Modelling Active Contours using Neural Networks Isomorphic to Boundaries

Y.V.Venkatesh & N.Rishikesh
Computer Vision and Artificial Intelligence Laboratory
Department of Electrical Engineering
Indian Institute of Science
Bangalore-560012, India.

Abstract : *We propose a new technique, based on self-organization, for localizing salient contours in an image, with applications to, for instance, object and character recognition, stereopsis and motion tracking. A neuronal network which is isomorphic to the template/initial contour is created. This network acts as an active contour, which, using self-organization, undergoes deformation in an attempt to* cling *on to the nearest salient contour in the test image. The application areas of the model proposed are similar to 'snake' [2], but* distinct *from it both in the underlying mathematics and implementation. The new technique is illustrated with some examples.*

Indexing terms : *Active contours, Deformable templates, Deformation of Patterns, Isomorphism, Mapping of Patterns, Neural networks, Self-Organization, Snakes.*

1 Introduction

A machine to recognize, localize and identify two- (and three-) dimensional objects still appears to be an elusive goal of the Artificial Intelligence community. Inspired by the remarkable abilities of the human vision system, we require that the performance of such a machine be independent of (i) standard transformations like shift, scaling, and rotation; (ii) missing information; and (iii) distortions. In human vision modelling, low-level operations, like edge or line detection, stereo matching, and motion tracking, are assumed to provide inputs to higher-level processing in a rigid hierarchy. However, it has now been widely recognized that the human vision system is more heterarchical than hierarchical, and the influence of higher-level processes in guiding/correcting low-level operations is significant. And this constitutes the motivation for an application of deformable contours for object recognition.

In using deformable templates, the *a priori* shape (or boundary) information is assumed to be available in the form of an inexact model of the object as a sketch or a binary template. The problem of object recognition is then formulated as the process of matching a deformable contour to the object boundary in an input image. For an early result on the representation of the deformation as a probabilistic transformation on the prototype template, see Grenander [1].

Snakes, proposed by Kass et al. [2], are an example of a technique of contour extraction by means of energy minimization. In this reference, the authors choose energy functionals which attract the snakes to salient features (lines, edges, and terminations) in images. From a given starting point, the snake deforms itself into conformity with the nearest salient contour. In effect, their model is a controlled-continuity spline under the influence of image forces and external constraint forces. The internal spline forces serve to impose a smoothness constraint, whereas the external constraint forces are responsible for positioning the snake near the desired local minimum.

Lai and Chin [3] present a generalized snake model for modelling contours, and discuss its application to the recognition of characters and to the analysis of noisy images. In this reference, the authors use the Generalized Hough Transform [4] for the initialization of the contours. In [5], these authors suggest a method to initialize snakes, and apply regularization theory for finding a min-max solution to the optimization problem.

In Jain et al. [6], deformable templates are employed for object detection. Here, prior knowledge of an object shape is described by a prototype template which consists of the representative contour/edges, and a set of probabilistic deformation transformations on the template. A Bayesian scheme, which is based on this prior knowledge and the edge information in the input image, is used to find a match between the deformed template

0-7803-4122-8/97 $10.00©1997 IEEE

and objects in the image.

Amongst the other contributions, mention should be made of (i) Staib et al. [7] and (ii) Yuille et al. [8]. The former uses a parametric model based on elliptic Fourier decomposition of the boundary of the object for locating its boundary. In the latter reference, the authors employ a parametrized template for an eye consisting of a circle bounded by two parabolas. The template is deformed to the image by optimizing a cost function based on morphological features. They also develop a similar template for the mouth.

2 Network Models based on Human Perception

Human Vision System is capable of recognizing patterns in spite of scale changes, rotation and shift. One possible way of achieving this could be by a conscious establishment of correspondence between significant features of the model and those of the retinal image. The classification result then would depend on the ease of correspondence with each of the model images (examplars). The exemplar with which the correspondence is established most easily could then considered to be the class to which the test pattern belongs. In other words, human recognition could be considered to be guided by the amount of mental deformation the exemplar has to undergo to match the given unknown pattern. This idea has been exploited in object and character recognition by [9], [12]. In these references, the authors have employed a network of neurons for each exemplar, with the neurons in each network arranged in exactly the same way as the corresponding exemplar. Finally, the correspondence between the test pattern and the exemplars is established by mapping the networks onto the test pattern, using a self-organization scheme similar to Kohonen's [10] algorithm. While this approach uses a locally co-operative weight-updating, self-organizing scheme similar to the Kohonen's algorithm [10], it is *completely* different from it in terms of the underlying architecture of the network.

The present paper, in the course of exploiting the simplicity and efficiency of the above approach, appropriately modifies it so as to be applicable to the problem of contour extraction. The paper introduces, in modifying the above method to suit the problem in hand, constraints on the network which are equivalent to the internal forces in 'snakes'(Sec. 3.1). The main contribution of the present paper is believed to be the exploitation of the principles of spatial isomorphism and self-organization to model an active contour to extract the nearest salient contour from a given image.

In the next section, a description of the proposed technique is given, and the results of its application to extraction of salient contours are presented in Sec. 4.

3 Neural Network Algorithm for Extracting Salient Contours

The approach proposed here, being a semi-automatic image interpretation technique, requires the initial contour to be given by the user with a suitable user interface. A neural network isomorphic to this initial contour is constructed, and subjected to deformation in order to map onto the nearest salient contour in the image. The correspondence between the salient contour and the network is established by mapping the latter onto the former, using a self-organization scheme [10], [11]. The steps involved in such a mapping are as follows:

1. Calculate the edge map of the test image.
2. Initialize the initial contour from where the system has to start with, using a suitable user interface (either through mouse or from a file). Choose the region of interest according to the location of the initial contour.
3. Get the edge points $E = \{x_i, y_i\}$ within the region of interest.
4. Construct a network with the number of neurons equivalent to the number of contour points (N_c) in the initial contour. Each neuron in the network gets two inputs (I_1, I_2), and the weights, $W^i = (W^i_1, W^i_2), i = 1, ..., N_c$, corresponding to these two inputs, are initialized to the co-ordinates of the initial contour points. In effect, a neural network isomorphic to the initial contour is constructed. Note, further, that N_c depends *only* on the initial contour.
5. Repeat the following process for a certain number of times (N_{iter}):

   (i) Select a point $P = (u, v) \in E$ randomly, and feed the (x, y) coordinates of the selected point P as inputs to every neuron in the network.

   (ii) Determine the neuron whose weight vector is closest to the input vector, and declare it as the winner neuron.

   If the "closeness" of the winner neuron's weight vector (W^w) is greater than a particular threshold T_{wmd}, then go to (i) without update.

   (iii) Update the weights of the neurons in the network using the following rule :

   For neuron i,

$$W^i = W^i + \eta * e^{-\|W^w - W^i\|^2/2*\sigma^2} * (P - W^i)$$

where W^i, P are vectors, and η, σ are the standard learning rate and neighborhood parameters.

Calculate the NCP (Neighborhood Constraint Parameter) of the contour as:

$$C_{ncp} = Max(Max(\mid W_1^i - W_1^{i+1} \mid, \mid W_2^i - W_2^{i+1} \mid)),$$

where (i) $i = 1, ..., N_c - 1$, and (ii) Max is taken with respect to all the points in the network.

If $C_{ncp} > T_{ncp}$, the threshold value of the neighborhood constraint parameter, then unupdate the network by replacing the old network weights.

(iv) Vary η and σ according to the rules:

$$\sigma = \sigma_{init} * (\sigma_{fin}/\sigma_{init})^{iter/N_{iter}};$$
$$\eta = \eta_{init} * (\eta_{fin}/\eta_{init})^{iter/N_{iter}};$$

where *iter* is the current iteration number.

3.1 Constraints on the Network

Two major constraints are placed on the proposed network model which implicitly imposes the smoothness constraint. In contrast, in the 'snake' [2] model, the external and internal constraint forces are made explicit, and an energy function comprising the two is minimized to get the final contour. On the other hand, in the proposed approach, the internal constraints are made implicit by placing the constraints listed below. As far as the external constraints (image forces, as called in [2]) are concerned, they are also implicitly taken care of by the input fed to network acting as the contour model. The two constraints on the contour model are: Constraints on the (i) Winner-Maximum-Distance (WMD); and (ii) Neighborhood Constraint Parameter (NCP). The two are discussed in brief below.

In accordance with the Winner-Maximum-Distance (WMD) constraint, the weights of the network are not updated, if the distance between the present input point and the winner neuron's weight vector is greater than a pre-specified threshold, T_{wmd}. It is to be noted that the WMD constraint is useful in avoiding the influence of spurious edge points (on the process of self-organization of the network) within the region of interest but which are not a part of the nearest salient contour. This is shown in Figure 1, where the network organizes itself to the ellipse, in spite of the spurious edge points inside the ellipse.

The second constraint is on the Neighborhood Constraint Parameter (NCP), which is basically the *maximum of the maximum* of the distances between the adjacent points of the contour in the x and y directions. This particular parameter is useful in maintaining the continuity of the contour as it gets deformed. If this parameter is not constrained by a threshold, many neurons will try to learn a particular point of the input image, leading to discontinuities in the final contour. The threshold parameter T_{ncp} is essentially the maximum permitted neighborhood violation. The validity of the constraint is illustrated in Figure 2, where a higher value of T_{ncp} leads to a highly broken contour, while a lower value gives a continuous one.

4 Experimental Results and Applications

The implementation of the model was accomplished using C++, with Xlib for graphical user interface. The experiments were conducted on the *HP*9000/715. For an image of size 128 × 128, the program takes 5-6 seconds to give the final contour and converge. Typical values of important parameters used in the above system are as follows: number of iterations, $N_{iter} = 300 - 600$, depending upon the size and shape of the contour; initial value of σ, $(\sigma_{init}) = 3 - 5$; its final value, $(\sigma_{fin}) = 0.1 - 0.3$; initial value of the learning rate parameter, $(\eta_{init}) = 0.9 - 0.7$; its final value $(\eta_{fin}) = 0.01 - 0.001$; and the threshold parameters T_{ncp}, $T_{wmd} = 2 - 5$, depending on the network.

We now indicate briefly some distinct characteristics and applications of the proposed approach:

- Noise in the input Image:

  The approach is found to be very robust to noise present in the input image. When a noisy image is an input to the network, the network has been found to be capable of extracting the nearest salient contour from the image. This is illustrated in Figure 3, where the contour even in a noisy image has been extracted successfully. In the illustrated example the percentage of noise is 20.

- Open Contours:

  The approach is also capable of extracting open contours from a given image. This is shown in Figure 4, where an open contour is shown with background noise. The network is shown to extract the salient contour.

- Kinks in the Contour:

  Figure 5(a) shows the initial contour being pulled off from the actual salient contour. The final contour is shown in Figure 5(b), where the network has deformed and 'snapped' itself appropriately to the actual contour.

- Stereo Images and Motion Sequences:

  This approach can be used efficiently for stereo matching problems in extracting the corresponding contours from the left and right images (for depth measurement). This approach can also be used for contour tracking in motion image sequences, where contour initialization is required only for the first of the images, and a *locking-on* mechanism can be invoked to track the contour.

- Other Applications:

  This approach has already been shown to be efficient for character and object recognition [12]. Application to face recognition and tracking in a video sequence is under study. It is expected that the use of the generalized Hough transform for contour initialization would improve the efficiency of the approach in these applications.

5 Conclusions

We have presented a new technique, involving active contours modelled with self-organization in a neural architecture, for extracting salient contours in an image. The neuronal network, which is created to be isomorphic to the initial contour, is mapped onto the given test object in the image by self-organization, during which the network undergoes deformation in an attempt to match the test object.

The model proposed has application areas similar to those of 'snake' [2]. But the model is quite *distinct* from the snake-model both in the underlying mathematics and computer implementation. The advantages of the proposed approach for object contour extraction are: simplicity and speed; ability to extract both open and closed contours; robustness to noise; no special energy functions are required. The new technique is illustrated by some examples.

References

[1] Y. Amit, U. Grenander, and M. Piccioni, Structural image restoration through deformable templates, Journal of Americal Statistical Association, Vol.86, No.414, pp. 376-387, June 1991.

[2] M. Kass, A. Witkin, and D. Terzopoulos, Snakes: Active contour models, Proc. First IEEE Conference on Computer Vision, 1987, pp 259-268.

[3] K.F. Lai and R.T. Chin, Deformable contours : Modelling and Extraction, IEEE Pattern Analysis and Machine Intelligence, Vol. 17, no 11, pp 1084 - 1090, 1995.

[4] D.H. Ballard, "Generalizing the Hough Transform to detect arbitrary shapes, Pattern Recognition, Vol 13(2), pp 111-122, 1981.

[5] K.F. Lai and R.T. Chin, On regularization and initialization of the active contour models (snakes), Proc. First Asian Conference on Computer Vision, 1993.

[6] A.K. Jain, Y. Zhong, and S. Lakshmanan, Object matching using deformable templates, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.18, No.3, pp 267-277, 1996.

[7] L.H. Staib and J.S. Duncan, Boundary finding with parametrically deformable models, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.14, No.11, pp 1061-1075, 1992.

[8] A.L. Yuille, D.S. Cohen, and P.W. Hallinan, Feature extraction from faces using deformable templates, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp. 104-109, June 1989.

[9] C.N.S. Ganesh Murthy and Y.V. Venkatesh, Pattern encoding and classification by neural networks, revised version submitted to the IEEE Trans. on Neural Networks (May 1996).

[10] T. Kohonen, Self Organization and Associative Memory, Springer-Verlag, 1989.

[11] C.N.S. Ganesh Murthy, K. Shanmukh and Y.V. Venkatesh, A New Method for Pattern Recognition Using Self-Organizing Networks, Proc. Second Asian Conference on Computer Vision(ACCV), Singapore, Vol.3, pp.111-115, December 1995.

[12] K. Shanmukh, C.N.S. Ganesh Murthy and Y.V. Venkatesh, Classification using self-organizing networks spatially isomorphic to patterns, International Conference on Robotic Vision for Industry and Automation, ROVPIA'96, Ipoh, Perak, Malaysia, pp. 298-303, November 28-30, 1996.