Proc. VIIth Digital Image Computing: Techniques and Applications, Sun C., Talbot H., Ourselin S. and Adriaansen T. (Eds.), 10-12 Dec. 2003, Sydney

A Model Set Based Object Segmentation Method Using Level Set Approach

Xun Wang, Zhigang Peng, Feng Gao, William G. Wee

Department of Electrical and Computer Engineering & Computer Science, University of Cincinnati, Cincinnati, OH 45221, USA

Abstract. A novel approach for model set based object segmentation is described. The proposed method enables the using of a model set to guide the object segmentation. The object shape model selection and segmentation based on the selected model is formulated as an integrated process of constrained contour energy minimization. The solution derived from this formulation produces an integrated searching process consisting of two alternating procedures of contour evolution and contour model fitting. The process stops at a final contour together with a distance measure to the selected model contour for model selection. The resulting contour with the smallest distance is then selected as the final result. Model selection and segmentation results are finally reported.

Key words: Deformable contour method, level set, variational approach, curve evolution, polynomial shape representation method

1 Introduction

Model based object segmentation is an important issue in computer vision [2] [9]. A limitation for model based object segmentation is that it often requires a condition that a statistical shape model [2] [9] that can well represent the shape of target object is given. In the paper, we are concerned with an object segmentation problem using a model set consisting of more than one shape model. Specifically, we are concerned with a condition that a set of shape models, one of which well represents the shape of target object, is given to assist object segmentation. We will list a few scenarios that lead to the conditions:

Consider that we are segmenting objects using model based segmentation methods [9]. To do so, a statistical shape model, which is based on the whole training set of object shape instances, is required. However, due to the diversity of specific target objects, the statistical shape model may not represent the shape of a single target object well. With our approach, we can construct an object shape model set, in which each shape model represents a subset of object instances with similar shapes. The object segmentation is then to select the shape model most similar to the object to guide object segmentation. Model set based object segmentation also has a potential of working with other object recognition techniques [1] [8] to improve their recognition accuracy. Though object recognition techniques are rather effective in recognizing a large number of objects, they may have difficulties in *accurately* recognizing objects with reflectance, shadings, occlusions, and noises [8]. Since the model selection can be viewed as a process of object recognition and it is more robust to most segmentation difficulties, model set segmentation can potentially improve the recognition accuracy in these cases.

In the paper, we propose a model set based object segmentation method. We formulate the selection of object shape model and object segmentation based on the selected model as a constrained contour energy minimization problem [7]. Algebraic contour distance fitting error, algebraic geometric invariants distance measure between the contour and shape model using Mahalanobis distance [5], and object interior features are incorporated directly or as constraints to the contour energy minimization formulation. The solution derived from this formulation produces an integrated searching process consisting of two alternating procedures of contour evolution and contour model fitting. The process stops at a final contour together with a distance measure to the selected model contour for model selection. The resulting contour with the smallest distance is then selected as the final result. Model selection and segmentation results are encouraging.

2. Overview of Approach

The problem of model set based object segmentation is composed of two components: 1) shape model set construction 2) shape model selection and object segmentation based on the selected shape model.

To build shape model set, we select a few object instances for each shape model. To efficiently reflect the differences between the shape models, we select implicit polynomial shape representation approach [1] [4] [5]. The probability function of each parameters of the polynomial shape representation can then be defined. In the paper, to facilitate model set construction, we take only one object instance to build each shape model, which will be discussed in Section 3.

When the shape model set is built, a model based object segmentation method guided by the polynomial shape model is then applied for each model. The method is a model based deformable contour within a framework of constrained contour energy minimization [7]. The solution derived from the method produces two alternating processes of contour evolution and contour model fitting. The contour evolution is generally driven by the information of image gradient, object interior features, and estimated shape model. The contour model fitting dynamically combines the current contour with the shape model to generate an estimated shape model contour, which is then used in contour evolution. The algorithm stops at a resulting contour together with a distance measure to the selected model contour for model selection. We select the resulting contour with the smallest distance to one of the shape models as the final result and the shape model as what target object belongs. The philosophy can be explained as “shape model competition”, in which each shape model in the model set competes to guide the object segmentation or shape formation, and the shape model that wins the competition, which is evaluated by the distance to the resulting contour, is declared as the winner, and its associated contour is the resulting contour.

3. Shape Model Construction

To build shape model set, we choose polynomial shape representation method and then construct shape models using the polynomial shape representation.

3.1 Polynomial Shape Representation and Invariant Feature Selection of a Close Contour

Let $f(x,y)$ be a polynomial of degree n in (x,y) given by

$$f(x,y) = \sum_{\substack{0\le i,j \\ 0\le i+j\le n}} a_{ij}x^i y^j = a_{00} + a_{10}x + a_{01}y + a_{20}x^2 + a_{11}xy + \Lambda + a_{0n}y^n = X^T A \quad \mathbf{(1)}$$

where $A = [a_{00}\ a_{10}\ a_{01}\ \Lambda\ a_{0n}]^T$ is the coefficient vector consisting of l entries, and $X = [1\ x\ y\ x^2\ xy\ \Lambda\ y^n]^T$ is also an l-vector. A close contour can be implicitly defined as,

$$Z_f(A) = \{(x,y) : f(x,y) = 0\} \quad \mathbf{(2)}$$

This $Z_f(A)$ can also be denoted as $C(q)$ and be represented by a finite contour point set $\Gamma_0 = \{(x_i, y_i) : i = 1,2\Lambda\ m < \infty\}$. An algebraic fitting error measure of this point set Γ_0 to $Z_f(A)$ can be defined as

$$\varepsilon(\Gamma_0) = \frac{1}{m}\sum_{i=1}^{m} \mathrm{dist}^2((x_i, y_i), Z_f(A)) \quad \mathbf{(3)}$$

where $\mathrm{dist}((x_i, y_i), Z_f(A))$ is the distance from the point $(x_i, y_i) \in \Gamma_0$ to $Z_f(A)$, and ε is the average square distance. It can be shown that as $m \to \infty$, Eq. (3) can be written as the continuous form,

$$\varepsilon(C(q)) = \int_0^1 \mathrm{dist}^2(x(q), y(q), Z_f(A))\,dq \quad \mathbf{(4)}$$

As shown in [1] [4], minimizing Eq. (3) can be approximated by minimizing,

$$\varepsilon'_T = \sum_{i=1}^{m} f^2(x_i, y_i) \quad \mathbf{(5)}$$

Since a direct solution to minimizing ε'_T is easily trapped at local minima, the 3L fitting algorithm of [4] is used by adding two additional terms to ε'_T resulting in ε''_T as

$$\varepsilon''_T = \varepsilon'_T + \varepsilon_{c_1} + \varepsilon_{c_2} \quad \mathbf{(6)}$$

with $\varepsilon_{c_1} = \sum_{\substack{i=1 \\ (x_i,y_i)\in\Gamma_\tau}}^{m_\tau} [f(x_i, y_i) - \tau]^2$, and $\varepsilon_{c_2} = \sum_{\substack{i=1 \\ (x_i,y_i)\in\Gamma_{-\tau}}}^{m_{-\tau}} [f(x_i, y_i) + \tau]^2$, where ε_{c_1} and ε_{c_2} are the two additional fitting error terms (similar to Eq. (5)) for an outside point set $\Gamma_{-\tau}$ to the outer contour ${}_{-\tau}Z_f = \{(x,y) : f(x,y) = -\tau\}$ and an inside point set Γ_τ to the inner contour ${}_{\tau}Z_f = \{(x,y) : f(x,y) = \tau\}$, respectively, satisfying,

$$\mathrm{dist}((x_i, y_i), Z_f(A)) = \tau \qquad \text{if } (x_i, y_i) \in \Gamma_{-\tau}, \text{or } \Gamma_\tau, \quad \mathbf{(7)}$$

with m_τ being the number of points from Γ_τ, and $m_{-\tau}$ being the number of points from $\Gamma_{-\tau}$. The value ε_T'' of Eq. (6) can also be expressed in matrix notation as

$$\varepsilon_T'' = \|X_{3L}A - b\|^2 \quad \textbf{(8)}$$

where $X_{3L} = \begin{bmatrix} X(\Gamma_{-\tau}) \\ X(\Gamma_0) \\ X(\Gamma_\tau) \end{bmatrix}$, $b = [-\tau, \ldots -\tau, 0, \ldots 0, \tau, \ldots \tau]$, X_{3L} is an m_T by l matrix with the size, $m_T = m_{-\tau} + m + m_\tau$. $X(\Gamma_{-\tau})$, $X(\Gamma_0)$, and $X(\Gamma_\tau)$ are matrices containing $m_{-\tau}$, m, and m_τ points of X^Ts from $\Gamma_{-\tau}$, Γ_0, and Γ_τ, respectively, with $m_{-\tau}$ and m_τ points being uniformly spaced (preferably) on $\Gamma_{-\tau}$ and Γ_τ, and Γ_0 the finite contour point sets as noted earlier. Likewise, b is an m_T vector having first $m_{-\tau}$ entries of $-\tau$ to be followed by m entries of 0, and the last m_τ entries of τ. Furthermore, shape features $\{\alpha_1, \alpha_2, \ldots, \alpha_k\}$ invariant under rotational transformations are derived from A [6] as,

$$\alpha_i = F_i(A); i = 1, 2 \ldots k \quad \textbf{(9)}$$

where $k = \frac{1}{2}(n+1)(n+2) - 1$ is the total number of rotational invariant features, with $\left(\left\lceil \frac{n}{2} \right\rceil + 1\right)^2$ linear and quadratic invariants [6], and $k - \left(\left\lceil \frac{n}{2} \right\rceil + 1\right)^2$ relative angle invariants, and n is the highest order of the polynomial (see Eq. (1)). Let $\overline{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_k]^T$. The similarity between two shapes represented by shape features $\overline{\alpha}$ from A and model shape features $\overline{\alpha^*}$ from the model coefficient vector A^* can be measured by the Mahalanobis distance of

$$D_M = (\overline{\alpha} - \overline{\alpha^*})^T \psi^* (\overline{\alpha} - \overline{\alpha^*}) \quad \textbf{(10)}$$

where ψ^* is the information matrix of the model shape $\overline{\alpha^*}$ [6]. Approximated by the first order Taylor series expansion around A^*, Eq. (10) can be written as,

$$D_M = (A - A^*)^T Q(A^*)(A - A^*) \quad \textbf{(11)}$$

with $Q(A^*) = F'(A^*)^T \psi^* F'(A^*)$.

D_M is a shape distance measure between two contours using rotational invariant features after a proper position translation and size normalization operation. It will be shown in Section 4 that D_M can be used both as a guide to the progress of the contour marching operation in gaining shape matching and ultimately as a measure for the shape model selection decision. Notice that both contours have to translate their centers to the origin and size normalized before D_M is computed.

3.2 Shape Model Construction

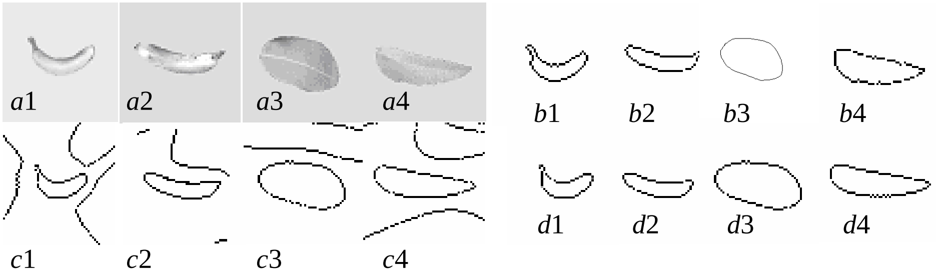


Figure 1 The images of row *a* are initial images (Banana1, Banana2, Leaf1, Leaf2). The images of row *b* are the corresponding manually drawn model contours. The 6th degree implicit polynomial fitting results are presented on the row *c*. The row *d* images are the clean up fitting results

To construct shape model, we first manually draw a model contours, then fit an implicit polynomial function according to Eq. (8), and finally select the shape features according to Eq. (9). Let k = 27 in Eq. (9) be the total number of invariant shape features of a 6th order polynomial with l= 28. The model information matrix ψ^* is the inverse of a covariance matrix of $\overline{\alpha^*}$ which can be estimated by sampling a Gaussian perturbation of the contour model [6]. Here, a sample size of 150 is used on a standard deviation of 0.01 perturbation. At a given iteration, both the current deformed contour and the model contour have to be translated to the same center with equal size before using Eq. (11) to compute D_M [1].

After we construct the shape models, the distinguished ability of the 6th order polynomial implicit representations through Mahalanobis distance has to be evaluated before any applications shown in Table 1.

Table1. The Mahalanobis distance between the models. Here, Banana1*, Banana2*, Leaf1*, Leaf2* is the original shapes after rotation of 45 degree.

	Banana1*	Banana2*	Leaf1*	Leaf2*
Banana1	5.6	17.6	91.8	56.7
Banana2	15.2	3.3	45.0	58.2
Leaf1	78.4	65.7	7.8	68.4
Leaf2	35.4	33.5	28.1	6.2

4. Model Set Based Object Segmentation

In this section, a model set based object segmentation approach is formulated. As discussed in Section 4.1, invariant features are used to define a distance between an input contour and model contour using Mahalanobis distance. This minimum distance measure is used as the shape model selection as the class belonging of an extracted contour to one of G models. For ease of presentation, a single model $Z_f(A^*)$ contour with rotational invariant features $\overline{\alpha^*} = \overline{\alpha}_g(A^*)$ is considered, and a complete model selection will be presented in the applications in Section 5.

Let $C(q,t)$ be a deforming close contour with interior $\Omega_c(t)$ at time t and parameter q, its polynomial representation be the coefficient vector A.

$$Z_f(A) = \{(x,y) : f(x,y) = X^T A = 0\}$$

The coefficient vector A is determined as discussed in Section 3. An algebraic fitting error measure between the two sets, $C(q,t)$ and $Z_f(A)$, can be shown as

$$\int_0^1 \text{dist}^2(x(q,t), y(q,t), Z_f(A))dq \le T_f \qquad \textbf{(12)}$$

where $(x(q,t), y(q,t)) \in C(q,t)$, and $T_f > 0$ is a threshold.

Our problem is to find $C(q,t)$ such that

$$E(C(q,t)) = w \oint_{C(q,t)} g(|\nabla I(C(q,t))|)ds + (1-w)(\overline{\alpha} - \overline{\alpha^*})^T \psi_g^{\ *}(\overline{\alpha} - \overline{\alpha^*}), 0 < w < 1 \qquad \textbf{(13)}$$

is minimized subject to the following constraints:

i) $$D(x,y) \ge T_V \quad \text{if } (x,y) \in \Omega_c(t) \qquad \textbf{(14)}$$

ii) $$\int_0^1 \text{dist}^2(x(q,t), y(q,t), Z_f(A))dq \le T_f, \qquad \textbf{(15)}$$

where s is the arc length, q is the contour parameter, $I(x,y)$ is the image brightness at (x,y) and $g(|\nabla I(C(q,t))|) = \dfrac{1}{1 + |\nabla I(C(q,t))|^2}$. $\nabla I(C(q,t))$ is the gradient of $I(x,y)$ with (x,y) on $C(q,t)$, $D(x,y)$ is any function characterizing the interior of expected target contour. Specific description of $D(x,y)$ will be provided later. T_V is a positive threshold.

To solve the problem, we take a Lagrange approach by minimizing $L_g(C(q,t),A)$

$$L_g(C(q,t),A) = w \oint_{C(q,t)} g(|\nabla I(C(q,t))|)ds + (1-w)\left\{(\overline{\alpha} - \overline{\alpha^*})^T \psi^* (\overline{\alpha} - \overline{\alpha^*})\right\}$$
$$- \lambda_1 \iint_{\Omega_C} [D(x,y) - T_V] dxdy + \lambda_2 \int_0^1 [\text{dist}^2(x(q,t), y(q,t), Z_f(A)) - T_f] dq \quad (16)$$

where $\lambda_1, \lambda_2 > 0$.

Since $\dfrac{\partial L_g(C(q,t),A)}{\partial t} = \dfrac{\partial L_g}{\partial C}\dfrac{\partial C}{\partial t} + \dfrac{\partial L_g}{\partial A}\dfrac{\partial A}{\partial t}$, (16') a direct computation is complicated and unnecessary. Here, we use a practical approach of partitioning the computation into two components. Each component is optimized while holding other component unchanged. Our first step is to hold A in Eq. (16') constant and deform $C(q,t)$. We consider a level set surface $D_f(x,y)$,

$$D_f(x,y) = \begin{cases} \mathrm{dist}^2((x,y), Z_f(A)) & \text{if } (x,y) \text{ is inside } Z_f(A) \\ 0 & \text{if } (x,y) \in Z_f(A) \\ -\mathrm{dist}^2((x,y), Z_f(A)) & \text{if } (x,y) \text{ is outside } Z_f(A) \end{cases} \tag{17}$$

It can be shown that minimizing Eq. (16) is equivalent to minimizing

$$L_f(C(q,t),A) = w \oint_{C(q,t)} g(|\nabla I(C(q,t))|)ds + (1-w)\left[(\overline{\alpha} - \overline{\alpha^*})^T \psi^* (\overline{\alpha} - \overline{\alpha^*})\right]$$
$$-\lambda_1 \iint_{\Omega_C(t)} (D(x,y) - T_V)dxdy - \lambda_2 \left[\iint_{\Omega_C(t)} D_f(x,y)dxdy - T_f\right] \tag{18}$$

According to [7], by minimizing Eq. (18), a curve evolution formula can be produced as

$$\frac{\partial C(q,t)}{\partial t} = \left[\lambda_1 [D(x,y) - T_V] + wkg(|\nabla I|) - w(\nabla g \cdot \vec{N})\right]\vec{N} + \lambda_2 D_f(x,y)\vec{N} \tag{19}$$

where k is the curvature of $C(q, t)$.

Our next step is to hold $C(q,t)$ constant and minimize Eq. (16') by adjusting A. In an effort to compute $\frac{\partial L_g}{\partial A}$, we return to the finite point set representations of both searched contour $C(q,t)$ and model contour.

Setting $\frac{\partial L_g}{\partial A} = 0$, we have

$$A = \left[\lambda_2' X_{3L}^T X_{3L} + (1-w)Q(A^*)\right]^{-1} \left[(1-w)Q(A^*)A^* + \lambda_2' X_{3L}^T b\right] \tag{20}$$

Notice that $\lambda_2' = m_T \lambda_2$ where m_T is the total number of points in Γ_0, Γ_τ, and $\Gamma_{-\tau}$.

Discussion: It is easy to see that the minimization of $L_g(C(q,t), A)$ is composed of two steps. The first step is the deformation of $C(q,t)$ driven by the curve evolution formula of Eq. (19) using an initial $Z_f(A)$. In this deformation, the contour point velocity is determined by the combined effect of region features $[D(x,y) - T_V]$, contour gradient function $\{wkg(|\nabla I|) - w(\nabla g \cdot \vec{N})\}$, and shape fitting measure, $[D_f(x,y) - T_f]$, between $C(q,t)$ and $Z_f(A)$. The second step is the recalculation of "A" using Eq. (20). In here, the resulting "A" is to produce a new $Z_f(A)$ that is close to both $C(q,t)$ and $Z_f(A^*)$. Then as the velocity in Eq. (19) is very small, and "A" in Eq. (20) remains the same with very little change from the earlier iteration value, the algorithm converges. Notice that the shape fitting measure ε_T is only trying to deform $C(q,t)$ toward $Z_f(A)$ without any size consideration. Therefore, if a larger weighting factor λ_2 or λ_2' is used, it can impede the speed of outward marching of $C(q,t)$ and lower the convergent speed.

For our implementation in the next section, we need to specify $D(x,y)$ letting

$$D(x,y)=\frac{1}{1+\left|\nabla G*I\right|^2}e^{-\frac{\left|I(x,y)-I_0\right|}{\sigma}}$$

where $\left|\nabla G*I\right|$ is the absolute value of the gradient of $I(x, y)$ smoothed by a Gaussian filter $N(0, \sigma_1^2)$. I_0 is the average intensity inside target boundary.

5. Algorithm and Experiments

The sensitivities between λ_2' and w settings in Eq. (20) are of importance to our computation of "*A*". With the model and object contour shown in Figures 2.1a and 2.1b, respectively, results are shown in Figure 2.2a to 2.2f using different λ_2' values. By holding w constant at 0.5, with large $\lambda_2'>1$, the resulting "*A*" and their respective contours are closer to the object and with the extreme case of $\lambda_2'=10$, the influence of the model does not exist, and with much small $\lambda_2'<<0.01$, they are closer to the model in the extreme case with little influence from the image data. A setting of $\lambda_2'=0.001$ is selected, and it is selected such that a balance influence between the object contour and the model contour is achieved. This setting will be dependent upon the shape configuration of selected model. The parameters needed to be set for the Eq. (19) are σ, λ_1, λ_2, and *w*.

Segmentation algorithm:

Step 1. Select a 5 by 5 contour inside the target object as initial contour $C(q,0)$. Keep $I_0=\hat{I}_0^{C(q,0)}$ constant, where $\hat{I}_0^{C(q,0)}$ is the average intensity inside the initial contour $C(q,0)$. With $C(q,0)$ and the given shape model, compute "*A*" according to Eq. (20).

Step 2. Evolve contour $C(q,t)$ by solving Eq. (19) using narrow band level set method for *l* iterations. Stop when the difference between two successive iterations is insignificant or a maximum of iteration number n_m has been reached.

Step 3. With the contour $C(q,t)$ given by Eq. (19), compute "*A*" according to Eq. (20).

Step 4. Update $I_0=\hat{I}_0^{C(q,t)}$, where $\hat{I}_0^{C(q,t)}$ is the average intensity inside contour $C(q, t)$. Go to Step 2.

These are two parts to our experiment. Part1 is to illustrate the segmentation capability of the algorithm. Here, we use the leaf with an "X1" mark as in the model shown in Figure 3(a). The final segmentation result of leaf with a "Y12" mark in Figure 3(a) is shown in Figure 3(b) with an initial inside contour. Figure 3(c) shows the resulting segmentation contour using the same velocity equation of Eq (19) with $\lambda_2=0$, i.e., there is no model. Similarly, using "X2" of Figure 3(d) as model provides a segmentation result of "Y22"on Figure 3(e). As for the illustration of the model selection result, we compute the D_m value of 4 leaves data, "Y11" and "Y12" for class 1 in Figure 3(a), and "Y21" and "Y22" for class 2 in Figure 3(a). The resulting D_m using different models deforming targets and classification decisions are shown in Table 2.

We also applied the method for extracting the external boundaries of intracranial in MRI brain images as shown in Figure 4.

Table2 Model Selection Result

Data(Deforming Model)	D_M	Class Decision
Y11 (Model1)	14.1	Class 1
Y11 (Model2)	30.7	
Y12 (Model1)	15.7	Class 1
Y12 (Model2)	38.0	
Y21 (Model2)	16.0	Class 2
Y21 (Model1)	30.6	
Y22 (Model2)	12.9	Class 2
Y22 (Model1)	22.4	

7. Conclusion

We have introduced a model set based object segmentation method based on the framework of constrained contour energy minimization. With the usage of polynomial shape representation, the algorithm can select from a shape model set the proper object model to guide object segmentation.

References

1. G. Taubin, “Estimation of Planar Curves, Surfces and Nonplanar Space Curves Defined by Implicit Equations, with Applications to Edge and Range Image Segmentation,” *IEEE Trans. on PAMI*, vol. 13, no. 11, pp. 1,115-1,138, Nov. 1991.
2. A. Chakraborty, and J. Duncan, “Game-Theoretic Integration for Image Segmentation” *IEEE Trans. On PAMI*, Vol. 21 No. 1, pp. 12 -30, Jan. 1999.
3. R. Malladi, J. Sethian and B. Vemuri, “Shape Modeling with Front Propagation”, *IEEE Trans on PAMI*, Vol. 17, No.2, pp. 158-171, Feb. 1995.
4. M. Blane, Z. Lei, H. Civi, and D. Cooper, “The 3L Algorithm for Fitting Implicit Polynomial Curves and Surfaces to Data” *IEEE Trans. on PAMI*, Vol. 22, No. 3, pp. 298 -313. Mar. 2000.
5. J. Subrahmonia, D.B. Cooper, and D. Keren, “Practical, Reliable, Bayesian Recognition of 2D and 3D Objects Using Implicit Polynomials and Algebraic Invariants,” *IEEE Trans. on PAMI*, vol. 18, no. 5, pp. 505-519, May 1996.
6. J. P. Tarel and D. B. Cooper, “A New Complex Basis for Implicit Polynomial Curves and Its Simple Exploitation for Pose Estimation and Invariant Recognition,” *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 111-117, 1998.
7. X. Wang, L. He, Y. Han, and W. Wee, “A Constrained Optimization Approach to Deformable Contour Method", pp. 183-192, British Machine Vision Conference, 2002.

8. T. Sebastian, P. Klein, and B. Kimia, “Recognition of Shapes by Editing Shock Graph,”, pp. 755-762, *IEEE ICCV* 2001.
9. L. Staib and J. Duncan, "Boundary Finding with Parametrically Deformable Models,", IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(11), 1061 (1992).

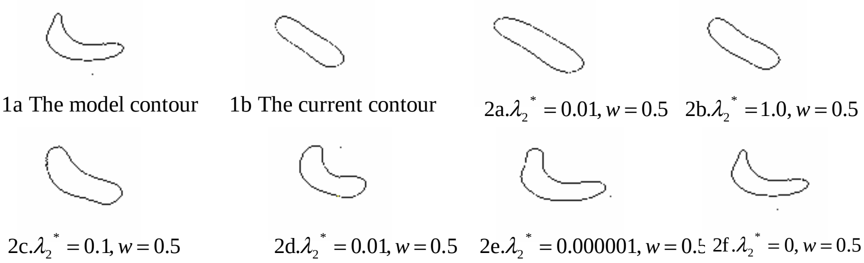
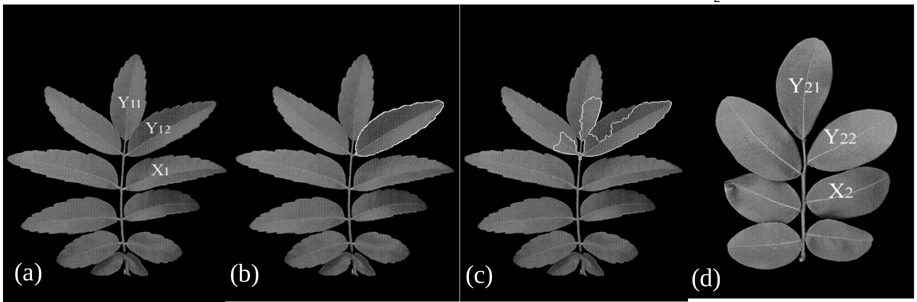


Figure 2 The new generated contour when using different λ_2^* and w



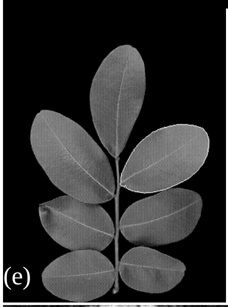


Figure 3 (a) Original image for segmentation (b) The segmentation results of (a) using shape model (c) The segmentation resulting without shape model. (d) Another original image. (e) Segmentation results of (d).

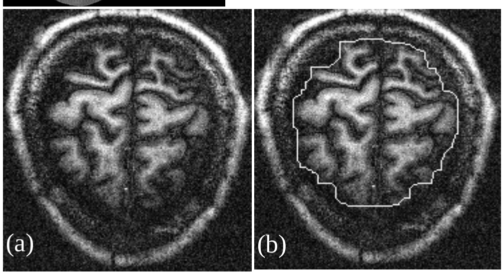


Figure 4 (a) Original MRI brain image (b) Extracted intracranial contour