SHAPE MEMORY FOR CLOSED ACTIVE CONTOURS

F. Abram, P. T. Sander

Thème Images/Réalité Virtuelle
Laboratoire UNSA/CNRS I3S
Les Algorithmes, Bâtiment Euclide
2000, route des Lucioles
Sophia Antipolis 06410 BIOT France
abram@i3s.unice.fr http://www.essi.fr/~abram
sander@essi.fr http://www.essi.fr/~sander

ABSTRACT

In many cases, the general overall shape of the contours of specific objects in images is known, e.g., bones and organs in biomedical images. This information can be used to initialize an active contour and to constrain its global evolution while fitting itself by local deformations to the object in the image. The model thus not only extracts the object in the image, but also provides an evaluation of the deformation of the extracted contour with respect to the given original shape. This paper describes an active contour segmentation technique based on such knowledge of contours in images. Following a rough initial placement of the model, the deformation process consists in three stages, a free deformation stage under the effect of image attraction or other external forces, a stage of regularization to constrain the previous deformation with respect to the *a priori* known shape of the contour, and finally a stage of re-positioning the contour using the theory of material systems.

Keywords: active contours, edge detection, segmentation, registration, regularization, material systems, free form deformation, differential geometry.

INTRODUCTION

Active contours, also known as snakes, were introduced as an interactive aid for the extraction of object contours in images [Kass87]. They introduce the notion of geometric deformation as a local deformation. Subsequent work on such contours has been in two areas: to simplify for the user the task of selecting the initial contour estimate [Neuenschwander94]; to control the evolution process of the contour [Delingette94], [Fua94], [Cohen96], and [Teboul97] from the given initialization.

Later, Free-Form Deformations (FFD), described in [Borrel91] and [Bechmann94], were developed to operate at a topological level by considering global deformations. Some works, [Aubert95], [Bechmann95], show the application domain of this method and the results obtained when preserving the topology.

In this paper, we consider both these aspects of active contours in the case of a particular class of images, those where the general overall shape of the structures to be extracted from the images are known in advance. This is often the case in biomedical image processing for example, where the typical structure of bones or organs is known *a priori* from atlases. In addition to simplifying the initialization process for the user, this model-based matching allows quantifying the deformation necessary for extracting the structure from the image. This information can be useful for further diagnosis.

Because it is important to consider both geometric and topological aspects, our model is based on built-in geometric and topological shape information which constrains the deformation of the initial model under the influence of image forces. In this paper, the topological constraint ensures that the contour be closed (current work is in progress to relax this constraint). In order to consider only intrinsic properties of the contour, the geometric model is based entirely on curvature. We are only interested in what a typical structure "looks like", not where it may be, nor what its orientation could be, nor at what scale it may occur in the image.

The following section very briefly covers some related work. We next present the geometric model and the topological constraint necessary to ensure closure of the contour under evolution. We then describe the evolution process, and present both synthetic control examples and an example from biomedical imagery. We conclude with some remarks on extensions currently in progress.

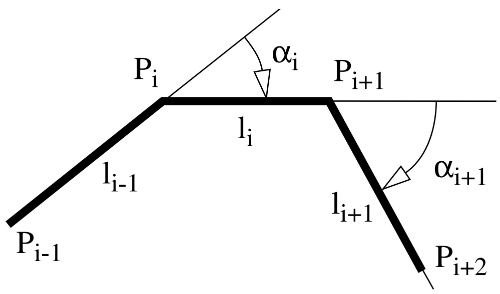
RELATED WORK

In the original snakes article [Kass87], the initialization is done by hand by drawing an initial state of the active contour near the image contour to be detected. Subsequent works exploited the evolution method to enable a more simple initialization. In [Neuenschwander94] and [Fua94], the initialization is reduced to a few characteristic points. In [Teboul97] and [Cohen96], the image contour only needs to lie inside the active contour. In [Menet90] and [Yuille90], the initialization is driven by the parametric curve describing the active contour.

However, the evolution is driven mainly by the active contour's geometric characteristics. In [Kass87], [Teboul97] and [Cohen96], in absence of an image force, a closed active contour would converge to one of two well-known states - either it melts to one single point and disappears or it expands to infinity. In [Menet90] and [Yuille90], the active contour evolves with respect to the parameters of the model chosen. The work of [Delingette94] constraints the evolution to a known shape described by the regulation process itself.

MODEL

Intuitively, a contour in an image is a curve that delimits an object, and could be represented in various ways - by a hand-drawn curve, a broken line, a parametric curve and so on. Before introducing in the next section what we use as a contour, we need to introduce some basic definitions. Let us define the notion of a *form*, f, as a closed broken line described by an ordered set of



Definition of the characteristics of a form.
Figure 1

n pairs

$$f = \{(\alpha_i, l_i), i \in [0, n-1]\}$$

where α_i represents the angle of the contour at P_i, more precisely, the angle between the vectors $\overrightarrow{P_{i-1}P_i}$ and $\overrightarrow{P_iP_{i+1}}$, and l_i represents the distance between P_i and P_{i+1} (l_i is the length of the vector $\overrightarrow{P_iP_{i+1}}$), see Fig. 1. At point 0, we define α_0 as the angle between the vectors $\overrightarrow{P_{n-1}P_0}$ and $\overrightarrow{P_0P_1}$, at point n, we define l_{n-1} as the length of vector $\overrightarrow{P_{n-1}P_0}$. This definition of a form gives a uniform role to all points, i.e., there are no extremities on a form. We denote F_n the set of all forms of length n.

Let us now introduce the notion of a *placement*, p, as being the transformation that maps a form $f \in F_n$ into a set of $n+1$ 2-D points

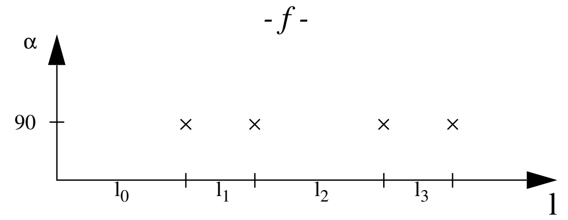
$$p\colon F_n \times \mathbf{R} \times \mathbf{R}^2 \to (\mathbf{R}^2)^{n+1}$$
$$p((f, a, G) \in \{(x_i, y_i) \in \mathbf{R}^2, i \in [0, n]\})$$

given the orientation a of the principal axis and the position G of the center of gravity that we want to give to the form in the 2-D plane of the image.

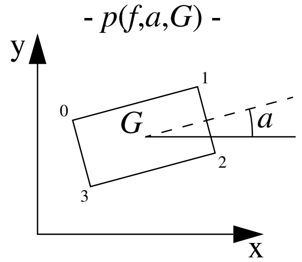
Fig. 2 shows a description of a form f of F_4 and its placement in the 2-D plane with an orientation a and a center of gravity G. This notion of placement comes directly from differential geometry where a curve in the plane is uniquely defined as a function of the curvature, with a given starting angle and position [doCarmo76]. We consider the center of gravity rather than a starting point and orientation in order to avoid the use of local geometric characteristics in the global notion of placement; that could result in increasing the instability of the whole process.

Finally, we define p^{-1}, the inverse transformation of the placement p, which produces the form f of F_n, the orientation a of the principal axis and the position G of the center of gravity, from a set of $n+1$ 2-D points

$$p^{-1}\colon (\mathbf{R}^2)^{n+1} \to F_n \times \mathbf{R} \times \mathbf{R}^2$$



Example of a rectangular form f. The illustration draws the discrete function of α with respect to $\sum l$



Example of the representation of the rectangle form f in the 2-D plane by the placement p.

Figure 2

Properties

The advantage of such a curvature-based contour representation is its invariance with respect to Euclidean transformations; whatever the position and orientation of a placement in the 2-D plane, p^{-1} will give the same form, f.

Closed curves rely on two essential characteristics. The first one is an intuitive property that states the fact that the first and the last point of a closed curve must superpose. Applied to the discrete closed curve forms we manipulate, this translates to the fact that the sum of the oriented vectors linking all pairs of successive points must be equal to 0.

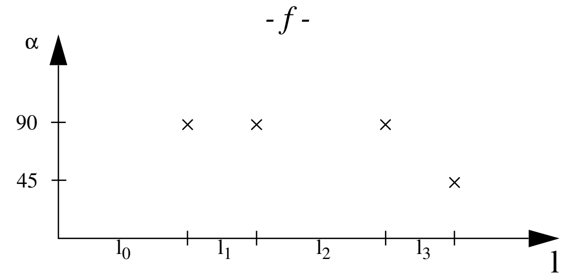
$$\sum_{i=0}^{n-1} \overrightarrow{P_i P_{i+1}} = \sum_{i=0}^{n-1} \mathrm{l}_i \cdot \begin{pmatrix} \cos \sum_{j=1}^{i} \alpha_i \\ \sin \sum_{j=1}^{i} \alpha_i \end{pmatrix} = 0 \qquad (1)$$

In this equation, we assume that $P_n = P_0$. In the 2-D case, this property gives a system of 2 equations expressed in the second members of Eq. 1.

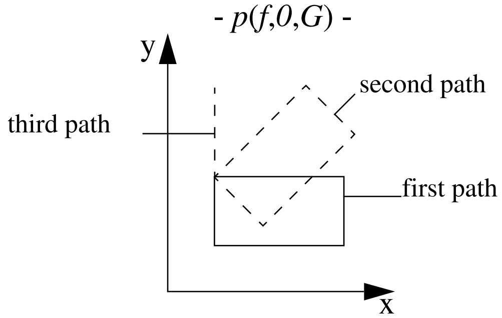
The second property states the fact that a closed curve can be traversed several times while following the same path. This characteristic is written in [doCarmo76] as

$$\int_{\Omega} \kappa(s) = 2\Pi \cdot I$$

where I denotes the *rotation index* of the curve.



Example of a rectangular curve. The illustration draws the discrete function of α with respect to $\sum l$



Example of the representation of the rectangle curve in the 2-D plane by the placement p. Notice the fact that, because of the value of α_3, there is an angle shift from one path to the following.

Figure 3

and applied to our discrete closed form this becomes

$$\sum_{i=0}^{n-1} \alpha_i = 2\Pi \cdot I \qquad (2)$$

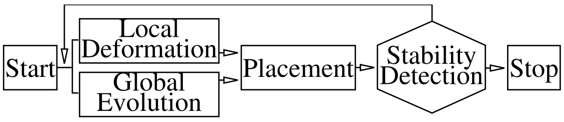
Notice that a curve that satisfies Eq. 1 is not a form if it does not respect Eq. 2. Fig. 2 illustrates a regular form while Fig. 3 highlights the case of a non closed curve. For the purpose of the discussion, we will denote by C_n the set of broken lines that are defined like forms but do not verify both conditions. On the set C_n, we define the following $+$ operator applied to two curves c and c'

$$+ : \mathrm{C}_n \times \mathrm{C}_n \to \mathrm{C}_n$$
$$c + c' = \{(\alpha_i + \alpha'_i, l_i + l'_i), i \in [0, n-1]\}$$

Notice that the addition of two forms f and f', or of a form f and a curve c does not necessarily give a curve that respects the conditions of a form.

CONTOUR

Now, we can introduce the contour model that we will be using all along this work. A contour consists of the following elements: a form f_o representing the reference form of the contour; a form f_d being the de-



Overview of the iterative process.
Figure 4

formed form that will be used to find an equilibrium state representing the contour in the image; an orientation a; and a 2-D position G.

EVOLUTION PRINCIPLE

The evolution of the active contour in the image field is performed as an iterative action of two separable processes. The first manages the local shape deformation of f_d under constraints with respect to the original shape f_o; the second manages the position with respect to the theory of material systems. In this section, we briefly sketch the processes that will be further detailed in the following sections.

The deformation of the contour is done with respect to the reference form f_o. To do so, after having applied the external constraints of the image, we process the error of the deformable form f_d with respect to f_o in the (α, l) coordinate system.

In addition to deformation, the active contour moves over the surface of the image under the applied forces. Motion management, the second part of the evolution process, computes the orientation and the position using the laws of material systems [Remion96].

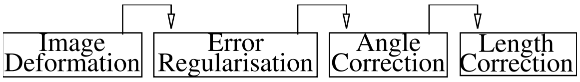
The iterative process is stopped either interactively or automatically with respect to some stability dectection. Fig. 4 shows an overview of the evolution process of the contour detection.

DEFORMATION PROCESS

In this part, we describe the deformation process to compute the form f_d at step $i+1$ considering the external and internal constraints. In order to deform the form f_d compared to the original form f_o, let us introduces a curve c_e such that

$$f_d = f_o + c_e$$

c_e can be regarded as the deformation of the form f_d compared to f_o, the reference form. To preserve similarity between f_o and f_d, the process of deformation will maintain smoothness of c_e through a regularization. Thus, a local external force applied to f_d will have an effect over some neighborhood. But, modify-



Details of the deformation process.
Figure 5

ing the curve c_e implies that the curve $f_o + c_e$ may no longer satisfy the constraints for a form (see the properties section). Let us call this temporary curve c_d. The last step of this part of the deformation process will be a closure of c_d to get the form f_d. Fig. 5 highlights the different stages of the deformation process.

Image interaction

First, the deformation of the model is applied to the form f_d placed in the 2-D image. To each point of the placed form, the image will apply a reaction R_i proportional to the derivative of the image gradient ∇I,

$$R_i = k_v \cdot \nabla|\nabla I(P_i)|$$

This reaction is interpreted as the displacement of the corresponding point and was already introduced with the original snakes [Kass87]. Finally, we compute the new form f_d, and then the error c_e.

Regularization

The point of regularization is to ensure a degree of resemblance between the current shape of the contour described by the form f_d and its original shape described by the form f_o. The regularization is performed on the error curve c_e at order 1, elasticity, and at order 2, plasticity, according to the following formula

$$[K + k_v \cdot \mathrm{Id}] \cdot X_{i+1} = k_v \cdot X_i \qquad (3)$$

where K is the regularization matrix, k_v, the viscosity coefficient and the vectors X_i and X_{i+1} are the data at step i and at step $i+1$, before and after regularization. More information about this regularization technique is available in [Kass87]. The matrix K is a band-diagonal matrix that contains on each line

$$\begin{aligned} K_{i,i-2} &= k_p \\ K_{i,i-1} &= -4k_p - k_e \\ K_{i,i} &= k_v + 6k_p + 2k_e \\ K_{i,i+1} &= K_{i,i-1} \\ K_{i,i+2} &= K_{i,i-2} \end{aligned}$$

The coefficients k_e and k_p, which act on elasticity and plasticity respectively, make it possible to balance the action of regularization on each criterion. Only relative values of k_e, k_p and k_v are important and will modify the process of regularization and as a result the evolution of the contour.

Adjustments

As we saw in the evolution principle section, the regularization does not lead to an expected form but to a curve. To restore the characteristics of this result, we must turn this curve into a form, applying corrections on it to satisfy the 2 conditions of a form expressed in the model section, Eq. 1 and Eq. 2. The constraints are applied to the angles and the lengths separately. We will adjust the angles first and then the lengths.

Angle Adjustment. The adjustment of the angles is distributed evenly over all the angles and is carried out by a simple shift of each angle by a single value,

$$\Delta\alpha = \frac{2\Pi - \sum \alpha_i}{n}$$

where $\forall i, \alpha_i = \alpha_i - \Delta\alpha$

We then obtain a form f_d which satisfies the condition Eq. 2.

Length Adjustment. To finish the process, it is necessary to adjust the lengths in order to close the curve c_d, thus satisfying the condition Eq. 1. To do so, we use the method of Lagrange multipliers [Luenberger69]. To describe the application of this method in our case, let us write $\hat{l}_i$ for the lengths obtained after regularization, l_i for the final lengths, and Δl_i for the correction to apply to each $\hat{l}_i$. We can then write the relation

$$l_i = \hat{l}_i + \Delta l_i \tag{4}$$

Let us define the function θ to be minimized as the sum of the squares of the errors $\theta(\Delta l_i) = \sum \Delta l_i^2$

The constraints to be applied are those expressed by the system of Eq. 1

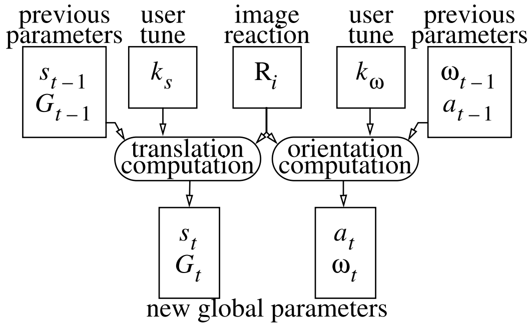
$$\varphi_1(\Delta l_i) = \sum_{i=1}^{n-1} l_i \cdot \begin{pmatrix} \cos\beta_i \\ \sin\beta_i \end{pmatrix} = 0$$

where $\beta_i = \beta_{i-1} + \alpha_i$ and $\beta_0 = 0$

The Lagrange method will solve the problem by calculating the Δl_i starting from

$$\Phi(\lambda_1, \lambda_2, \Delta l_i, \forall i) = \theta + \lambda_1 \varphi_1 + \lambda_2 \varphi_2$$

where λ_1 and λ_2 are the multipliers. From the system



Detailed scheme of the global evolution.
Figure 6

with $n + 2$ equations defined as follows,

$$\begin{cases} \dfrac{\partial\Phi}{\partial\Delta l_i} = 2\Delta l_i + \lambda_1 \cos\beta_i + \lambda_2 \sin\beta_i = 0 \\ \varphi_1 = 0 \\ \varphi_2 = 0 \end{cases}$$

Solving the system gives us the Lagrange multipliers λ_1 and λ_2. The calculation of each value Δl_i enables us to correct of all the lengths according to Eq. 4.

GLOBAL EVOLUTION

In order to move the active contour during the deformation process in accordance with the external forces, the parameters a and G must respect the laws of mechanics. In our application, in spite of the fact that our form is being deformed, we apply the principle of material systems [Remion96], where the center of gravity and the moment of inertia pertain to translation and rotation respectively. We apply the Newtonian laws of motion, expressed as

$$m\frac{\partial^2 X}{\partial t^2} + k\frac{\partial X}{\partial t} = F$$

to calculate the evolution of these two global parameters of the form. At each iteration, the two parameters of the form are computed and applied when placed in the 2-D plane. The global evolution is described in detail in Fig. 6.

Evolution of G

A global force R, the sum of all forces R_i applied to any point of the form in the image plane, will be applied globally on the form (independently of the local deformation). The objective is then not only to find the new position G_t of the center of gravity but also to calculate the new value of its translation speed s_t. The

Newtonian laws give the following differential equation (ode), according to the mass m of the system and the viscosity coefficient k_s

$$m\dot{s}_t + k_s s_t = \mathrm{R}$$

where $\dot{s}_t = ds_t/dt$

The solution is given by

$$s_t = \frac{\mathrm{R}}{k_s} + \left(s_{t-1} - \frac{\mathrm{R}}{k_v}\right)\exp\left(-\frac{k_s t}{m}\right)$$

with $v_0 = 0$ and $G_t = G_{t-1} + s_t t$

Thus, it will be easy to place the contour at the correct position by translating the center of gravity of the evolving form G_{t-1} to G_t.

Evolution of a

Each force R_i applied to each point of $p(f_d)$ creates a torque T_i which will tend to turn it around its center of gravity. The sum T of all the T_i represents the global torque applied. The goal is not only to find the orientation a_t of the contour but also to calculate the angular speed ω_t. The Newtonian laws determine the evolution from the following ode according to the mass m of the system and the viscosity coefficient k_v

$$m\dot{\omega}_t + k_\omega \omega_t = \mathrm{T}$$

where $\dot{\omega}_t = d\omega_t/dt$

ω and t being the angular speed and the applied torque respectively.

The solution is given by

$$\omega_t = \frac{\mathrm{T}}{k_\omega} + \left(\omega_{t-1} - \frac{\mathrm{T}}{k_v}\right)\exp\left(-\frac{k_\omega t}{m}\right)$$

with $\omega_0 = 0$ and $a_t = a_{t-1} + \omega_t t$

The computed angle a represents the orientation of the vector corresponding to the principal axis of inertia of the form. This axis corresponds to the eigenvector associated with the greatest eigenvalue of the matrix of inertia expressed relative to the center of gravity

$$I = \sum_{i=0}^{n-1} \begin{bmatrix} y_i^2 & x_i y_i \\ x_i y_i & x_i^2 \end{bmatrix}$$

Thus, it will be easy to orient the form correctly by rotating the contour from orientation a_{t-1} to a_t.

RESULTS

The snapshots in Fig. 7 show some results of this algorithm as a segmentation method and the additional information that is available to evaluate the deformation with respect to the original form. From left to right we present the initialization of the reference form in the image, the result obtained as a deformed form, the final equilibrium characteristics, i.e., the curvature and length. The maximum value of the characteristics appears in the upper left corner of the corresponding figure, and is expressed in radians for angles and in pixels for lengths. In the images,

(a) shows an ellipse matching an image of an ellipse of the same eccentricity,
(b) shows an ellipse matching an image of an ellipse of the same eccentricity, but with a different initialisation,
(c) shows an ellipse of eccentricity 0.6 matching an image of an ellipse of eccentricity 0.54,
(d) shows an ellipse matching an image of a square,
(e) shows a real example of an X-ray of a jaw where we have localized a tooth.

Result (a) is presented here to illustrate the accuracy of the method. Considering the fact that the synthetic image contains a digitized ellipse with all the problems involved in calculating the gradient, we obtain an accuracy of 0.03 radians on angles and 0.06 pixels on lengths.

Result (b) presents the same case as result (a) except that the initial orientation of the contour is 20°. As we can observe, the final contour is near the image contour, but we can deduce from the comparison with result (a) that the orientation is not found exactly, in fact its value is about 5°.

In result (c), an ellipse of eccentricity 0.6 is dipped into an image of an ellipse of eccentricity 0.54. We observe that the matching is well done. By studying the characteristics, we see exactly where the contour has to force or to relax its curvature to obtain a good match. For instance, between markers 0 and 3, and 1 and 2, the curvature has to be increased, between markers 0 and 1, and 2 and 3, the curvature has to be decreased.

Result (d) proposes an extreme case where an ellipse has to match a square. We observe that the equilibrium of the deformed contour does not match the image contour. The constraints to apply on the contour to match the square are too strong with respect to the *a priori* model used.

Result (e) shows a real case where a tooth, an incisor, is localized in an X-ray image of a jaw. Regarding segmentation, the positionning is good. With the help of the characteristics, we are able to localize deformations between the original form and the contour that may be interesting for further medical analysis.

The results contain much information: shape of the segmented contour, location of the object giving the position and orientation, and a local evaluation of the deformation with respect to the reference form. This last point has to be continued with the development of specific tools. In all the above results, it is most important to separate the two somewhat conflicting notions of local and global deformation, i.e., the two notions of geometry and topology. While preserving the topology of the contour, our process enables the geometry to evolve under constraint. The exact values of the constraint however, remain to be further studied.

CONCLUSIONS

In this paper, we have presented a technique for introducing *a priori* shape information into active contours in order to constrain the image-driven deformation during the matching process. This information is also useful for quantifying the deformation applied to a typical model necessary to match an image structure.

This article represents very recent work, and evidently it will be necessary to apply this method to a significant number of images to build an accurate data-base of forms corresponding to real cases.

In addition to the object segmentation presented in this article, we are working on similar methods for generating animated images. In this case, we constrain the initial model to pass through certain given positions and shapes. Tracking the evolution thus produces a smooth animated sequence of images.

The final stage of this work is to generalize our methods to volume data in order to carry out processing in 3-D and 4-D.

ACKNOWLEDGMENTS

Thanks to Dr. Jean-Paul Alibert for the dental stereotypes used during the development of the test program.

REFERENCES

[Aubert95] Fabrice Aubert and Dominique Bechmann. Déformation d'objets à volume constant. In *Proc. of AFIG*, pages 149–158, 1995.

[Bechmann94] Dominique Bechmann. Space deformation models survey. *Computers and Graphics*, 18(4):571–586, July–August 1994.

[Bechmann95] Dominique Bechmann, Yves Bertrand, and Sylvain Thery. Déformation continue de forme libre. In *Proc. of AFIG*, pages 87–95, 1995.

[Borrel91] P. Borrel and D. Bechmann. Deformation of [n]-dimensional objects. *Internat. J. Comput. Geom. Appl.*, 1(4):427–453, 1991.

[Cohen96] L. D. Cohen and I. Cohen. Global minimum for active contour models: A minimal path approach. In *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 666–673, 1996.

[Delingette94] H. Delingette. Intrinsic stabilizers if planar curves. In Jan-Olof Ecklundh, editor, in *Proc. of ECCV'94*. Springer-Verlag, 1994.

[doCarmo76] M. P. do Carmo. *Differential Geometry of Curves and Surface*. Prentice Hall, 1976.

[Fua94] P. Fua, O. Kubler, W. Neuenschwander, and G. Szekely. Making snakes converge from minimal initialization. In *ARPA Image Understanding Workshop*, volume II, November 1994.

[Kass87] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *Academic Publishers*, 1987.

[Luenberger69] David G. Luenberger. *Optimization by Vector Space Methods*. John Wiley & Sons, 1969.

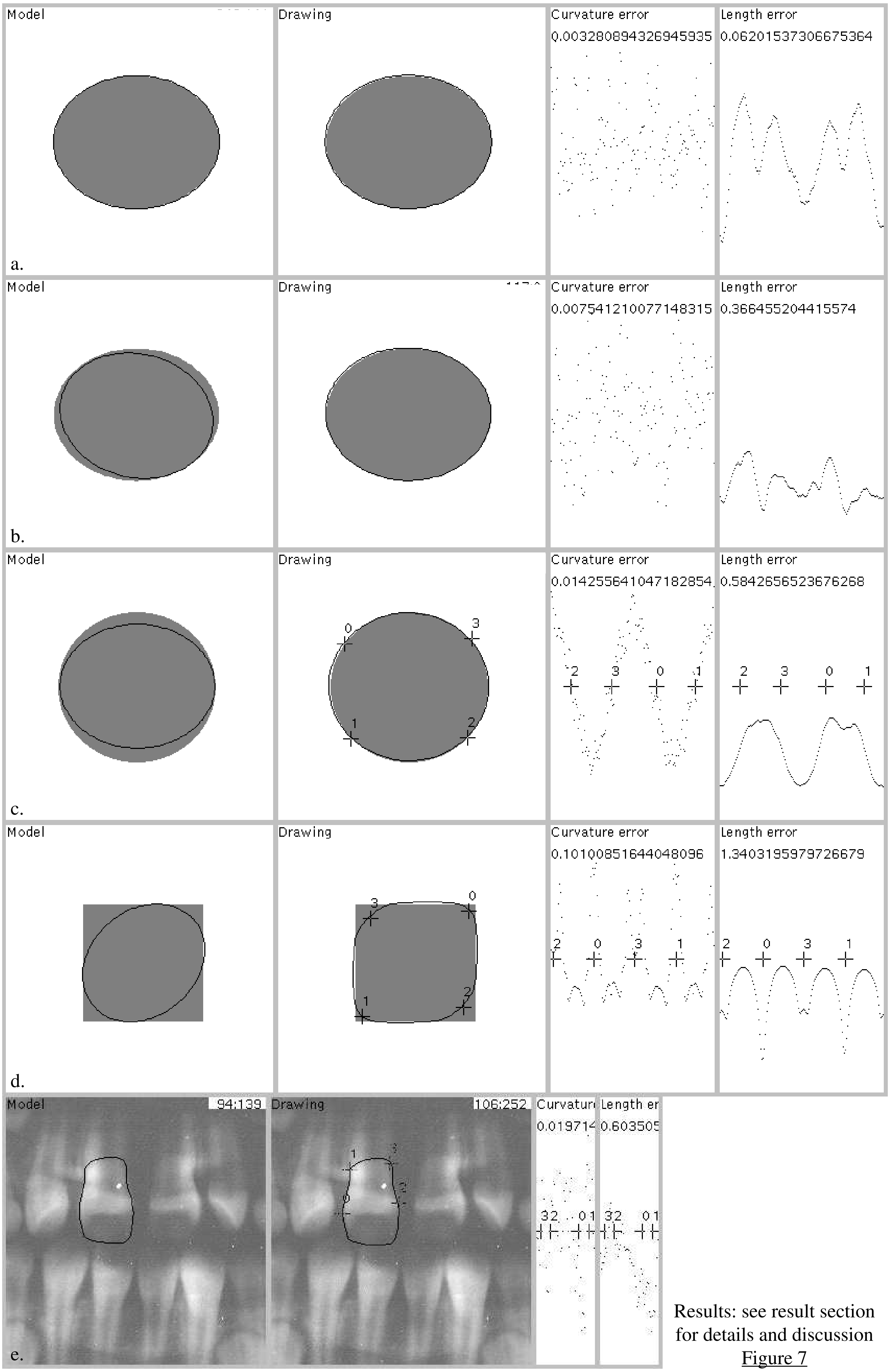
[Menet90] S. Menet, P. Saint-Marc, and G. Medioni. [b]-snakes: implementation and application to stereo. In *Proceedings DARPA Image Understanding Workshop*, September 1990. Pittsburgh.

[Neuenschwander94] W. Neuenschwander, P. Fua, G. Székely, and O. Kübler. Initializing snakes. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 658–663, June 1994. IEEE Computer Society Press.

[Remion96] Yannick Rémion. Systèmes matériels. Technical Report 96-09-01, LERI, September 1996.

[Teboul97] S. Teboul, L. Blanc-Féraud, G. Aubert, M. Barlaud, and J.Darcourt. Contours déformables et reconstruction tomographique en imagerie médicale. In *Gretsi97*, pages 1495–1498, 1997.

[Yuille90] A. L. Yuille, D. Cohen, and P. Hallinan. Facial feature extraction by deformable templates. Technical Report 88-2 Harvard University, Harvard Robotics Laboratory, 1990.



Results: see result section for details and discussion
Figure 7