracting 'through objects' as it would be attracted toward their edges. The attractive force would also tend to pull the snake into the hollows of a concave boundary, provided that the restoring 'elastic force' was not too great.

- (c) One of the properties of edges that is difficult to model is their behaviour when they can no longer be seen. If we were looking at a car and a person stood in front of it, few of us would have any difficulty imagining the contours of the edge of the car that were occluded. They would be 'smooth' extensions of the contours either side of the person. If the above elastic band approach were adopted it would be found that the band formed a straight line where the car was occluded (because it tries to minimise energy, and thus length in this situation). If however the band had some stiffness (that is a resistance to bending, as for example displayed by a flexible bar) then it would tend to form a smooth curve in the occluded region of the image and be tangential to the boundaries on either side (figure 1).

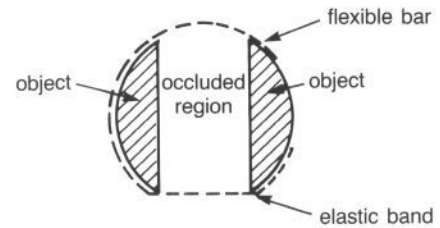


Figure 1 - Snake interpolating across an occlusion.

Again a flexible bar tends to form a shape so that its elastic energy is minimised. The elastic energy in bending is dependent on the curvature of the bar, that is the second derivatives. To help force the snake to emulate this type of behaviour the parametric curve $\underline{u}(s) = (x(s), y(s))$ is chosen so that it tends to minimise the functional

$$\int_{s=0}^{s=1} \left\{ \left(\frac{d^2x}{ds^2} \right)^2 + \left(\frac{d^2y}{ds^2} \right)^2 \right\} ds \quad \dots(3)$$

which represents a pseudo-bending energy term. Of course, if a snake were made too stiff then it would be difficult to force it to conform to highly curved boundaries under the action of the forcing term (1).

Three desirable properties of snakes have now been identified. To incorporate all three into the snake at once the parametric curve $\underline{u}(s) = (x(s), y(s))$ representing the snake is chosen so that it minimises the functional

$$I(x(s), y(s)) = \int_{s=0}^{s=1} \left\{ \alpha(s) \left\{ \left(\frac{d^2x}{ds^2} \right)^2 + \left(\frac{d^2y}{ds^2} \right)^2 \right\} + \beta(s) \left\{ \left(\frac{dx}{ds} \right)^2 + \left(\frac{dy}{ds} \right)^2 \right\} - F(x(s), y(s)) \right\} ds \quad \dots(4)$$

Here the terms $\alpha(s) > 0$ and $\beta(s) \geq 0$ represent respectively the amount of stiffness and elasticity that the snake is to have. It is clear that if the snake approach is to be successful then the correct balance of these parameters is crucial. Too much stiffness and the snake will not correctly hug the boundaries; too much elasticity and closed snakes will be pulled across boundaries and contract to a point or may even break away from boundaries at concave regions. The negative sign in front of the forcing term is because minimising $-\int F(x, y) ds$ is equivalent to maximising $\int F(x, y) ds$.

As it stands, minimising the functional (4) is trivial. If the snake is not closed then the solution degenerates into a single point $(x(s), y(s)) = \text{constant}$, where the point is chosen to minimise the edge strength $F(x(s), y(s))$. Physically, this is because the snake will tend to pull its two end points together in order to minimise the elastic energy, and thus shrink to a single point. The global minimum is attained at the point in the image where the edge strength is largest. To prevent this from occurring it is necessary to fix the positions of the ends of the snake in some way. That is, 'boundary conditions' are required. It turns out to be necessary to fix more than just the location of the end points and two further conditions are required for a well posed problem. A convenient condition is to impose zero curvature at each end point.

Similarly, the global minimum for a closed-loop snake occurs when it contracts to a single point. However, in contrast to an fixed-end snake, additional boundary conditions cannot be applied to eliminate the degenerate solution. The degenerate solution in this case is the true global minimum.

Formulation

Clearly the ideal situation is to seek a local minimum in the locality of the initial position of the snake. In practice the problem that is solved is weaker than this: Find a curve $\underline{u}(s) = (\hat{x}(s), \hat{y}(s)) \in H^2[0, 1] \times H^2[0, 1]$ such that

$$\left. \frac{\partial I(\hat{u}(s) + \epsilon v(s))}{\partial \epsilon} \right|_{\epsilon=0} = 0 \quad v(s) \in H_0^2[0, 1] \times H_0^2[0, 1] \quad \dots(5)$$

Here $H^2[0, 1]$ denotes the class of real valued functions defined on $[0, 1]$ that have 'finite energy' in the second derivatives (that is the integral of the square of the second derivatives exists

[4]) and $H_0^2[0, 1]$ is the class of functions in $H^2[0, 1]$ that are zero at $s = 0$ and $s = 1$. To see how this relates to finding a minimum consider $\hat{u}(s)$ to be a local minimum and $\hat{u}(s) + \epsilon v(s)$ to be a perturbation about the minimum that satisfies the same boundary conditions (ie $v(0) = v(1) = 0$). Clearly, considered as a function of ϵ , $I(\epsilon) = I(\hat{u}(s) + \epsilon v(s))$ is a minimum at $\epsilon = 0$. Hence the derivative of $I(\epsilon)$ must be zero at $\epsilon = 0$. Equation (5) is therefore a *necessary* condition for a local minimum. Although solutions to (5) are not guaranteed to be minima for completely general edge strength functions [6], it has been found in practice that solutions are indeed minima.

Standard arguments in the calculus of variations [6] show that problem (5) is equivalent to another problem, which is simpler to solve:

Find a curve $(\hat{x}(s), \hat{y}(s)) \in C^4[0, 1] \times C^4[0, 1]$ that satisfy the pair of fourth order ordinary differential equations

$$-\frac{d^2}{ds^2} \left\{ \alpha(s) \frac{d^2 \hat{x}}{ds^2} \right\} + \frac{d}{ds} \left\{ \beta(s) \frac{d \hat{x}}{ds} \right\} + \frac{1}{2} \frac{\partial F}{\partial x} \Big|_{(\hat{x}, \hat{y})} = 0 \quad \dots(6)$$

$$-\frac{d^2}{ds^2} \left\{ \alpha(s) \frac{d^2 \hat{y}}{ds^2} \right\} + \frac{d}{ds} \left\{ \beta(s) \frac{d \hat{y}}{ds} \right\} + \frac{1}{2} \frac{\partial F}{\partial y} \Big|_{(\hat{x}, \hat{y})} = 0 \quad \dots(7)$$

together with the boundary conditions

$\hat{x}(0), \hat{y}(0), \hat{x}(1), \hat{y}(1)$ given, and

$$\left. \frac{d^2 \hat{x}}{ds^2} \right|_{s=0} = \left. \frac{d^2 \hat{y}}{ds^2} \right|_{s=0} = \left. \frac{d^2 \hat{x}}{ds^2} \right|_{s=1} = \left. \frac{d^2 \hat{y}}{ds^2} \right|_{s=1} = 0 \quad \dots(8)$$

The statement of the problem is for the case of a fixed-end snake, but if the snake is to form a closed loop then the boundary conditions above are replaced by periodicity conditions. Both of these problem can easily be solved using finite differences; a description of the approach is given in the following section.

Finite Differences

The finite difference approach starts by discretising the interval $[0, 1]$ into $N - 1$ equispaced subintervals of length $h = \frac{1}{N-1}$ and defines a set of *nodes* $\{s_i\}_{i=1}^{i=N}$ where $s_i = (i - 1)h$.

The method seeks a set of approximations $\{(x_i, y_i)\}_{i=1}^{i=N}$ to $\{(x(s_i), y(s_i))\}_{i=1}^{i=N}$ by replacing the differential equations (6) and (7) in the continuous variables with a set of difference equations in the discrete variables [4]. Replacing the derivatives in (6) by difference approximations at the point s_i gives

$$\begin{aligned}
& -\frac{1}{h^2} \left\{ \alpha_{i+1} \frac{(x_{i+2} - 2x_{i+1} + x_i)}{h^2} - 2\alpha_i \frac{(x_{i+1} - 2x_i + x_{i-1})}{h^2} \right. \\
& \quad \left. + \alpha_{i-1} \frac{(x_i - 2x_{i-1} + x_{i-2})}{h^2} \right\} \\
& + \frac{1}{h} \left\{ \beta_{i+1} \frac{(x_{i+1} - x_i)}{h} - \beta_i \frac{(x_i - x_{i-1})}{h} \right\} \\
& + \frac{1}{2} \frac{\partial F}{\partial x} \bigg|_{(x_i, y_i)} = 0 \quad \text{for } i = 3, 4, \dots, N-2
\end{aligned} \tag{9}$$

where $\alpha_i = \alpha(s_i)$ and $\beta_i = \beta(s_i)$. Similarly a difference approximation to (7) may be derived. Note that the difference equation only holds at internal nodes in the interval where the indices referenced lie in the range 1 to N . Collecting like terms together, (9) can be written as

$$a_i x_{i-2} + b_i x_{i-1} + c_i x_i + d_i x_{i+1} + e_i x_{i+2} = f_i$$

where

$$a_i = -\frac{\alpha_{i-1}}{h^4}$$

$$b_i = \frac{2\alpha_i}{h^4} + \frac{2\alpha_{i-1}}{h^4} + \frac{\beta_i}{h^2}$$

$$c_i = -\left\{ \frac{\alpha_{i+1}}{h^4} + \frac{4\alpha_i}{h^4} + \frac{\alpha_{i-1}}{h^4} + \frac{\beta_{i+1}}{h^2} + \frac{\beta_i}{h^2} \right\}$$

$$d_i = \frac{2\alpha_{i+1}}{h^4} + \frac{2\alpha_i}{h^4} + \frac{\beta_{i+1}}{h^2}$$

$$e_i = -\frac{\alpha_{i+1}}{h^4}$$

$$f_i = -\frac{1}{2} \frac{\partial F}{\partial x} \bigg|_{(x_i, y_i)}$$

Discretising both the differential equations (6) and (7) and taking boundary conditions into account the finite difference approximations $\underline{x} = \{x_i\}$ and $\underline{y} = \{y_i\}$ to $\{x(s_i)\}$ and $\{y(s_i)\}$ respectively satisfy the following system of algebraic equations

$$K\underline{x} = \underline{f}(x, y), \quad K\underline{y} = \underline{g}(x, y) \quad \dots(10)$$

The structure of the matrices K and the right hand vectors \underline{f} and \underline{g} are different depending on whether closed or open snake boundary conditions are used. If the snake is closed then fictitious nodes at s_0, s_{-1}, s_{N+1} and s_{N+2} are introduced and the difference equation (9) is applied at nodes 0, 1, $N-1$ and N .

Periodicity implies that $x_0 = x_N, x_{-1} = x_{N-1}, x_{N+1} = x_1$ and $x_2 = x_{N+2}$. With these conditions in force the coefficient matrix becomes

$$K = \begin{pmatrix} c_1 & d_1 & e_1 & & & & a_1 & b_1 \\ b_2 & c_2 & d_2 & e_2 & & & & a_2 \\ a_3 & b_3 & c_3 & d_3 & e_3 & & & \\ & a_4 & b_4 & c_4 & d_4 & e_4 & & \\ & & \cdot & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & & \\ e_{N-1} & & & & a_{N-1} & b_{N-1} & c_{N-1} & d_{N-1} \\ d_N & e_N & & & & a_N & b_N & c_N \end{pmatrix}$$

and the right hand side vector is

$$(f_1, f_2, \dots, f_N)^T$$

For fixed-end snakes fictitious nodes at s_0 and s_{N+1} are introduced and the difference equation (9) is applied at nodes s_1 and s_{N+1} . Two extra difference equations are introduced to approximate the zero curvature boundary conditions $\frac{d^2x}{ds^2} \bigg|_{s_1} = \frac{d^2x}{ds^2} \bigg|_{s_N} = 0$, namely $x_0 - 2x_1 + x_2 = 0$ and

$x_{N-1} - 2x_N + x_{N+1} = 0$. The coefficient matrix is now

$$K = \begin{pmatrix} c_2 - a_2 & d_2 & e_2 & & & & & & & \\ b_3 & c_3 & d_3 & e_3 & & & & & & \\ a_4 & b_4 & c_4 & d_4 & e_4 & & & & & \\ & a_5 & b_5 & c_5 & d_5 & e_5 & & & & \\ & & \cdot & \cdot & \cdot & \cdot & \cdot & & & \\ & & & \cdot & \cdot & \cdot & \cdot & \cdot & & \\ & & & & a_{N-2} & b_{N-2} & c_{N-2} & d_{N-2} & & \\ & & & & & a_{N-1} & b_{N-1} & c_{N-1} & -e_{N-1} & \end{pmatrix}$$

and the right hand side vector is

$$(f_2 - (2a_2 + b_2)x_1, f_3 - a_3x_1, f_4, \dots, f_{N-3}, f_{N-2} - e_{N-2}x_N, f_{N-1} - (2e_{N-1} + d_{N-1})x_N)^T$$

The right hand side vector for the difference equations corresponding to (7) is derived in a similar fashion.

System Solution

The system (10) represents a set of non-linear equations that has to be solved. The coefficient matrix is symmetric and positive definite, and banded for the fixed-end snake. For a closed-loop snake with periodic boundary conditions it is

banded, apart from a few off-diagonal entries. As the system is non-linear it is solved iteratively. The iteration performed is

$$\frac{(x_{n+1} - x_n)}{\gamma} + Kx_{n+1} = f(x_n, y_n) \text{ for } n = 0, 1, 2, \dots$$

$$\frac{(y_{n+1} - y_n)}{\gamma} + Ky_{n+1} = g(x_n, y_n) \text{ for } n = 0, 1, 2, \dots$$

where $\gamma > 0$ is a stabilisation parameter. This can be rewritten as

$$\left(K + \frac{1}{\gamma}I\right)x_{n+1} = \frac{1}{\gamma}x_n + f(x_n, y_n) \text{ for } n = 0, 1, 2, \dots$$

$$\left(K + \frac{1}{\gamma}I\right)y_{n+1} = \frac{1}{\gamma}y_n + g(x_n, y_n) \text{ for } n = 0, 1, 2, \dots$$

This system has to be solved for each n . For a closed-loop snake the matrix on the left hand side is difficult to invert directly because the terms that are outside the main diagonal band destroy the band structure. In general the coefficient matrix K can be split into the sum of a banded matrix B plus a non-banded matrix A ; $K = A + B$. For a fixed-end snake the matrix A would be zero. The system of equations is now solved for each n by performing the iteration

$$\left(B + \frac{1}{\gamma}I\right)x_{n+1}^{(k+1)} = -Ax_{n+1}^{(k)} + \frac{1}{\gamma}x_n + f(x_n, y_n)$$

for $k = 0, 1, 2, \dots$

$$\left(B + \frac{1}{\gamma}I\right)y_{n+1}^{(k+1)} = -Ay_{n+1}^{(k)} + \frac{1}{\gamma}y_n + g(x_n, y_n)$$

for $k = 0, 1, 2, \dots$

The matrix $\left(B + \frac{1}{\gamma}I\right)$ is a band matrix and can be expressed as a product of Cholesky [5] factors LL^T . The systems are solved at each stage by first solving

$$L\bar{x}_{n+1}^{(k+1)} = -Ax_{n+1}^{(k)} + \frac{1}{\gamma}x_n + f(x_n, y_n)$$

$$L\bar{y}_{n+1}^{(k+1)} = -Ay_{n+1}^{(k)} + \frac{1}{\gamma}y_n + g(x_n, y_n)$$

followed by

$$L^T\bar{x}_{n+1}^{(k+1)} = \bar{x}_{n+1}^{(k+1)}$$

$$L^T\bar{y}_{n+1}^{(k+1)} = \bar{y}_{n+1}^{(k+1)}$$

Notice that the Cholesky decomposition only has to be performed once.

Results

A closed loop snake was implemented using the finite difference method and tested on a set of 28 facial images. The images were initially processed using a Laplacian type of operator [2]. The output of the operator is modified by a sigmodal function which suppresses small levels of activity due to noise as well as very strong edges while leaving intermediate values barely changed. By this means, the snake is presented with a smoother edge image which reduces its tendency to oscillate about its equilibrium position. The operation also enhances weak edge contours such as may follow the line of the chin. The snake is initialised surrounding the area of the image in which the head is expected to lie and is then allowed to contract under its own internal elasticity. The snake converged on the head boundary in 15 of the images. In 9 images, it managed to locate most of the boundary but at some point it became trapped on a feature external to the face such as a collar. In the remaining 4 images, the snake penetrated the head boundary and proceeded to contract into the interior of the face. Some methods for overcoming these problems are discussed in [2]. The initial position of the snake for each image tested is shown in figure 2a. Various stages of its contraction including the final equilibrium position are shown in figures 2b,c,d. Figure 2e shows a case where the snake has been trapped.



Fig 2a



Fig 2b

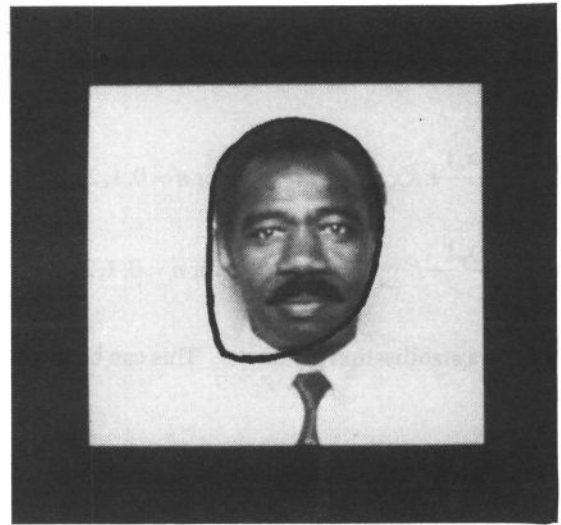


Fig 2e



Fig 2c



Fig 2d

Figures 2a-e - Operation of the snake - a) initial position of snake. b) and c) intermediate stages of contraction. d) equilibrium position of snake. e) case where the snake has been trapped.

References

1. **Pratt W. K.**, *Digital Image Processing*, Wiley-Interscience, 1978.
2. **Welsh W. J., Searby S, Waite J B.** "Model Based Image Coding", *British Telecom Technology Journal*, July 1990.
3. **Kass M., Witkin A., Terpozopoulus D.** "Snakes: Active Contour Models", *International Journal of Computer Vision*, 321-331, 1988.
4. **Keller H. B.**, *Numerical Methods for Two-Point Boundary Value Problems*, Blaisdell, 1968.
5. **Johnson L. W., Riess R. D.**, *Numerical Analysis*, Addison-Wesley, second edition, 1982.
6. **Waite J. B., Welsh W. J.**, "Head Boundary Location Using Snakes", *British Telecom Technology Journal*, July 1990.