Using Lateral Coupled Snakes for Modeling the Contours of Worms

Qing Wang[1,2], Olaf Ronneberger[1,2], Ekkehard Schulze[3], Ralf Baumeister[2,3], and Hans Burkhardt[1,2]

[1] Chair of Pattern Recognition and Image Processing, Department of Computer Science, University of Freiburg
[2] Centre for Biological Signalling Studies (bioss), University of Freiburg
[3] Faculties of Biology and Medicine, FRIAS LIFENET, ZBSA, University of Freiburg
qwang@informatik.uni-freiburg.de

Abstract. A model called lateral coupled snakes is proposed to describe the contours of moving C. elegans worms on 2D images with high accuracy. The model comprises two curves with point correspondence between them. The line linking a corresponding pair is approximately perpendicular to the curves at the two points, which is ensured by shear restoring forces. Experimental proofs reveal that the model is a promising tool for locating and segmenting worms or objects with similar shapes.

1 Introduction

Active contour models (snakes), introduced by Kass et al. in 1987 [1], are dynamic curves that evolve to minimize the energy functional, which is composed of internal and external parts. The internal energy defines the physical property of the curve and serves to impose regularity constraints on the snake. The external energy is calculated from the image and gives rise to forces that push the snake toward the desired image features such as lines and edges.

A snake must be initialized near the real boundary to ensure convergence. To overcome this limitation, Cohen added a pressure force to the model so that it acts like a balloon [2]. Xu and Prince proposed a new type of external forces: the gradient vector flow (GVF), which is computed as a diffusion of the gradient of the edge map derived from the image [3]. GVF snakes are insensitive to initialization and can be attracted to boundary concavities. Neither balloon forces nor GVF can be derived from a potential. The models must now be formulated directly in terms of forces instead of energies.

Snakes are an example of deformable models, the essence of which is to take account of prior information about the objects and the image contents simultaneously. With more (correct) prior knowledge being incorporated, the results can become more reliable and robust. In this paper, we will show a model developed for describing and finding the exact contours of Caenorhabditis elegans. The model will be introduced mainly in the force-balance framework.

The roundworm C. elegans is one of the prime model organisms in medical biology, and is used e.g. for the functional analysis of human disease genes. The

J. Denzler, G. Notni, and H. Süße (Eds.): DAGM 2009, LNCS 5748, pp. 542–551, 2009.

small size and plethora of manipulations possible with this animal allow the setup of large-scale genetic and pharmacological analysis with C. elegans. Automated readouts, e.g., the measurement of behavioral differences of worms being treated are essential for high-throughput experiments. Appropriate description of the shape and position of a worm is crucial towards successful behavior analysis. In the state-of-the-art system for C. elegans movement analysis [4], the position of a worm is represented by the skeleton of its binary image, obtained through thresholding and morphological operations. This approach is simple but has low accuracy and may fail when a worm forms a loop while moving (see Fig. 4).

On 2D microscopic images, the contour of a worm consists of two sides that are almost parallel to each other but meet at the ends. Based on the physical property of the worm body, we have developed a model for describing its contour using its internal coordinates. The model is made up of two curves and each point on one curve corresponds to a point on the other. The line linking a corresponding pair is approximately normal to the two curves at the two points, which is ensured by shear restoring forces. The two curves support and constrain each other through the coupling. We call this model *Lateral Coupled Snakes* (LCS). As will be shown in the experiments, the LCS model yields highly accurate description of worm contours.

A model called ribbon snake [5] has similar applications as the LCS model. It is made up of only one snake with each point associated with a width. It has been applied to road extraction from aerial imagery [6]. Despite the similarity, the two models differ in many ways. Unlike ribbon snakes, with LCS, the external forces act directly on the model points and the regularity of the two sides is directly controlled. The two models therefore often have different behaviors, especially near the ends. There is a mechanism for the LCS model to grow naturally along the object boundary. When extended to 3D, the two models will have different extensions. In addition, we have studied how to employ the LCS model to find the contours of worms from bad starting points. The technique can be used for segmenting other elongated objects with two almost parallel sides. This work, as far as we know, has not been done with the ribbon snake model.

In the very first paper on snakes [1], coupling was already proposed for a pair of stereo snakes for surface construction. The additional energy is defined as a function of the distance of snake contours on the left and right images. In [7], it is proposed to impose epipolar geometry constraints to a pair of stereo snakes to achieve consistency and robustness in stereo tracking. The dual active contour technique [8] was developed to overcome the problems of sensitivity to initialization and parameters. With this technique, one contour expands from inside the target features, the other contracts from the outside. The two contours are interlinked to provide an ability to reject "weak" local energy minima. Hohenhuser and Hommel [9] put these models as well as the original snakes into a general framework and called it coupled active contour. They then used two snakes which are interlinked beyond different image domains (intensity and dense range images, more specifically) for segmentation. Our model also falls largely into this framework. What makes it distinct is the property of the coupling.

2 The Model

The LCS model is made up of two curves $\mathbf{x}_0(s)$ and $\mathbf{x}_1(s)$, $s \in [0,1]$ and

$$\mathbf{x}_k(s) = \big[x_k(s), y_k(s)\big], \qquad k = 0, 1 \ . \tag{1}$$

Optionally one can include the constraints

$$\mathbf{x}_0(0) = \mathbf{x}_1(0) \qquad \text{and} \qquad \mathbf{x}_0(1) = \mathbf{x}_1(1) \tag{2}$$

to make their ends meet. The line linking the coupled pair $\mathbf{x}_0(s)$ and $\mathbf{x}_1(s)$ should be approximately perpendicular to the curves at the two points. See Fig. 1a for illustration.

Each curve is like a traditional snake and has internal energy

$$dE_k(s) = \frac{1}{2}\left(\alpha(s)\|\mathbf{x}_k'(s)\|^2 + \beta(s)\|\mathbf{x}_k''(s)\|^2\right) ds \ , \tag{3}$$

where α and β are weight parameters that control the elasticity and rigidity of the curve. The internal energy (3) gives rise to forces

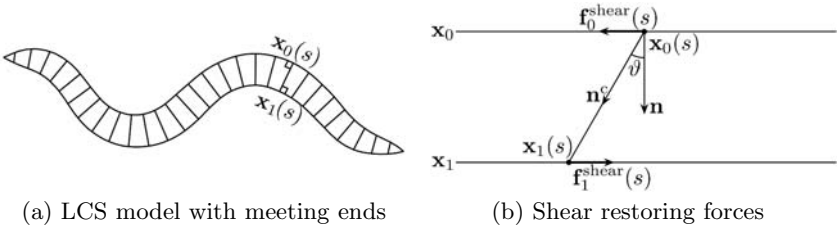
$$\mathbf{f}_k^{\text{elasticity}}(s) = \alpha(s)\mathbf{x}_k''(s) \qquad \text{and} \qquad \mathbf{f}_k^{\text{rigidity}}(s) = -\beta(s)\mathbf{x}_k''''(s) \ . \tag{4}$$

To insure that the linking line is approximately perpendicular to the curves, we introduce *shear restoring forces* to resist shear deformation $\tan\vartheta$ (see Fig. 1b). For simplicity, first assume the two sides are perfectly parallel at $\mathbf{x}_0(s)$ and $\mathbf{x}_1(s)$. Let the unit normal vector be $\mathbf{n}$ and the unit vector pointing from $\mathbf{x}_0(s)$ toward $\mathbf{x}_1(s)$ be $\mathbf{n}^{\text{c}}$, the shear restoring forces can be written as

$$\mathbf{f}_0^{\text{shear}}(s) = \mu(s)\frac{\mathbf{n}^{\text{c}} - (\mathbf{n}^{\text{c}} \cdot \mathbf{n})\mathbf{n}}{\|\mathbf{n}^{\text{c}} \cdot \mathbf{n}\|} \qquad \text{and} \qquad \mathbf{f}_1^{\text{shear}}(s) = -\mathbf{f}_0^{\text{shear}}(s) \ . \tag{5}$$

where μ is the shear modulus. When the two sides are not perfectly parallel, the mean normal direction should be used in place of $\mathbf{n}$.

Above the parameters α, β and μ are written as functions of s for flexibility. For common applications, it is usually enough to assign constant values for the whole model. The same also applies to other weight parameters introduced later.



(a) LCS model with meeting ends (b) Shear restoring forces

Fig. 1.

One can also apply balloon forces to the sides or introduce new types of forces that are defined on coupled pairs. Some examples of the latter are: 1. Forces that keep the width vary smoothly. Define the width at s as $w(s) = \|\mathbf{x}_1(s) - \mathbf{x}_0(s)\|$. The forces are proportional to $w''(s)$ and act along the coupling direction. 2. Repulsion and attraction between a coupled pair. They can be used to control the allowed maximal and minimal widths.

Forces on the ends need to be treated specially, which we defer until section 2.2 after the discrete formulation of the model is given. To deal with the self-occlusion problem as shown in Fig. 3, forces between different parts of the model are necessary to prevent the two tips of the model from occupying the same position. Due to limitation of space, we will not elaborate on them.

2.1 Discrete Formulation

When formulated discretely, curves are given by a set of ordered points. Suppose there are $l+1$ points for each snake. The LCS model is represented by

$$\left\{\mathbf{x}_k(i) \mid i = 0, 1, \cdots, l \text{ and } k = 0, 1\right\} \ . \tag{6}$$

The formulas with the continuous model imply natural parameterization of the curve, which means that the points should be evenly spaced along the curve. For the LCS model, we define

$$d_k(i) = \|\mathbf{x}_k(i) - \mathbf{x}_k(i-1)\| \qquad i = 1, \cdots, l \quad \text{and} \quad k = 0, 1 \ , \tag{7}$$

and require that $\left(d_0(i) + d_1(i)\right)/2 = h$ be constant with regard to i after reparameterization. We call the constant h the *spacing*.

As $d_k(i)$ are not constant, we need to be careful with the elasticity and rigidity forces. To ensure that the parameters α and β can be chosen largely invariant to the spacing, we define

$$\mathbf{a}_k(i) = \frac{\mathbf{x}_k(i) - \mathbf{x}_k(i-1)}{d_k(i)} \qquad i = 1, 2, \cdots, l \tag{8}$$

$$\mathbf{b}_k(i) = \frac{\mathbf{a}_k(i+1) - \mathbf{a}_k(i)}{\left(d_k(i+1) + d_k(i)\right)/2} \qquad i = 1, 2, \cdots, l-1 \ . \tag{9}$$

The elasticity force on point $\mathbf{x}_k(i)$ $(i = 1, 2, \cdots, l-1)$ is formulated as

$$\mathbf{f}_k^{\text{elasticity}}(i) = \frac{1}{h}\left(\alpha(i+1)\mathbf{a}_k(i+1) - \alpha(i)\mathbf{a}_k(i)\right) \tag{10}$$

and the rigidity force on point $\mathbf{x}_k(i)$ $(i = 2, 3, \cdots, l-2)$ is given by

$$\mathbf{f}_k^{\text{rigidity}}(i) = \frac{1}{h^2}\left(2\beta(i)\mathbf{b}_k(i) - \beta(i-1)\mathbf{b}_k(i-1) - \beta(i+1)\mathbf{b}_k(i+1)\right) \ . \tag{11}$$

To calculate the shear restoring forces, we find first the tangential direction of the curves at $\mathbf{x}_k(i)$:

$$\mathbf{T}_k(i) = \boldsymbol{\mathcal{N}}\left(\mathbf{a}_k(i) + \mathbf{a}_k(i+1)\right) \tag{12}$$

where $\mathcal{N}$ is the normalization operator defined as $\mathcal{N}\mathbf{v} \equiv \mathbf{v}/\|\mathbf{v}\|$ for any non-zero vector $\mathbf{v}$. As $\mathbf{T}_0(i)$ usually differs somewhat from $\mathbf{T}_1(i)$, we use their mean direction as the effective tangential direction

$$\mathbf{t}(i) = \mathcal{N}\left(\mathbf{T}_0(i) + \mathbf{T}_1(i)\right) \quad . \tag{13}$$

The effective normal direction $\mathbf{n}(i)$ can be obtained by turning $\mathbf{t}(i)$ by $\pi/2$. The unit vector pointing from $\mathbf{x}_0(i)$ toward $\mathbf{x}_1(i)$ is

$$\mathbf{n}^{\mathrm{c}}(i) = \mathcal{N}\left(\mathbf{x}_1(i) - \mathbf{x}_0(i)\right) \quad . \tag{14}$$

Given $\mathbf{n}(i)$ and $\mathbf{n}^{\mathrm{c}}(i)$ we are able to calculate shear restoring forces using (5).

2.2 Forces on the Ends and Their Neighbors

The forces on the ends and the points next to them need to be treated separately. In the following we only give the forces on $\mathbf{x}_k(0)$ and $\mathbf{x}_k(1)$. Forces on $\mathbf{x}_k(l-1)$ and $\mathbf{x}_k(l)$ can be derived accordingly.

The elasticity forces given in (4) make a snake contract continuously, which is usually undesired but sometimes necessary (see section 2.4). Depending on the situation, the elasticity forces on the ends can be set to 0 or be kept as

$$\mathbf{f}_k^{\text{elasticity}}(0) = \frac{1}{h}\alpha(1)\mathbf{a}_k(1) \quad . \tag{15}$$

As no rigidity energy is associated with the ends, the rigidity forces on $\mathbf{x}_k(1)$ and $\mathbf{x}_k(0)$ are

$$\mathbf{f}_k^{\text{rigidity}}(1) = \frac{1}{h^2}\left(2\beta(1)\mathbf{b}_k(1) - \beta(2)\mathbf{b}_k(2)\right) \tag{16}$$

$$\mathbf{f}_k^{\text{rigidity}}(0) = -\frac{1}{h^2}\beta(1)\mathbf{b}_k(1) \quad . \tag{17}$$

When the ends of the two sides meet, i.e. when $\mathbf{x}_0(0) = \mathbf{x}_1(0)$, the elasticity and rigidity forces applied to an end will be from both sides. For example,

$$\mathbf{f}^{\text{rigidity}}(0) = \mathbf{f}_0^{\text{rigidity}}(0) + \mathbf{f}_1^{\text{rigidity}}(0) \quad . \tag{18}$$

Usually the model near the ends takes a convex form like in Fig. 2. The sum of the rigidity forces on an end thus often has a longitudinal component pointing forward, which enables the model to grow along the object boundary. When this is not desired, the longitudinal component should be set to 0. The lateral component shall always be kept so that the end can be swung into the correct position.

To make the model grow along the boundary of the object, one can also assign a dragging force along the longitudinal direction on the end:

$$\mathbf{f}^{\text{drag}}(0) = -\gamma(0)\,\mathcal{N}\left(\mathbf{a}_0(1) + \mathbf{a}_1(1)\right) \tag{19}$$

where γ is the weight parameter for the dragging forces. It should be small enough compared to the image force parameter (will be given in section 2.3) so that the model can be attracted to the desired positions while growing.

2.3 Evolution and Multiscale Approach

Image forces are calculated from the image. For the experiments in this paper, we use the GVF derived from the gradient magnitude of the image as the image force field. Let the GVF field be $\mathbf{g}(\mathbf{x})$, the external force acting upon a model point is then

$$\mathbf{f}_k^{\text{image}}(i) = \kappa(i)\mathbf{g}(\mathbf{x}_k(i)) \tag{20}$$

where $i = 1, 2, \cdots, l-1$ and κ is the weight parameter for the external forces.

The net force on a model point is the vector sum of all the internal and external forces. Let it be $\mathbf{f}_k(i)$. To enable the model to reach a force balance state, its movement is defined by

$$\dot{\mathbf{x}}_k(i,t) = \mathbf{f}_k(i,t) \tag{21}$$

where $k = 1, 2$ and $i = 0, 1, \cdots, l$. This system of equations can be used directly to update the model. To free ourselves from the problem of adjusting timesteps at development phase, however, we use the ordinary differential equation (ode) solver from the gnu scientific library (gsl) [10] for updating.

It is computationally efficient to take a multiscale approach for the model evolution, with coarser scales for faster convergence followed by finer scales for better localization. The scale of the external force field is controlled by the size of smoothing kernels for calculating edge maps and numbers of iterations for computing GVF. The scale of the model is represented by its spacing h.

2.4 LCS Model for Object Segmentation

The LCS model is more powerful than what it was designed for. Since the two coupled curves support, constrain and provide information to each other, the model is rather robust. It can be used for searching and segmenting worms or objects of similar shapes with initializations far away from the real positions. If it is possible to tell whether an end is clearly outside of the object (the LCS model should make the determination easier), one can follow three steps for segmentation (see Fig. 5 for an example):

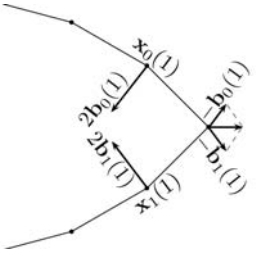


Fig. 2. The model near an end often takes a convex form, enabling the sum of the rigidity forces on the end to point outward largely along the longitudinal direction

1. Alignment. The model aligns to the the object boundary. A coarse scale can be used for fast convergence. The elasticity force should be allowed for an end when it lies far outside of the object. Weak balloon forces are used to overcome the attraction of edges from inside the object. Balloon forces also usually ensure that the ends are in a convex form. The rigidity force on an end can push the end growth and should be allowed when the end is not far outside of the object.

2. Length Adaption. When the model only covers a part of the object as a result of the first step, dragging forces are exerted on one or both ends for the model to grow until it covers the whole object. A finer scale can be used for this step.

3. Relaxation and exact localization. Dragging forces and balloon forces should be reduced to a minimum. Fine scale is used for this step.

3 Experiments

Data and preprocessing: In the recording setup, worms move freely on an agar plate under a microscope. Images with the size of 1392×1040 pixels are taken by a camera at 3 frames per second. The resulting pixel size is around 10 μm. The images undergo brightness stabilization and shading correction. For initialization of the LCS model, binary masks of worms are obtained by thresholding.

Description of worm shapes: The model is initialized as close to the desired position as possible. If there is no loop present (which can be determined by a simple analysis of the binary mask), the boundary of the binary mask is carefully analyzed and corresponding pairs are searched based on the *perpendicular requirement* of LCS. These pairs, combined with interpolation and estimation, initialize the model. In the presence of loops, the model is initialized with the result of the preceding frame. In this case, the model usually does not align completely with the object and it actually takes the steps described in section 2.4 for the model to localize.

As shown in Fig. 3, the model finds the correct boundaries and describes worm shapes with high accuracy. The position of a worm is often represented by its longitudinal center line. In [4], the skeleton of the binary mask is taken as the center line. With LCS, the center line is just given by

$$\left\{(\mathbf{x}_0(i) + \mathbf{x}_1(i))/2 \mid i = 0, 1, \cdots, l\right\} \quad . \tag{22}$$

We compare the center lines obtained by the two methods in Fig. 4. LCS yields smoother and more accurate center lines and can handle the situation when a worm forms a loop.

Location and segmentation with rough initialization: LCS models can also be used for segmentation with an initialization that is quite different from the true boundary. In Fig. 5, the model is initialized as a straight one, overlapping only a little of the object. When a part of one side aligns with the true boundary, the corresponding part on the other is dragged or pushed to the correct position.

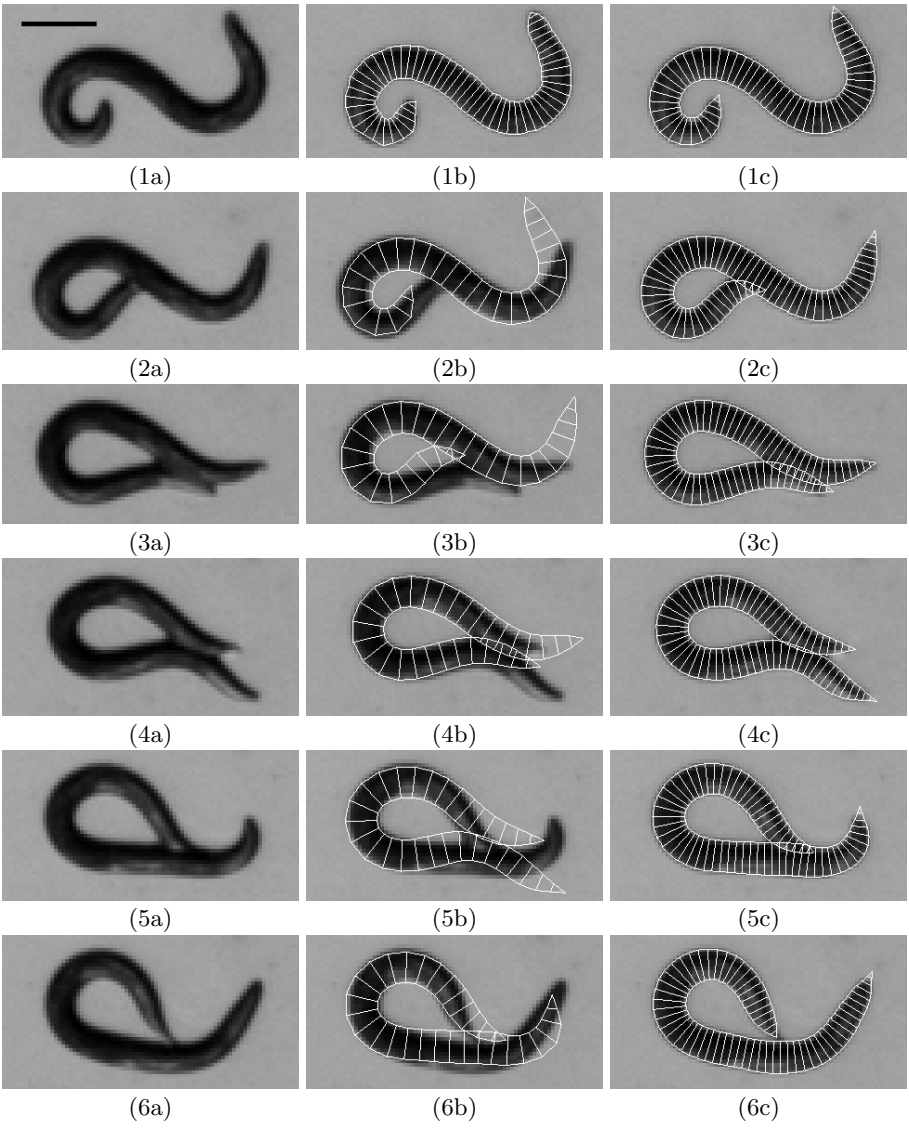


Fig. 3. LCS model with close initializations. Scale bar in (1a) corresponds to 20 pixels, or about 200 μm. The first column shows images of a worm in temporal order cropped from a video clip. The second column shows the initialization positions of the LCS model. The last column gives the final contours. In the first row, there is no loop, the model is initialized by analyzing the binary mask. In the other rows, the model is initialized with the result of the preceding frame and we take a multiscale approach for the model to evolve. The main weight parameters are set as $\alpha = 0.06$, $\beta = 4.0$, $\mu = 1.6$ and $\kappa = 1.0$.

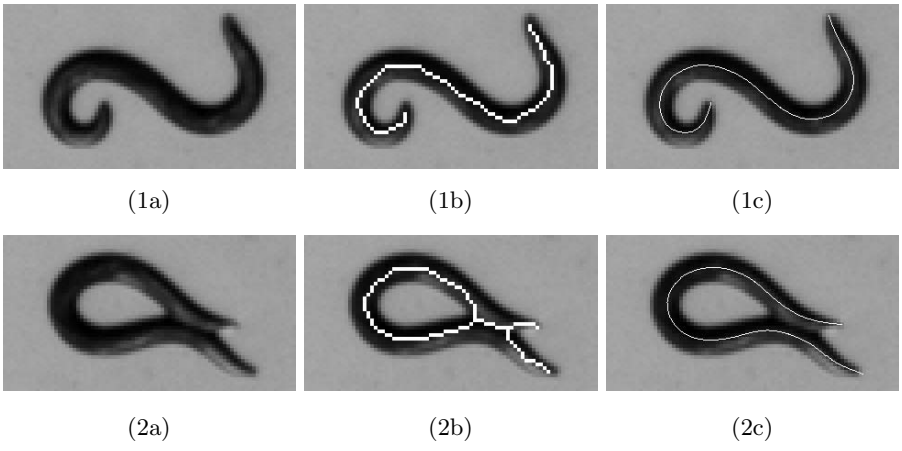


Fig. 4. Center lines of worms found by skeletonizing the binary masks (1b and 2b) and by deriving from the LCS model (1c and 2c), respectively

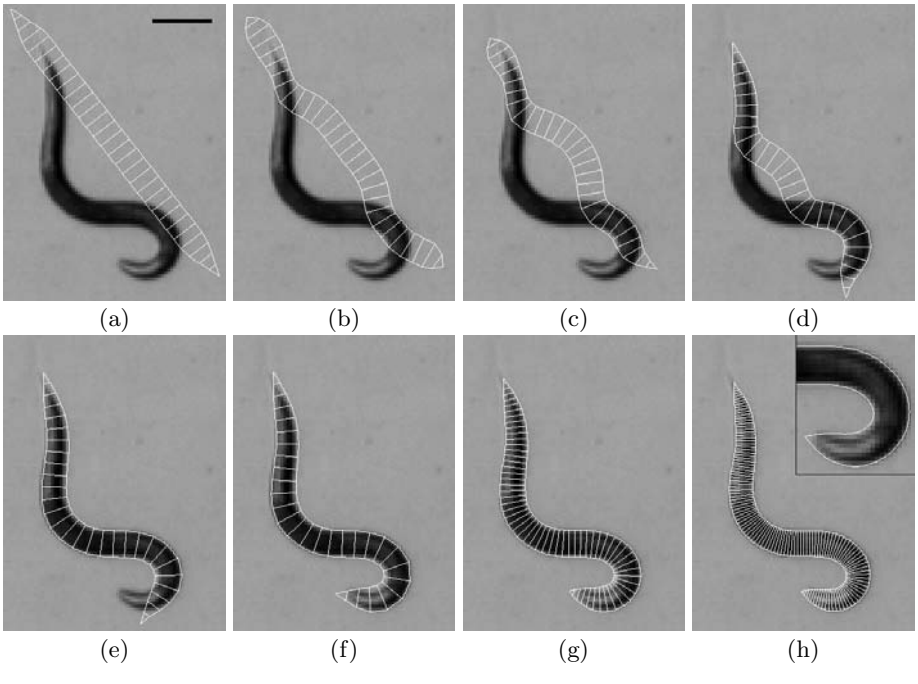


Fig. 5. Evolution of LCS model with a straight initialization. Scale bar in (a) corresponds to 20 pixels, or about 200 μm. (a) Initialization. (b)-(f) Intermediate steps of alignment. (g) The length having been adapted. (h) Relaxation and exact localization. The inner image in (h) shows detailed contour with no linking lines being displayed. With this example, the second step as described in section 2.4 is actually not necessary. The main weight parameters are set as $\alpha = 0.06$, $\beta = 4.0$, $\mu = 1.6$ and $\kappa = 1.0$.

4 Conclusion and Outlook

The proposed lateral coupled snake model can describe shapes and positions of worms in their internal coordinates with subpixel accuracy. It also proves to be a promising tool for segmenting elongated objects with two sides that are largely parallel.

The evolution of the model in this paper makes use of the gsl ode solver. The alignment process shown in Fig. 5 takes less than 10 seconds on a mainstream PC. We will test whether a direct update with timestep adaptation schemes tailored for the model can reduce the computation time.

A future direction of our work is to extent the model to 3D so that it can be applied to pipe-like objects. The ideas of coupling snakes through shear restoring forces and using forces on the ends flexibly shall also be useful there.

Acknowledgment

This study was supported by the Excellence Initiative of the German Federal and State Governments (EXC294, bioss, FRIAS) and FRISYS (funded by BMBF FORSYS).

References

1. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International Journal of Computer Vision 1(4), 321–331 (1988)
2. Cohen, L.D.: On active contour models and balloons. Computer Vision, Graphics, and Image Processing. Image Understanding 53(2), 211–218 (1991)
3. Xu, C., Prince, J.L.: Snakes, shapes, and gradient vector flow. IEEE Transactions on Image Processing 7(3), 359–369 (1998)
4. Geng, W., Cosman, P., Berry, C., Feng, Z., Schafer, W.: Automatic tracking, feature extraction and classification of c. elegans phenotypes. IEEE Transactions on Biomedical Engineering 51(10), 1811–1820 (2004)
5. Fua, P.: Model-Based Optimization: An Approach to Fast, Accurate, and Consistent Site Modeling from Imagery. In: Firschein, O., Strat, T.M. (eds.) RADIUS: Image Understanding for Intelligence Imagery, pp. 903–908. Morgan Kaufmann, San Francisco (1997)
6. Mayer, H., Laptev, I., Baumgartner, A.: Multi-scale and snakes for automatic road extraction. In: Burkhardt, H.-J., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1406, pp. 720–733. Springer, Heidelberg (1998)
7. Cham, T., Cipolla, R.: Stereo coupled active contours. In: Proc. CVPR, pp. 1094–1099 (1997)
8. Gunn, S.R., Nixon, M.S.: A robust snake implementation; a dual active contour. IEEE Trans. Pattern Anal. Mach. Intell. 19(1), 63–68 (1997)
9. Hohnhuser, B., Hommel, G., Berlin, T.U., Iv, F.: 3D pose estimation using coupled snakes. J. WSCG 12, 1213–6972 (2003)
10. gsl: `http://www.gnu.org/software/gsl/`