Computación y Sistemas Vol. 2 Nos. 2-3 pp. 87-94
© 1999, CIC - IPN. *ISSN 1405-5546* Impreso en México

Eels: Electric Snakes

Antonio Marín-Hernández
Facultad de Física e Inteligencia Artificial
Universidad Veracruzana
Sebastián Camacho # 5, Xalapa, Veracruz, México.
amarin@mia.uv.mx

Homero V. Ríos-Figueroa
Laboratorio Nacional de Informática Avanzada, A. C.
Rébsamen # 80, Xalapa, Veracruz, México
hrios@xalapa.lania.mx

Article received on September 18, 1998; accepted on February 12, 1999

Abstract

This paper presents the incorporation of a new internal force for active contours. This internal force is generated by an electric charge, distributed all-over the active contour. Our extended contour model can describe objects with high curvature, without increasing the number of control points.

Keywords:

Active contours, internal energy, electric forces.

1 Introduction

Snakes or active contours were originally proposed in (Kass *et al.*, 1987), as a segmentation tool based on energy minimizing. This kind of contours has been used in many image analysis applications, particularly to locate object boundaries.

Simulated internal and external forces deform snakes to adapt to the object's shape. The internal forces are normally the rigidity and the tension, while the external forces are defined as a scalar potential function over the image plane.

A snake is a parameterized curve **v** with components $x(s)$ and $y(s)$, where s is defined in the unit domain [0,1]. We represent the contour **v** locally by piecewise polynomials, generated by cubic B-Splines.

There are several methods for achieving the minimal energy state. One of the most used is dynamic programming proposed by (Amini, 1990). However the time complexity is $O(nm^{k+1})$, where k indicates the highest order derivative of the contour geometry, n is the number of control points and m is the number of possible choices at each control point.

Basically there are two mayor problems with this representation. We need more control points in the regions with high curvature and we cannot interpolate control points with B-Splines. By replicating control points, one can force a B-Spline to interpolate the control points. In (Menet *et al.*, 1990) control points are duplicated in regions where after N steps of contour deformation, the curvature is higher than a threshold. However, if we increase the number of control points the time complexity increases too.

Schnabel and Arridge (1995) proposed the introduction of Multiscale Differential Operators (MDO), who are invariant to linear intensity transformations such as contrast or brightness adjustments and independent of coordinate transformations. In order to introduce MDO to achieve matching, the active contour energy model has to be modified, integrating information of the image and contour curvature for every patch in the contour. This increases the complexity of the algorithm making it slower and reduces possibilities for tracking objects.

Another important idea that contributed to this paper is the introduction of Gradient Vector Flow (GVF) as a new external energy (Xu and Prince, 1997). GVF is computed as a diffusion of the gradient vectors derived from the image. The GVF is like an electrostatic potential produced by an electrical charged object. The introduction of GVF makes the active contour able to converge to boundary concavities, but the reported computation of GVF for a 256 x 256-pixel image takes more than 50 seconds.

We propose an electric charge density uniformly distributed over each span of the initial contour, in order to concentrate more control points in the regions of high curvature as a result of the internal electrostatic forces, without increasing the number of control points.

2 Active Contour Model

For active contours we define an energy function as:

$$E(\mathbf{v}) = E_i(\mathbf{v}) + E_e(\mathbf{v}), \qquad (1)$$

where E_i is the internal energy and E_e is the external energy, over the contour **v**. The contour **v** is a mapping from the unit parametric domain $s \in [0,1]$ into the image plane.

We define the internal energy as:

$$E_i(\mathbf{v}) = \int_0^1 \left(\omega_1(s) |\mathbf{v}_s|^2 + \omega_2(s) |\mathbf{v}_{ss}|^2 \right) ds \qquad (2)$$

where the subscripts on v denoted differentiation with respect to s, w_1 and w_2 are parameter functions which control the tension and the rigidity, respectively.

Originally (Kass et al., 1987), the external energy was divided in two parts:

$$E_e(\mathbf{v}) = E_{image}(\mathbf{v}) + E_{rest}(\mathbf{v}),$$

where E_{image} contains image information and E_{rest} contains external restrictions. This division is not necessary, because sometimes, as explained in the introduction, a combination of these two energies is used in one term.

We use the external energy as follows (Terzopoulos and Szeiliski, 1992):

$$E_e(\mathbf{v}) = \int_0^1 P(\mathbf{v}(s))\, ds \qquad (3)$$

where $P(x,y)$ is a scalar potential function defined over the image plane, as:

$$P(x, y) = -c \left| \nabla \left[G_\sigma * I(x, y) \right] \right|, \qquad (4)$$

where c controls the magnitude of the potential and $G\sigma * I$ denotes the image convolved with a Gaussian filter. The ∇ operator represents the gradient of the image intensities.

Some authors (Terzopoulos and Szeliski, 1992) define a kinetic energy for the contour, in order to make a dynamic energy minimization. Then they define the Lagrangian (Terzopoulos, 1987) in terms of the kinetic and potential energy of the contour, and then get the Lagrange equations of motion.

In this paper we did not use the Langrangian formulation, instead we use dynamic programming for the energy minimization, as described below.

3 Contour Representation

There are several contour representations that can be divided in two classes, depending on whether they are global or local. Global representations are typically compact but changes in one shape parameter affect the entire contour, and conversely, local changes of the contour affect all parameters.

On the other hand, local representations control the contour shape by various parameters, which depend locally on contour shape; this makes local representations well suited in a shape reconstruction context. Most local representations used describe contours in terms of piecewise polynomials. Each segment is described by a polynomial in s.

Cubic B-Splines present an efficient way to represent curved objects, where each segment of the curve $\mathbf{v}_i$ is defined by four control points ($\mathbf{P}_{i-1}$, $\mathbf{P}_i$, $\mathbf{P}_{i+1}$, $\mathbf{P}_{i+2}$) as:

$$\mathbf{v}_i(s) = \begin{bmatrix} s^3 & s^2 & s & 1 \end{bmatrix} \cdot \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{P}_{i-1} \\ \mathbf{P}_i \\ \mathbf{P}_{i+1} \\ \mathbf{P}_{i+2} \end{bmatrix} = \begin{pmatrix} x_i(s) \\ y_i(s) \end{pmatrix} \qquad (5)$$

One of the problems with this representation is the fact that B-Splines do not interpolate control points. Sometimes this problem is solved duplicating control points when the curvature is higher than a threshold (Menet et al., 1990).

In (Gavrila, 1996) a Hermite contour representation is proposed, which is obtained replacing the square matrix in (5) by:

$$\begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

and the control vector by:

$$\begin{bmatrix} \mathbf{P}_{i-1} & \mathbf{P}_i & \tau^{+}_{i-1} & \tau^{-}_{i} \end{bmatrix}^T$$

where $\mathbf{P}_{i-1}$, $\mathbf{P}_i$ are control points and, τ^{+}_{i-1}, τ^{-}_{i} are tangent vectors at points *i*-1 and *i* respectively.

This technique has the following advantages:

- Efficiently represent both smooth and sharp contours.
- Easily interpolates control points.

But has the following disadvantage. It is necessary to have information of the tangent vector in every control point in order to get the span parameters. This can be solved using a template matching strategy, but then we need templates for every object.

4 Energy Minimization

There are many ways to solve the minimization problem. One possibility for discrete spaces is to use AI techniques, but they need an exhaustive enumeration of the possible solutions.

We use dynamic programming popularized by (Amini, 1990) in multiple scales to decrease the computational cost. Basically, the dynamic programming consists in search around a control point and finding which neighbor produces a minimum energy (Fig. 1).

We accelerate the dynamic programming using multiple scales, convolving the image at every scale with a Gaussian filter of standard deviation σ. We begin the search for every control point on the curve, over the eight closer neighbors at a distance s, as described in Fig. 2.

Once we find a contour with a minimum energy we reduce the scale and repeat the same process for each scale, where we use as the initial contour the one found in the previous scale.

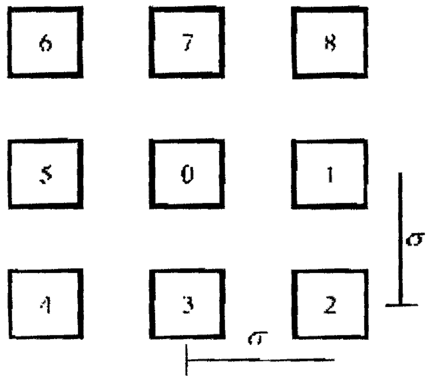


Figure 2. Eight closer neighbors of a control point P_i at scale σ.

This technique has a problem when the control points are close and σ has approximately the same value. In this case a control point can find a minimum in a region between the neighbors control points. The resulting curve will have a twist over itself as shown in Fig 3. One way to solve this problem is using a threshold. If the distance between two control points is smaller than the threshold that position is invalid. Another way is reducing the neighbor's search along lines that are normal to the initial curve (Curwen and Blake, 1992). An interesting reduction of complexity for this search is shown in (Olstad and Tysdahl, 1993). It is based principally on constrains for the selection of candidate points.

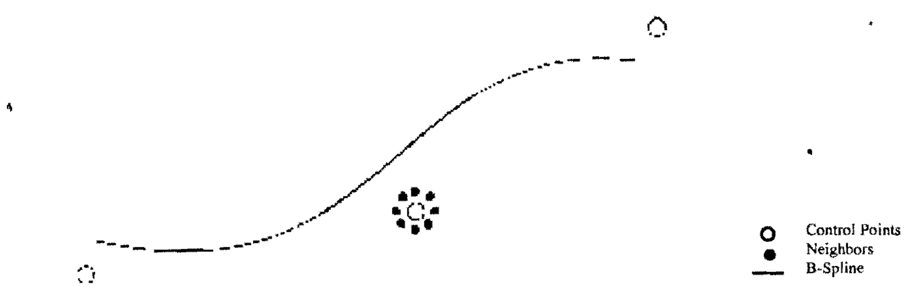


Figure 1. Dynamic programming search.

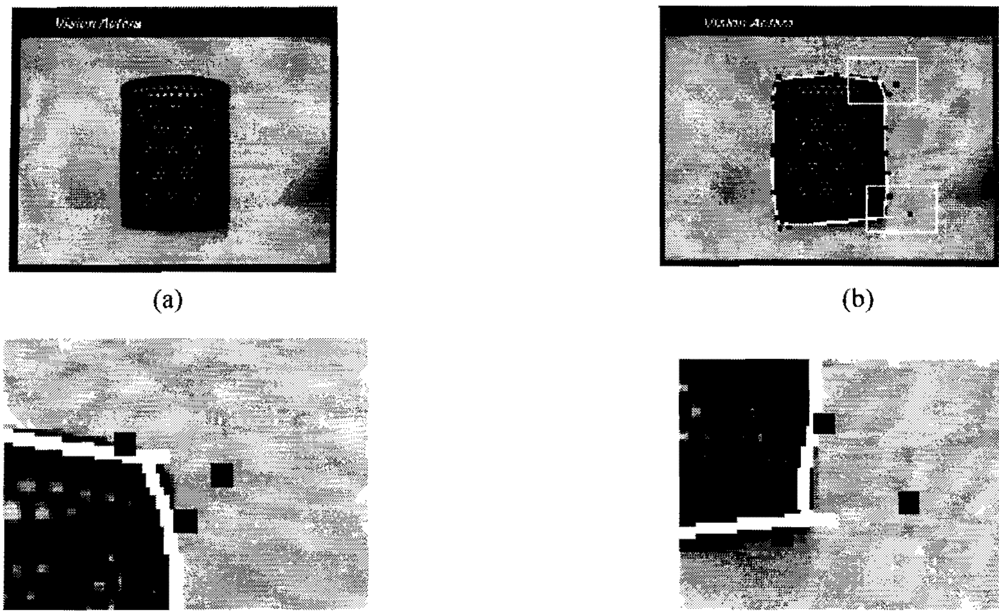


Figure 3. (a) Object, (b) contour minimization, (c) and (d) contour twisted.

5 Electric Snakes

We propose to assign a constant electric charge Q over the contour, in order to get an electric force between every pair of points in the contour, described as:

$$\mathbf{F} = k\frac{q^2}{r^2} \qquad (6)$$

where q is the electric charge associated to every point, r is the distance between points and k is a constant.

The electric charge is distributed uniformly over the initial snake, in order to make that each span has the same charge. The main purpose is to make a repulsion force between each point associated with control points, where the energy is calculated. This force produces that control points can approach to one another only when the other forces in the model are greater than the electric force.

We use the following physics results (Weidner and Sells, 1971):

(a) In a conductor of arbitrary shape with a net charge, all of the net charge will reside on the outer conductor surface.

(b) The electric field lines are always perpendicular to the exterior conductor surface.

(c) The electrical field inside any conductor is exactly zero.

(d) Once the conductor has reached the equilibrium the free charges do not move over the surface.

Therefore, the electric charge density in a conductor with a net charge will be greater in regions where the curvature is high (Feynman et al., 1964; Price and Crowley, 1985).

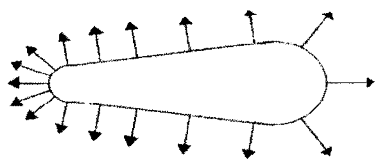


Figure 4. In a charged conductor the electric field is strong at points of high curvature (Weidner and Sells, 1971).

Then, when equilibrium is reached the control points will be form clusters in regions where the curvature is high, and they are not going to move around the surface.

The total energy equation will be:

$$E_t(\mathbf{v}) = \int\left(\omega_1(s)|\mathbf{v}_s|^2 + \omega_2(s)|\mathbf{v}_{ss}|^2\right)ds + \int k\frac{q^2}{r^2}\,ds + \int P(\mathbf{v})\,ds \qquad (7)$$

Using equation (7) in energy minimization, the control points will define in a better way the contour, without increasing the number of control points and the time complexity.

6 Results

The following images show the results of contour fitting with standard and electric snakes.
Figure 5 (a) displays the image energy computed as described in equation (4), in (b) the standard snake fits a mouse embryo. Here we can find regions where the active contour is twisted (dotted circles). The contour is evolving towards the object in (c) and in (d) the object is fitted. The distribution of control points with electrical forces makes a better description of the contour.

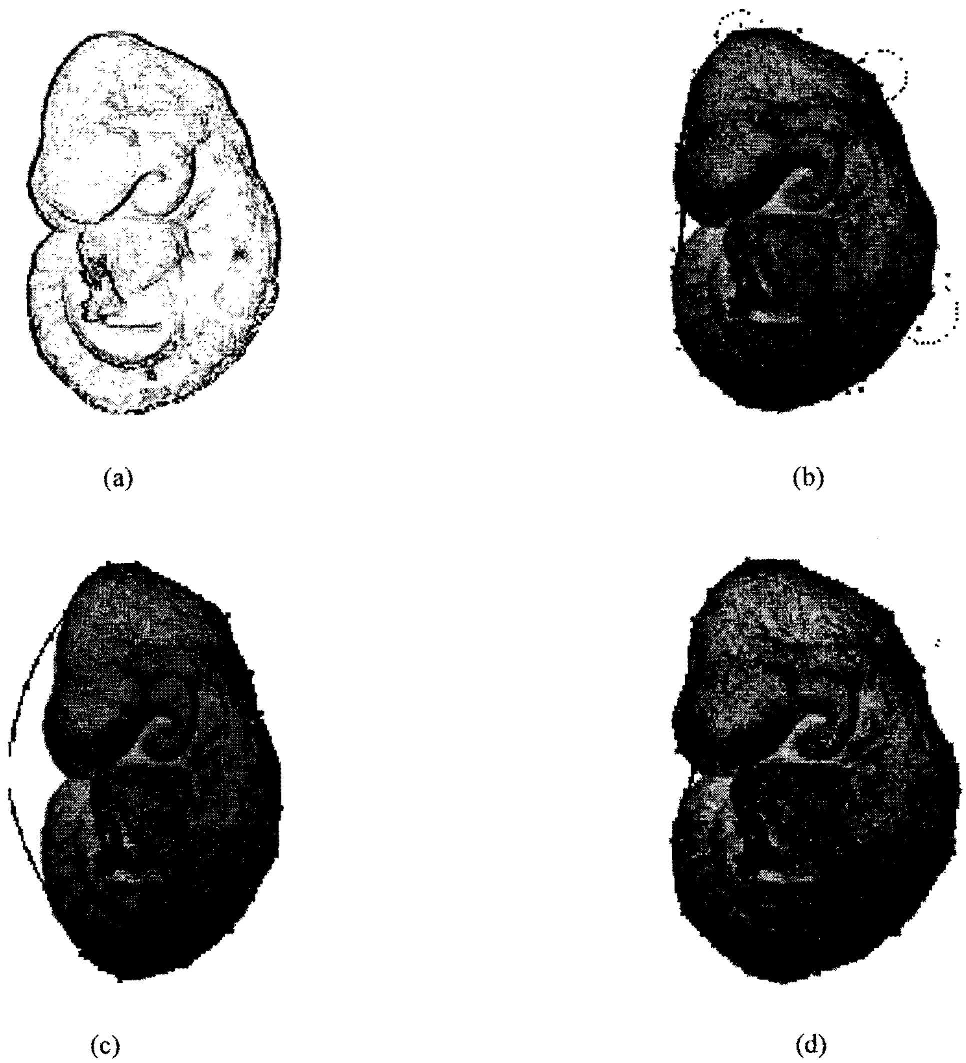
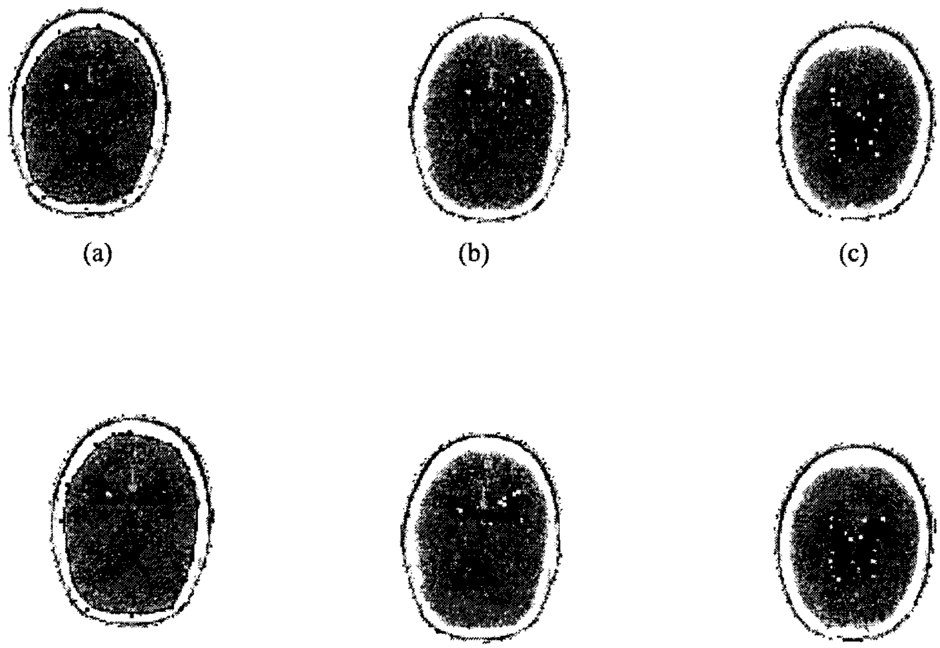


Figure 5. (a) Potential energy in mouse embryo image, (b) contour adjust with standard snake, (c) snake evolution with electric forces and (d) final fit with electric snake.

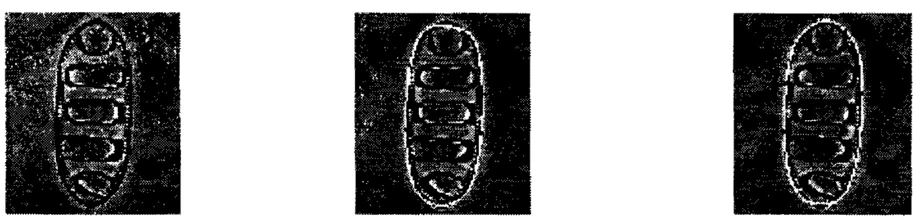
Figures 6 (a), (b) and (c) show anatomical structures in head tomographies, identified by standard snakes. The same structures are better fitted by electric snakes (fig. 6.d, e and f). Another example of segmentation of biological shapes (Phytoplankton) is presented in figure 7.



(a) (b) (c)

(d) (e) (f)

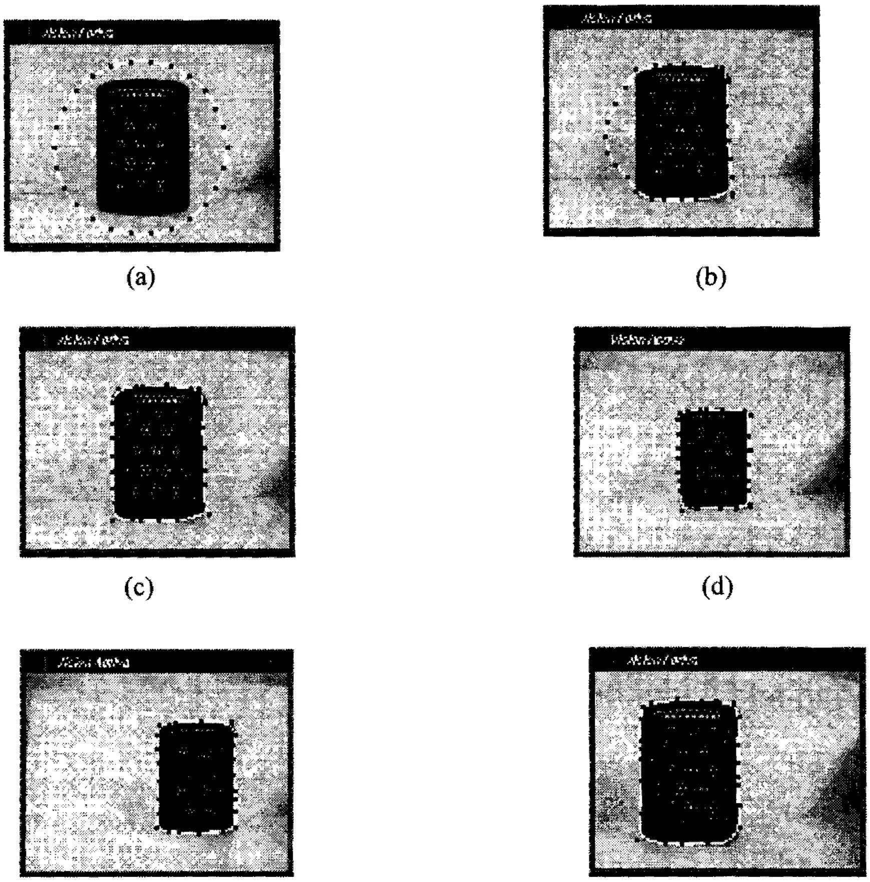
Figure 6. (a), (b) and (c) final fit with standard snake, (d), (e) and (f) final fit using electric forces.



(a) (b) (c)

Figure 7. Phytoplankton adjustment (a) original image, (b) standard snake and (c) electric snake.

A final example of electric snakes is object tracking (figure 8). The final snake fit in each frame is used as initial contour for the nest frame. The perfomance of our implementation on a SGI Indy is between 8 to 16 per second, depending on the number of control points used.



(a) (b) (c) (d) (e) (f)

Figure 8. Object tracking: (a) initial contour, (b) contour evolution, (c) final fit, (d) (e) y (f) electric snake tracking.

7 Conclusions

We have presented a new internal force for active contours, which adapts to points of high curvature. This force follows electromagnetic properties of matter and is modeled by introducing a uniformly distributed charge over the initial snake. As the contour evolves the control points concentrate around regions of high curvature, without the need of introducing additional control points. The number of control points needed depends on the number of regions of high curvature and the application, but it was shown that a few points are enough to describe and track objects not too complex.

Acknowledgements

This work was supported by the Mexican National Council of Science and Technology (CONACyT) as project ref. C098-A and 97-05-002-V.

We wish to thank Dr. Boris Escalante-Ramírez, for providing the head tomography and also to the image processing group of CICESE, for the phytoplankton image.

References

Amini, A.A., T.E. Weymouth and R. C. Jain "Using dynamic programming for solving variational problems in vision". *IEEE Transaction on Pattern Analysis and Machine Intelligence* Vol. 12, No. 9, pp. 855-867, 1990.

Curwen R. and A. Blake "Dynamic Contours: Real-time Active Splines". In: A. Blake and A. Yuille (Eds.) *Active Vision*, MIT Press, Cambridge, pp. 39-57, 1992.

Feynman R.P., R.B. Leighton and M. Sands, *The Feynman Lectures on Physics, Mainly Electromagnetism and Matter*, Volume II, Addison-Wesley Massachusetts, 1964.

Gavrila D.M., "Hermite Deformable Contours". Technical Report, Center for Automation Research, University of Maryland, CAR-TR-817, 1996.

Kass M., A. Witkin and D. Terzopoulos, "Snakes: Active Contour Models". *Proceedings of the 1st International Conference of Computer Vision,* Vol. 1, pp. 259-268, 1987.

Menet S., P. Saint-Marc and G. Medioni, "B-snakes: Implementation and Aplication to Stereo". *Proc. ARPA Image Understanding Workshop*, pp. 720-726, 1990.

Olstad B. and H.E. Tysdahl "Improving the Computational Complexity of Active Contour Algorithms". *Proc. IAPR*, Vol. 1, pp. 257-263, 1993.

Price R.H. and R.J. Crowley "The Lighting-rod Fallacy". *American Journal of Physics*, september, pp. 843, 1985.

Schnabel J.A. and S.R. Arridge, "Active Contour Models for Shape Description Using Multiscale Differential Invariants". *Proc. British Machine Vision Conference*, Birmingham, UK, 197-206, 1995

Terzopoulos D., "On matching deformable models to images: Direct and iterative solutions". *In Topical Meeting on Machine Vision*, Technical Digest Series, Vol. 12, pp. 160-167, 1987.

Terzopoulos D. and R. Szeliski, "Traking with Kalman Snakes". In: Blake A. And A. Yuille (Eds) *Active Vision.* MIT Press, Cambridge, pp. 3-20, 1992.

Weidner R.T. and R.L. Sells, *Elementary Classical Physics*. Allyn and Bacon, USA, Vol. 2, pp. 692-696, 1971.

Xu C. and J.L. Prince, "Gradient Vector Flow: A New External Force for Snakes". *IEEE Proc. Computer Vision and Pattern Recognition*, pp. 66-71, 1997.

__Antonio Marín-Hernández__ degree in Physics in 1995 and M.Sc. degree in Artificial Intelligence in 1998 both from the University Veracruzana (UV). From 1996 to now working as Lecturer at the Faculty of Physics and Artificial Intelligence of the UV in Xalapa, Mexico, and recently as a full time Researcher in the same Faculty.

__Homero V. Ríos-Figueroa__ received the B.Sc. degree in Mathematics and the M.Sc. degree in Computer Science from the National Autonomous University of Mexico (UNAM) in 1987 and 1989, respectively. The Ph.D. degree in Computer Science and Artificial Intelligence from the University of Sussex, England, in 1994. From 1988 to 1990 he was a Lecturer with the UNAM. In 1994, he joined the National Laboratory for Advanced Informatics (LANIA) in Xalapa, Mexico, where he is now a full time Researcher. His research interests include computer vision and virtual reality. He is a member of the Mexican Society of Artificial Intelligence (SMIA), and the Mexican Society of Computer Science (SMCC).