M.Tech. (Computer Science) Dissertation Series.

OBJECT TRACKING USING STRESS INDUCED SURFACE SEEKING MODEL

A dissertation submitted in partial fulfillment of the requirements for the Master of Technology (Computer Science) degree of the Indian Statistical Institute.

By

Sitansu Kumar Das

Under the supervision of

Dr. Dipti Prasad Mukherjee

INDIAN STATISTICAL INSTITUTE

203, Barrackpore Trunk Road

Kolkata-700108

2003

CERTIFICATE OF APPROVAL

This is to certify that the thesis entitled '**Object Tracking Using Stress Induced Surface Seeking Model**' is an authentic record of dissertation carried by **Sitansu Kumar Das** under my supervision and guidance. The work fulfils the requirements for the award of the M.Tech. degree in Computer Science.

Dated: The 1st of July, 2003.

Signed:

(Dr. Dipti Prasad Mukherjee)

Supervisor

Contents

ACKNOWLEDGEMENT

I take this opportunity to express my deep sense of gratitude and indebtedness to my guide **Dr. Dipti Prasad Mukherjee** of Electronics and Communication Science Unit of Indian Statistical Institute, Kolkata for his generous and whole hearted support to me in completing this project.

I also wish to place on record my sincere thanks to Research Scholars of electronics and Communication Science Unit for their kind help and cooperation that they extended to me throughout the tenure of my project

I would like to express my profound gratitude to all other teachers of Indian Statistical Institute, Kolkata and sincere thanks to all of my classmates who were the source of inspiration during the M.Tech programme.

Sitansu Kumar Das

OBJECT TRACKING USING STRESS INDUCED SURFACE SEEKING MODEL

Abstract

Detection and tracking of object is one of the basic goals in video image processing. We use a region based active contour model. The proposed model is a modified version of active snake model by Kass et al. (1987) [1]. While the original active contour model relies only on edge gradient force, we have designed stress density model for the image region within the object. The image space is assumed as deformable metal plate. The stress distribution on the plate due to presence of an object on the plate is modeled. The intensity of pixels representing object forms the surface where stress energy is calculated. The performance of the proposed algorithm is attractive as it introduced a range of the external force term in this model. We have applied our technique to segment synthetic and real images. It is also shown that the algorithm can successfully track objects in a real video sequence.

1 Introduction

Detection and tracking of an object in a movie sequence is one of the most interesting and important tasks in image processing and computer vision field. It is basically an image segmentation process. Multitudes of techniques have been proposed in order to approach this problem such as Snake: Active contour model [1][2], Deformable surface spine model [3], Morphological Analysis with Active Nets [4], Anticipating snake model [5]. In general techniques can be classified in two broad categories:

i) Satisfying homogeneity property in image features over a large region (for example: Anticipating snake model [5]).
ii) Detecting abrupt changes in image features within a small region (for example: Snake: Active contour model [1][2]).

The first approach extracts the region as a whole within which some measure show the homogeneity of image feature. The second one detects the edge between two regions. A particularly attractive idea consists of obtaining a model of segmentation that integrates the behavior of two image segmentation processes stated above. In the proposed model in this paper, active snake model is improved by modifying the external force terms, which makes the model exploit region and contour property rather than only contour property.

In proposed model, it is assumed that image is an elastic metal plate that is deformed towards important image features. Alternatively it can be explained as follows. Without the tracking object image is assumed to be a plane undeformed elastic metal plate that has no internal stress. Internal stress is induced due to deformity made by external forces on the metal plate. All the points above the plate are situated in a particular constant low elevation resulting lower potential energy at every point of the plate. So without the presence of the object strain energy and potential energy of each point of the plate is negligible. However, when the object comes in the picture it changes the intensity value of the region it represents by a considerable amount than its surrounding region so that viewer can distinguish the object. As a result the metal plate that is assumed to be undeformed becomes distorted due to the introduction of the object. Surrounding region of the object is assumed as mildly stressed region. The region represented by object inside the image plane becomes a high stressed and elevated zone in terms of image intensity compared to its surrounding region. As a result energy density for the object region proportional to the elevation of points therein, is higher than that of its surrounding region.

From the above discussion, it is clear that inside an image a distinct object means a high area consist of high stress energy density and its surrounding area consists of low stress energy density. So if segmentation is performed between the object and its surrounding, homogeneity of energy density can be taken as discriminating feature. The mechanics of this segmentation process is guided by a variable elastic contour capturing the region of the image for which stress energy density is maximum. The high stress region represents object. In case of tracking of object, the contour is approximated by the user near the object in the first image frame. The contour encloses the object by seeking high stress energy density zone. In the next frame the contour is placed in same position where the existence of object is found in first frame. In second frame starting from that position contour seeks again for high stress energy density zone until it is found. In the third frame contour is initialized at the position where it gets object in second frame and searches for the object. This process continues in fourth and e frames onwards until the object in all frames is tracked. In all the cases stated above it is assumed that the position of the object in two-consequence frame should have a common overlapping zone. otherwise algorithm fails to track the object.

To test the behavior of the model in a real domain it is applied to track the image of white blood cell flowing inside the blood vessel and good result is obtained.

The next section describes active contour model briefly. In section 3 the methodology of proposed model is presented and in section 4 some of the tested result is given. In appendix section detail derivation of the proposed model is incorporated.

2 A brief over view on Snake model [1][2][5]

The basic structure of active contour model utilized an elastic but rigid closed contour. This contour is guided by an internal energy that controls stiffness of the contour and an external energy that guides the contour towards object in the image. The contour can be defined as geometric mappings from material (parametric) co-ordinate domains into 2D Euclidian space $\Re^2$ by the following mapping M [3]:

$$M: s \Rightarrow c(s) = (X(s), Y(s)), \qquad ..(1.2)$$

where $s \in [0,1]$, is univariate material co-ordinates and c(s) represents contour. The contour or snake can move freely in image plane. Main goal of the movement of snake is the minimization of an energy term. The energy term is given by:

$$\varepsilon = \int_0^1 \left\{ \frac{1}{2}\alpha \left[\left(\frac{\partial X}{\partial s}\right)^2 + \left(\frac{\partial Y}{\partial s}\right)^2 \right] + \frac{1}{2}\beta \left[\left(\frac{\partial^2 X}{\partial s^2}\right)^2 + \left(\frac{\partial^2 Y}{\partial s^2}\right)^2 \right] \right\} ds - \int_0^1 F_{ext}\, ds \qquad ..(2.2)$$

Where α is elasticity parameter and β is rigidity parameter. In the above equation the first order derivative terms are strain energy terms and second order derivative terms are bending energy terms of the contour. F_{ext} is the external energy, which tries to bend the contour. Actually this external energy is the pixel intensity values on the contour. The negative sign before the term consisting of external energy implies that when the total energy term tends to minimize then the term containing the external energy tends towards maxima. The external energy term tries to distort the contour and shift the contour points outwards in positive gradient region while pull inwards in negative gradient region. The elasticity and rigidity terms try to keep the contour as stiff as possible in its previous position. When a trade off between these two opposing energy terms gets minima. the resultant contour gets its equilibrium position. This scenario is very much clear if result of minimization of above energy term is taken as the basis of analysis. If above energy term is minimized with the help of calculus of variation following two equations are derived [2]:

$$-\vec{f}_X + \mathbf{A}\,\vec{X} = 0 \qquad ..(3.2)^*$$

$$-\vec{f}_Y + \mathbf{A}\,\vec{X} = 0 \qquad ..(4.2)$$

where $\vec{f}_X$ and $\vec{f}_Y$ are gradient vector in X and Y direction on the contour points and $\vec{X}$ and $\vec{Y}$ are X co-ordinate vector and Y co-ordinate vector of the points on the contour respectively. **A** is the coefficient matrix may be called as rigidity/elasticity matrix [5]. These two equations give the extrema condition of total energy term for a contour position. As differentiation of energy term with respect to distance gives the force so equation (3.2) and (4.2) give the amount of force generated to move contour points in X and Y directions respectively. To find the set of points on the contour for which energy is minimum, gradient descent method used to equations (5.2) and (6.2). The corresponding numerical equations are:

$$\vec{X}^{t+1} = (\mathbf{I} + \mathbf{A})^{-1}\left(\vec{f}_X + \vec{X}^t\right)\Delta t \qquad ..(5.2)$$

$$\vec{Y}^{t+1} = (\mathbf{I} + \mathbf{A})^{-1}\left(\vec{f}_Y + \vec{Y}^t\right)\Delta t \qquad ..(6.2)$$

Where $\vec{X}^{t+1}$ and $\vec{Y}^{t+1}$ are the iterated points at (t+1)th iteration while $\vec{X}^t$ and $\vec{Y}^t$ are the previous set of points and **I** is an identity matrix. Here Δt is the time step by which amount of movement of contour is controlled for numerical stability. From the above equations it is clear that previous points incremented by gradient vectors give the new sets of points. So if the value of the gradient of intensity in image plane is positive coordinate values of the point are incremented and for the negative gradient this is reversed. This fact is illustrated in figs 1.2 and 2.2.

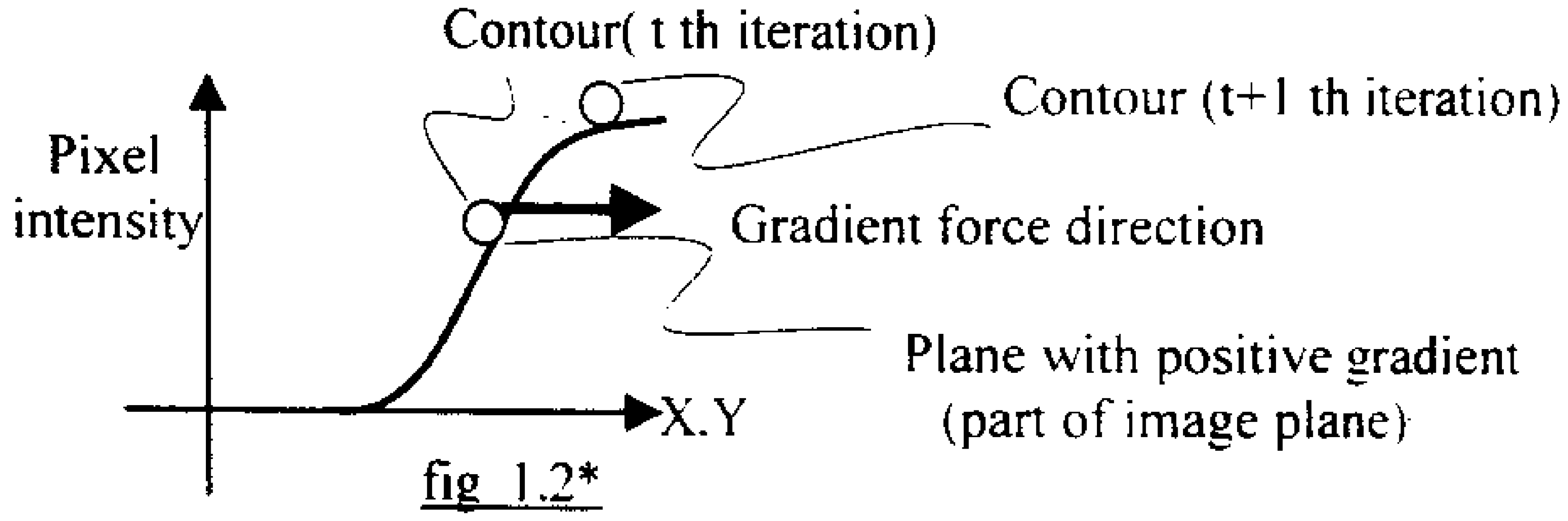


fig 1.2*

Figure shows the direction of gradient force activated on contour in positive gradient plane.

---

*First digit in equation number before point denotes serial number of equation in an article and second digit after point denotes the article number. Same convention is also followed for figure number.

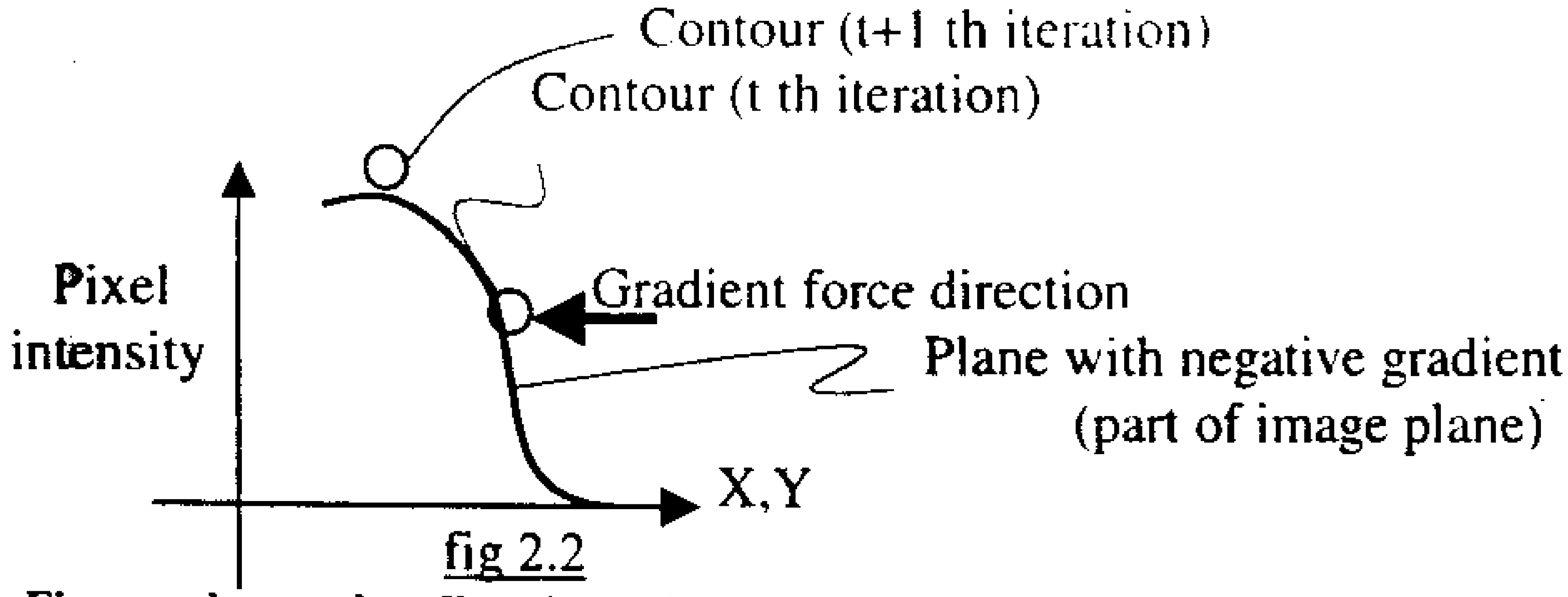


fig 2.2

Figure shows the direction of gradient force activated on contour in negative gradient plane.

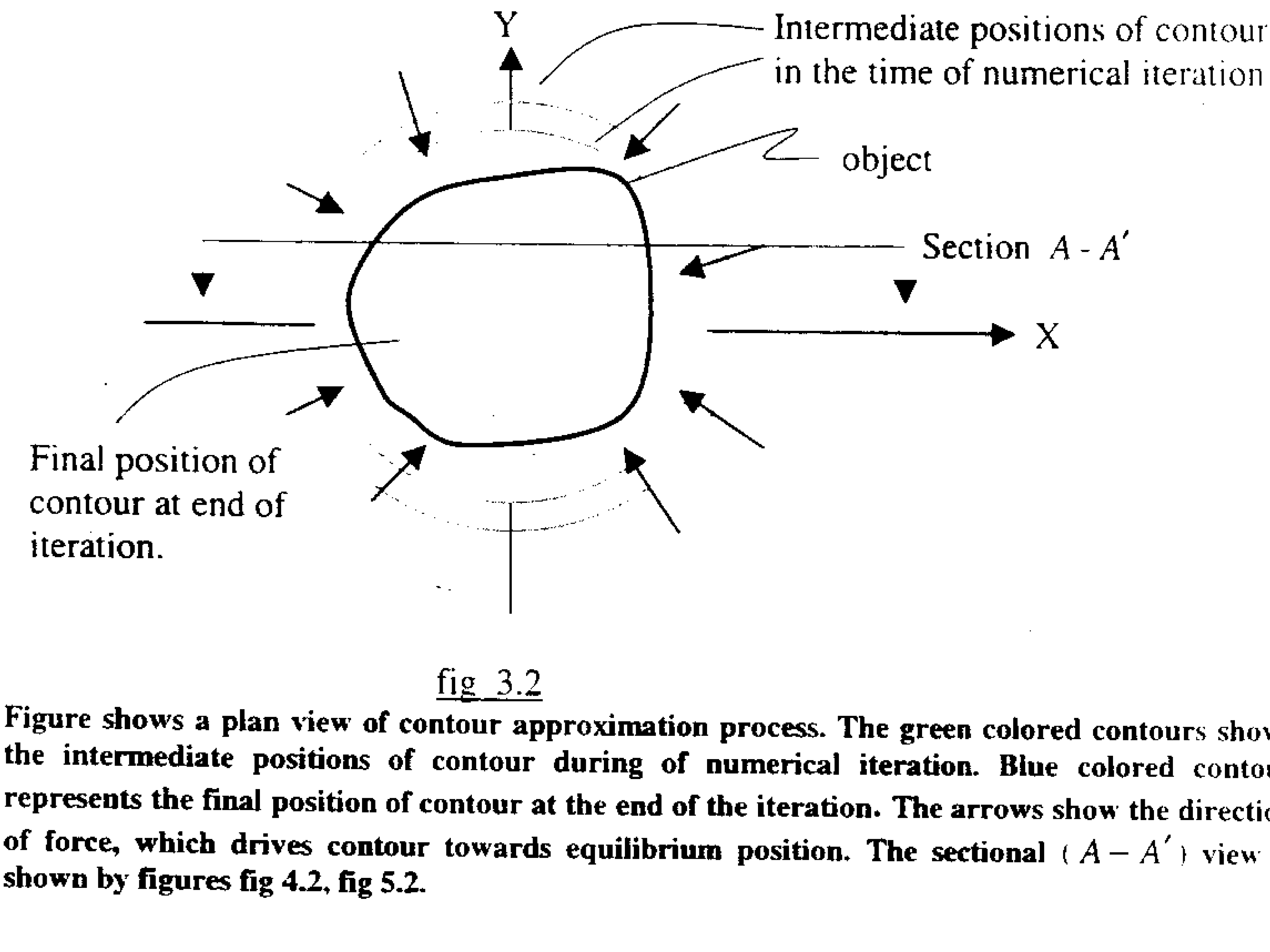


fig 3.2

Figure shows a plan view of contour approximation process. The green colored contours shows the intermediate positions of contour during of numerical iteration. Blue colored contour represents the final position of contour at the end of the iteration. The arrows show the direction of force, which drives contour towards equilibrium position. The sectional ($A - A'$) view is shown by figures fig 4.2, fig 5.2.

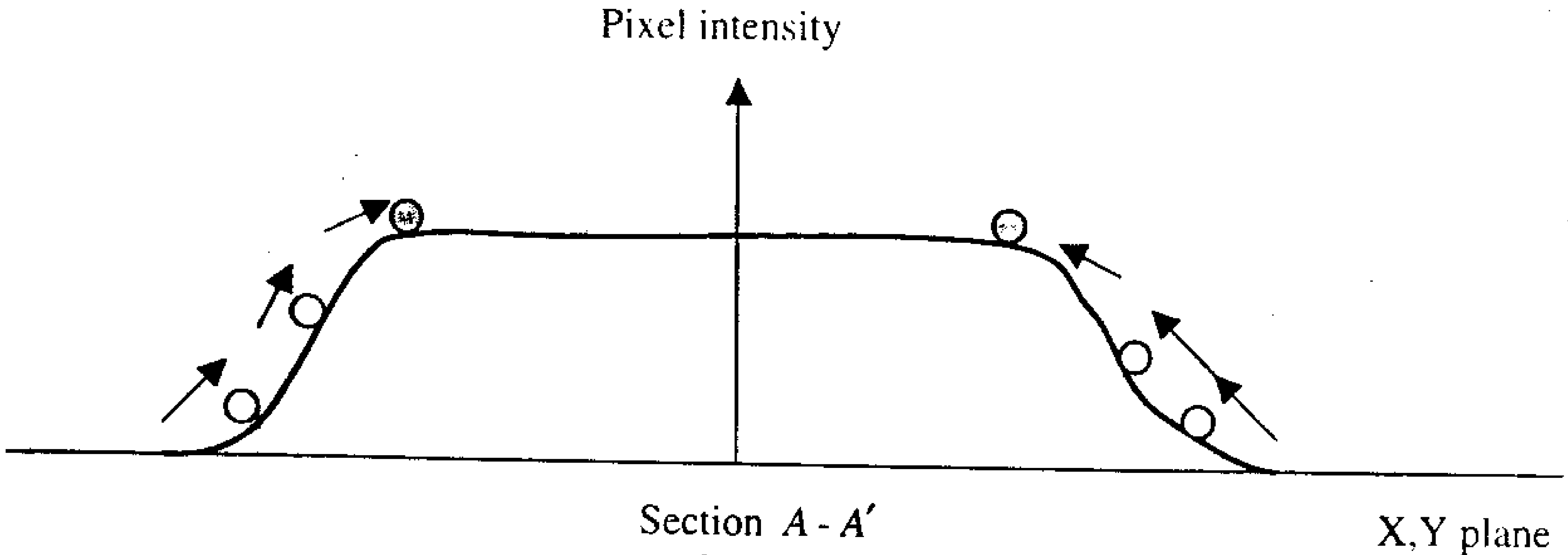


fig 4.2

Figure shows the elevation view of A- A' section of fig 3.2. The color notation of contour is same as in fig 3.2.

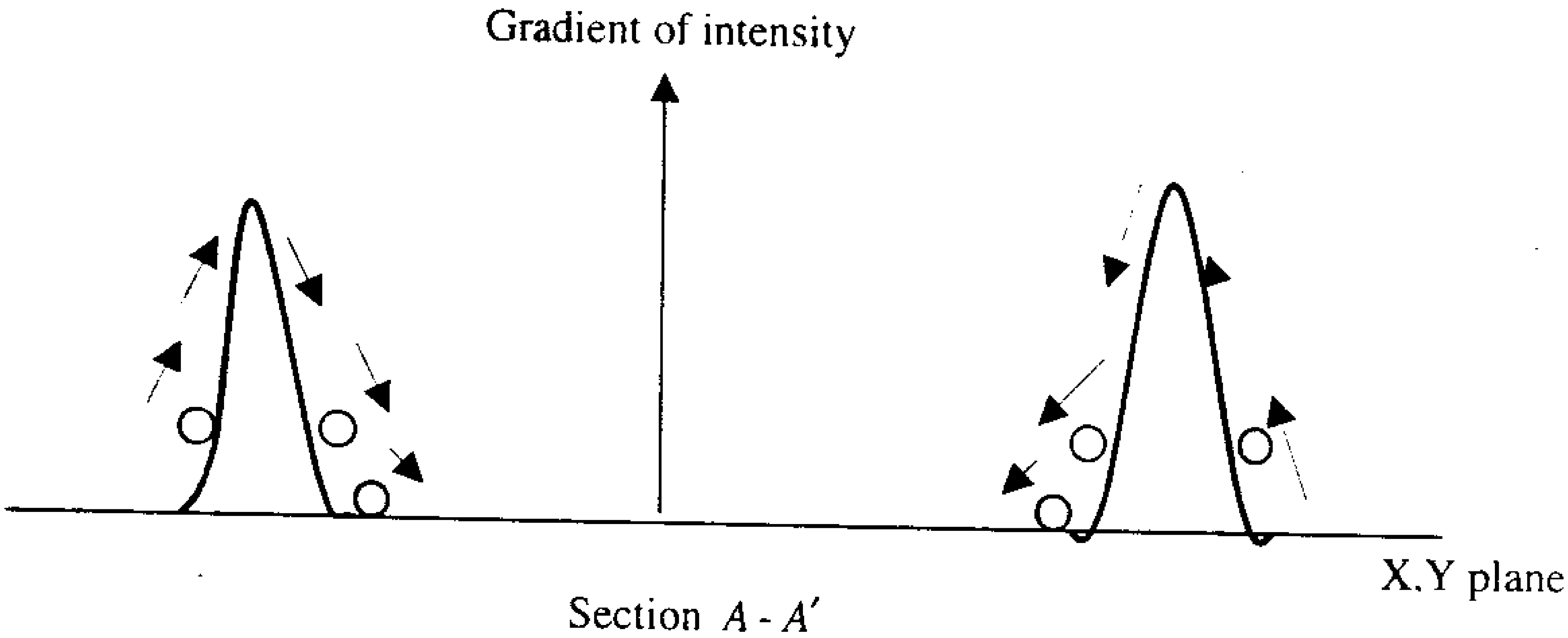


fig 5.2

Figure shows the elevation view of A - A' section of fig 3.2. Y-axis represents the gradient of intensity in image plane. The color notation of contour is same as in fig 3.2. The figure shows that contour gets a equilibrium position where gradient of intensity is zero.

Fig 3.2, 4.2 and 5.2 shows a brief overview of mechanics of snake model. It is noticeable that accuracy depends on strength of the edge. The equilibrium position occurs near the position of the contour where $\frac{\partial F}{\partial X} = 0$ and $\frac{\partial F}{\partial Y} = 0$ (where F is the intensity value of pixel at (X, Y) position).

Energy term represents the amount of work done required to change the shape of the contour from its equilibrium position. Say, for the small change of contour at (t+1)th iteration, the point $P_t(s)$ is shifted to point $P_{t+1}(s)$ in the direction u(s). In this shift amount of displacement is say l(s) and the force activated is $F_{p_t}(s)$ then the amount of work done is given by $\partial w = l(s)F_{p_t}(s).u(s)$. The process is illustrated in fig 6.2.

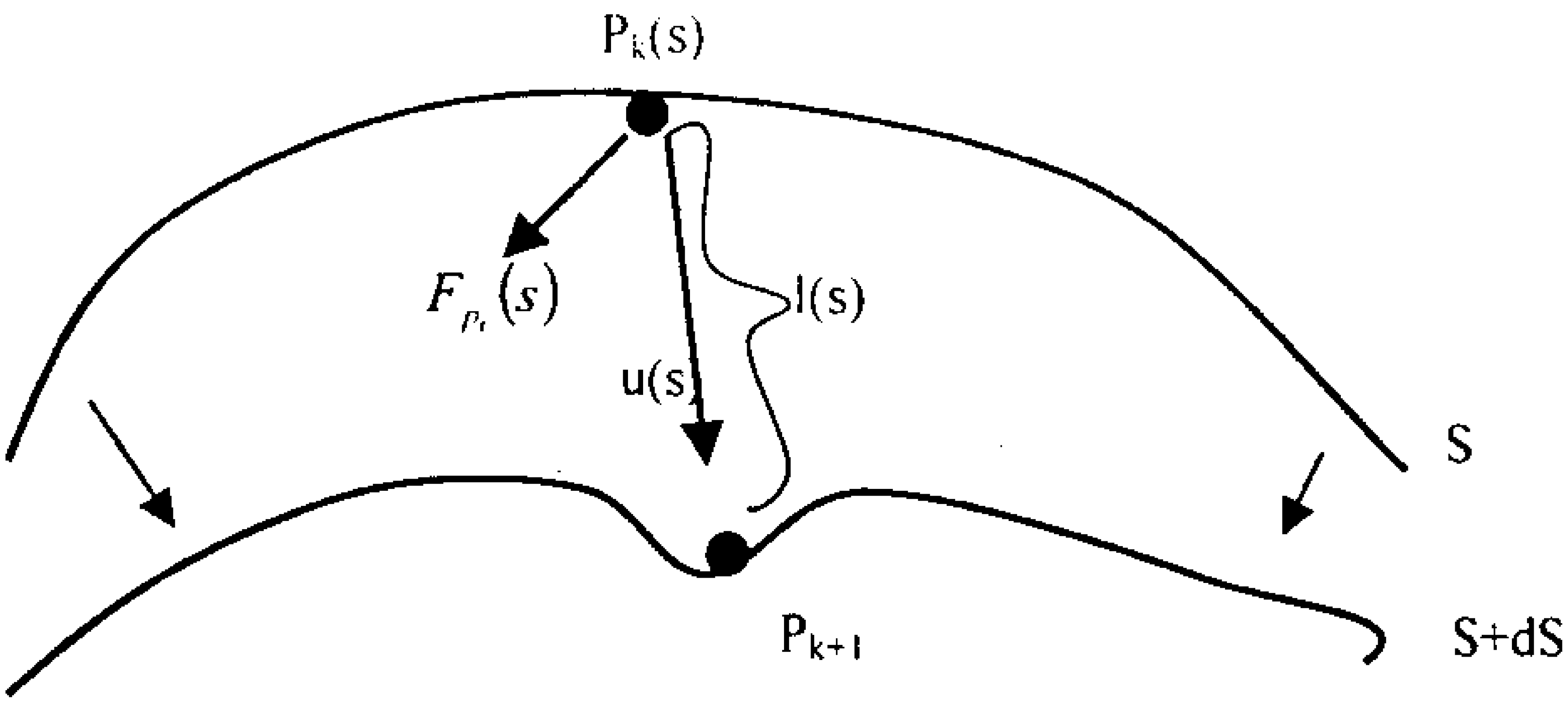


fig 6.2
Figure shows mechanics of deformation of contour

Any particular contour is a solution of the contour model if forces generated in all the points above the contour are zero.

$$F_{p_i}(s) = 0, \forall \psi \ .$$

where ψ is the set of points above the contour.

So the change of energy for all the points on contour is set to be zero.

$$\partial w = 0, \forall \psi$$

From the discussion made so far, it is clear that to drive the contour the force is the most important criteria. In snake model force is mainly the gradient force, which is directly proportional to image gradient. So absence of proper gradient force may lead the snake model to a meaningless contour approximation. The model prescribed in this paper, depends not only on image gradient but also on interior property of image surface, which yields comparatively better result than snake model. The next section of the report describes the methodology of proposed model.

3 Proposed methodology

The high stress density zone searching model uses a two dimensional surface representing an image plane enclosed by a contour. This contour is guided by an internal energy (same as active contour model) and an external energy, which is designed in a different way from the snake model. As the image is a two dimensional surface, it can be defined in 3D Euclidean space $\Re^3$ as:

$$F=f(X, Y),$$

where F is the intensity value at the point (X, Y). As image plane is assumed as a metal plate, energy at any point is due to its elevation from its datum level. The potential energy, which is induced due to elevation of intensity surface, is directly proportional to height from the datum level. Datum level is taken as zero intensity value. So potential energy induced is directly proportional to the F. Strain energy induced is directly proportional to the square of the first order derivatives of F and bending energy is proportional to the square of the second order derivatives of F. So assuming metal plate model for image surface external energy at any point on image surface can be defined as [3]:

$$\varepsilon_p(X,Y) = w_{10}F + w_{20}\left|\frac{\partial F}{\partial X}\right|^2 + w_{21}\left|\frac{\partial F}{\partial X}\right|\left|\frac{\partial F}{\partial Y}\right| + w_{22}\left|\frac{\partial F}{\partial Y}\right|^2 + w_{30}\left|\frac{\partial^2 F}{\partial X^2}\right|^2 + w_{31}\left|\frac{\partial^2 F}{\partial X \partial Y}\right|^2 + w_{32}\left|\frac{\partial^2 F}{\partial Y^2}\right|^2 \quad ..(1.3)$$

where w_{10} is gravitational parameter, w_{20}, w_{21}, w_{22} are elasticity parameters and w_{30}, w_{31}, w_{32} are rigidity parameters. Subscript p is to denotes energy at point (X,Y). Fig 1.3 shows geometrical structure of the scheme. Actual contour moves in X-Y plane at Z=0. Projection of the contour in image plane encloses the object. From the discussion in section 1 contour moves in a way that the difference between the energy density of region inside the projected contour and region outside the projected contour tends to maximized.

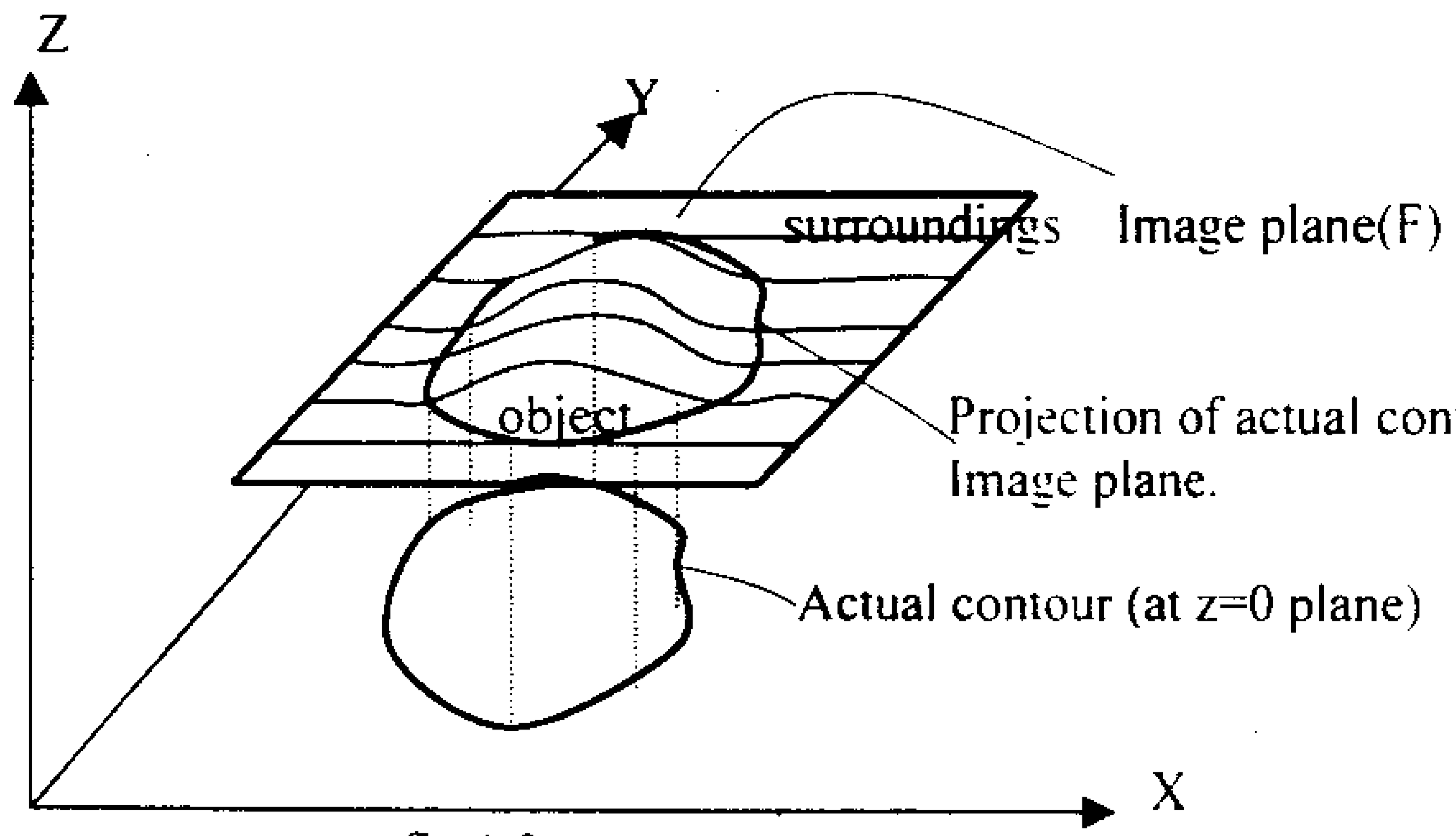


fig 1.3

Figure shows relative position of image plane and contour

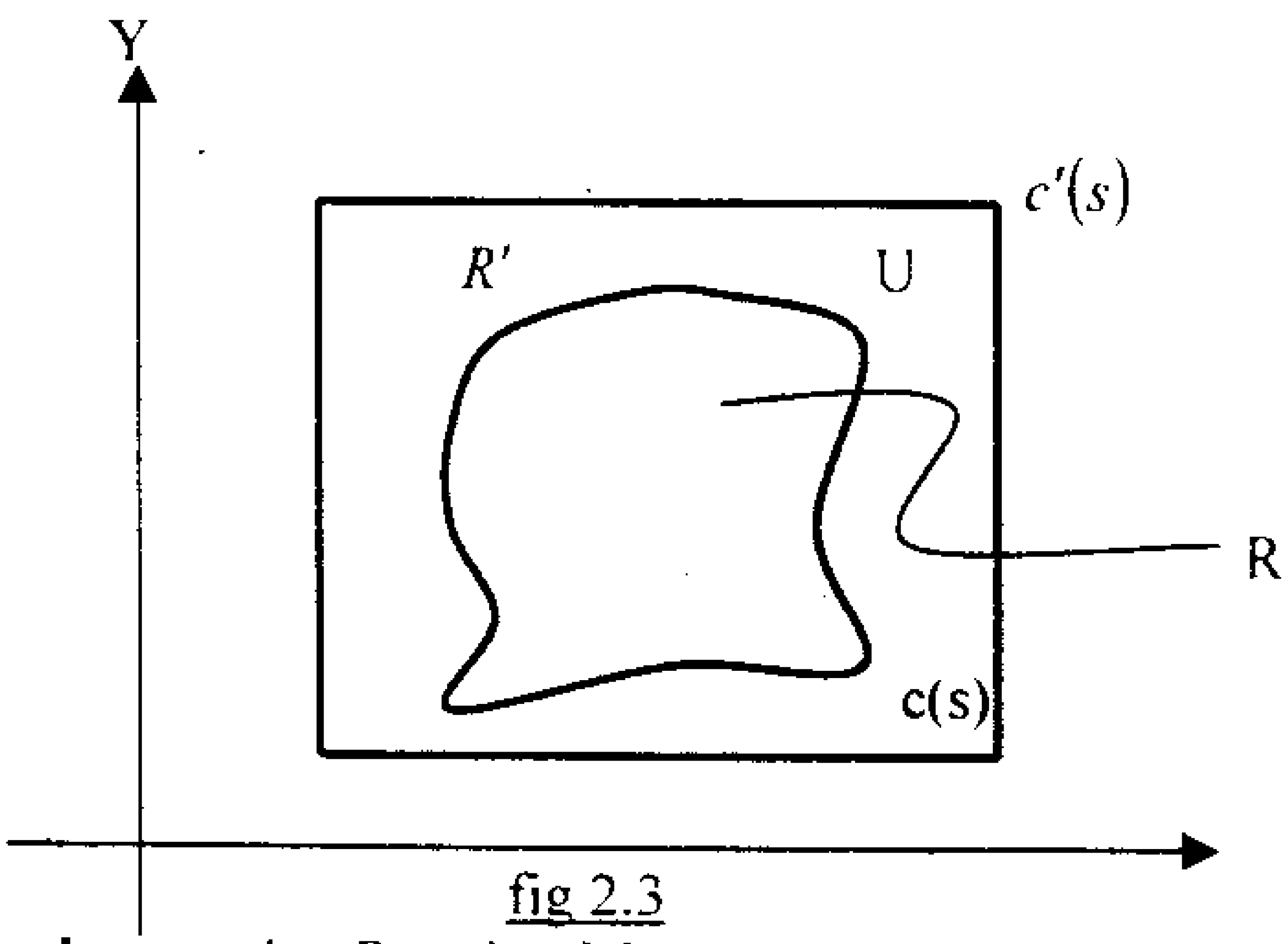


fig 2.3

Figure shows region R enclosed by contour c(s) and region R' is bounded by contour c(s) and $c'(s)$ from inside and outside respectively. U is the total region bounded by $c'(s)$

Let R denotes the region inside the contour and R' denotes the region outside the contour. Contour is denoted by c(s). The entire image plane is bounded by a rectangular shaped fixed contour $c'(s)$. So c(s) moves freely insides the region enclosed by $c'(s)$ and the region R' is bounded by c(s) from inside and bounded by $c'(s)$ from outside as shown in fig 2.3. U is the total region enclosed by contour $c'(s)$. Energy density or average energy distribution of region R bounded by contour c(s) is given by:

$$E(\varepsilon_p)_{c(s)} = \frac{\iint_{R(c(s))} \varepsilon_p(X,Y)dXdY}{\iint_{R(c(s))} dXdY} \qquad ..(2.3)$$

Expression in (2.3) gives the energy density of object region. Energy density of surrounding region can be found out by evaluating the energy distribution of region R' as follows:

$$E(\varepsilon_p)_{c'(s)-c(s)} = \frac{\iint_{R'(c'(s),c(s))} \varepsilon_p(X,Y)dXdY}{\iint_{R'(c'(s),c(s))} dXdY} = \frac{\iint_{U(c'(s))} \varepsilon_p(X,Y)dXdY - \iint_{R(c(s))} \varepsilon_p(X,Y)dXdY}{\iint_{U(c'(s))} dXdY - \iint_{R(c(s))} dXdY} \qquad ..(3.3)$$

$E(\varepsilon_p)_{c(s)}$ is a function of c(s), whereas $E(\varepsilon_p)_{c'(s)-c(s)}$ is function of both the contour c(s) and $c'(s)$. $c'(s)$ is fixed, $E(\varepsilon_p)_{c'(s)-c(s)}$ is also the function of only c(s). A new parameter is being introduced here known as distance function as defined bellow:

$$d_{RR'}(c(s)) = E(\varepsilon_p)_{c(s)} - E(\varepsilon_p)_{c'(s)-c(s)} \qquad ..(4.3)$$

The distance function gives the difference between energy density of region inside the contour and energy density of region outside the contour. Our objective is to maximize the distance function with respect to contour position.. Result of maximization gives a contour, which should identify the object correctly. Expanding the terms in the right hand sides of the expression (4.3) the distance function can be written as:

$$d_{RR'}(c(s)) = \frac{\iint_{R(c(s))} \varepsilon_p(X,Y)dXdY}{\iint_{R(c(s))} dXdY} - \frac{\iint_{U(c'(s))} \varepsilon_p(X,Y)dXdY - \iint_{R(c(s))} \varepsilon_p(X,Y)dXdY}{\iint_{U(c'(s))} dXdY - \iint_{R(c(s))} dXdY} \qquad ..(5.3)$$

Above expression is a functional of variable contour. To find out maxima of distance function it is preferable to take the help of variational calculus [7]. Variational calculus cannot solve the problem for nonlocal function [7] where numerator and denominator both contain the variable with respect to which extrema is to be found out. Expression (5.3) is a nonlocal type of function as variable c(s) is present both in the numerator and denominator. It requires redefinition in such a way that it should fulfill two basic criteria. First of all, the redefined function should be a nonlocal function and secondly it holds the extrema value for the point where distance function also holds the same.

Maximization of any composite function, which is a ratio of two positive functions (function which always gives positive value) means that maximization of function in numerator and minimization of function in denominator. For an example, if f_1, a composite function of f_2 and f_3, is of the form $f_1 = \frac{f_2}{f_3}$, then maximization of f_1 means maximization of f_2 and minimization of f_3. So maximization of f_1 is equivalent to maximization of function $\phi_1 = f_2 - f_3$. By the equivalent term it is denoted that for the values of variables for which f_1 is maximum, should also make ϕ_1 maximum.

From the above discussion it can be concluded that if expression (5.3) is to be maximized, then it is equivalent to the maximization of expression given below:

$$d_{RP}(c(s)) = \iint_{R(c(s))} \varepsilon_p(X,Y)dXdY - \iint_{R(c(s))} dXdY$$
$$-\left[\left(\iint_{R(c'(s))} \varepsilon_p(X,Y)dXdY - \iint_{R(c(s))} \varepsilon_p(X,Y)dXdY\right) - \left(\iint_{R(c'(s))} dXdY - \iint_{R(c(s))} dXdY\right)\right]$$

Further simplification of the above expression leads to:

$$d_{RP}(c(s)) = 2\left[\iint_{R(c(s))} \varepsilon_p(X,Y)dXdY - \iint_{R(c(s))} dXdY\right]$$
$$-\left[\iint_{R(c'(s))} \varepsilon_p(X,Y)dXdY - \iint_{R(c'(s))} dXdY\right] \quad ..(6.3)$$

Contour c'(s) is a fixed contour. When change of $d_{RP}(c(s))$ with respect to c(s) is found out for extrema condition, then contribution of the terms that depend on c'(s) is zero. So $d_{RP}(c(s))$ is further reduced to the equivalent expression:

$$d_{RP}(c(s)) = \left[\iint_{R(c(s))} \varepsilon_p(X,Y)dXdY - \iint_{R(c(s))} dXdY\right] \quad ..(7.3)$$

Above term is the amount of external energy equivalent to term $\int_0^1 F_{ext} ds$ in (2.2). So the complete energy expression of high stress density region searching snake model, which is a combination of internal energy term (strain energy and bending energy of contour) and external energy term (distance function) is as follows:

$$\varepsilon = \int_0^1 \left\{ \frac{1}{2}\alpha\left[\left(\frac{\partial X}{\partial s}\right)^2 + \left(\frac{\partial Y}{\partial s}\right)^2\right] + \frac{1}{2}\beta\left[\left(\frac{\partial^2 X}{\partial s^2}\right)^2 + \left(\frac{\partial^2 Y}{\partial s^2}\right)^2\right]\right\} ds$$
$$-\left[\lambda \iint_{R(c(s))} \varepsilon_p(X,Y)dXdY - \xi \iint_{R(c(s))} dXdY\right] \quad ..(8.3)$$

The two terms λ and ξ is taken for making dimensional equality with the other energy terms of equation (8.3). Values of λ and ξ are taken as one. Minimization of above energy term with the help of variational calculus gives two extrema in X and Y direction (Detail derivations are given in the appendix):

$$\left(-\mathbf{E}_p+\mathbf{I}\right)\left(\vec{Y}'-\vec{Y}\right)+\mathbf{A}\vec{X}=0 \qquad ..(9.3)$$

$$\left(\mathbf{E}_p-\mathbf{I}\right)\left(\vec{X}'-\vec{X}\right)+\mathbf{A}\vec{Y}=0 \qquad ...(10.3)$$

$\mathbf{E}_p$ is the energy matrix. If there are n points on contour then diagonal element of energy matrix is vector $\vec{\varepsilon}_p=\left[\varepsilon_p(X_1,Y_1),\varepsilon_p(X_2,Y_2),...,\varepsilon_p(X_n,Y_n)\right]^T$. Terms $\vec{X},\vec{Y}$ and A carry the same meaning as in equation (2.2) and (3.2). $\vec{Y}'$ represents vector $\left[Y_2,Y_3,....,Y_n,Y_1\right]^T$, similarly $\vec{X}'$ represents vector $\left[X_2,X_3,...,X_n,X_1\right]^T$ (For detail explanation refer appendix). Unlike snake model the left side of above two equation gives the amount of drag force which drives the contour towards minima. For the above two equations the minima finding iterative equation given by gradient decent methods are:

$$\vec{X}^{t+1}=\left(\mathbf{I}+\mathbf{A}\right)^{-1}\left(\left(-\mathbf{E}_p+\mathbf{I}\right)\left(\vec{Y}''-\vec{Y}'\right)+\vec{X}'\right)\Delta t \qquad ..(11.3)$$

$$\vec{Y}^{t+1}=\left(\mathbf{I}+\mathbf{A}\right)^{-1}\left(\left(\mathbf{E}_p-\mathbf{I}\right)\left(\vec{X}''-\vec{X}'\right)+\vec{Y}'\right)\Delta t \qquad ..(12.3)$$

In next section some real domain application results of the proposed algorithm is presented.

4 Results

The high-stress seeking algorithm is applied for testing different types of images for object segmentation and tracking problem. The weighting coefficients ($\alpha,\beta,w_{10},w_{20},w_{21},w_{30},w_{31},w_{32}$) are set empirically. The values of the weighting coefficients are influenced by the imaging condition for example image intensity and noise characteristics. For the image sequence shown in results, the proposed algorithm is tested on a 797 MHz microcomputer. For both object segmentation and tracking, an initial contour is manually placed closed to the actual object. For all the examples shown the initial contour consists of 62 sample points. The iterative energy minimization technique derived using equations (11.3) and (12.3) deforms the initial contour and ultimately encircles the actual object.

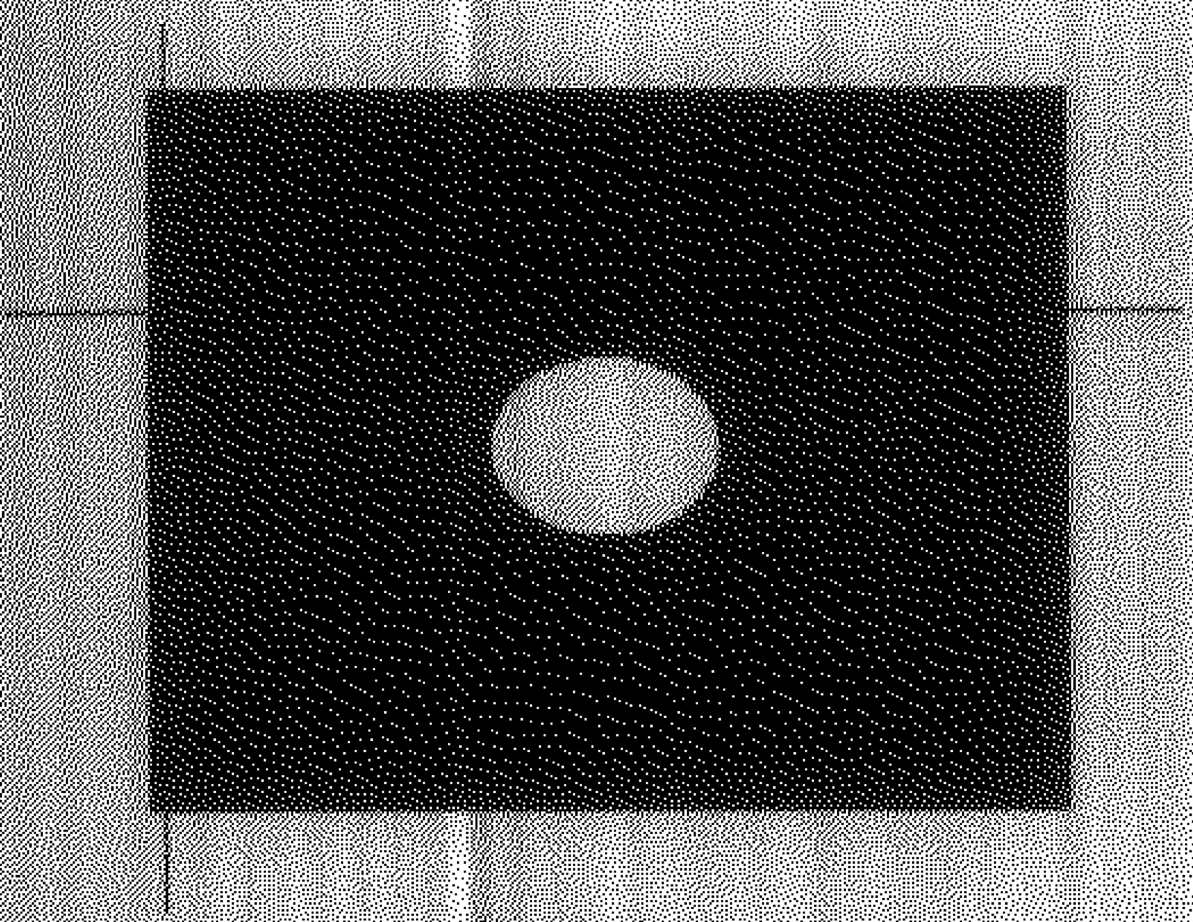


fig 1.4

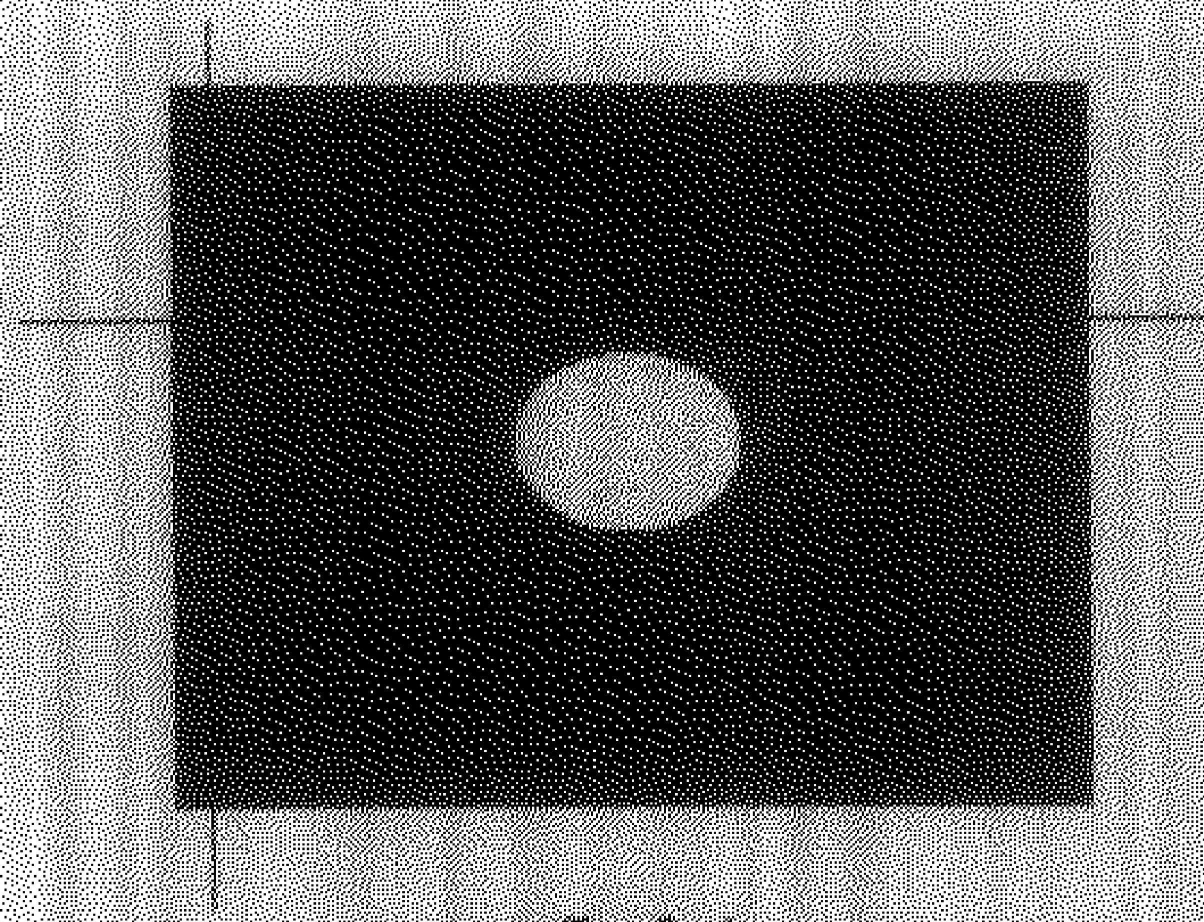


fig 2.4

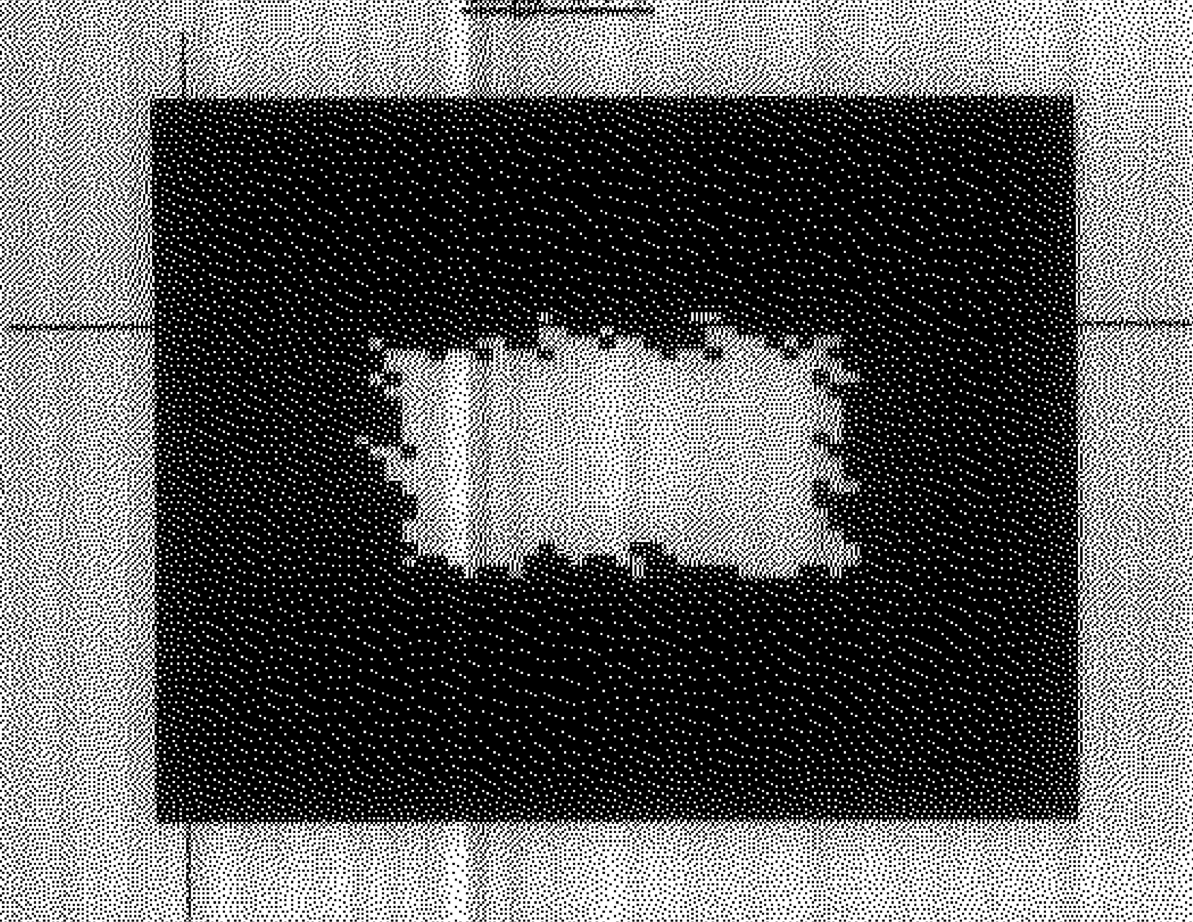


fig 3.4

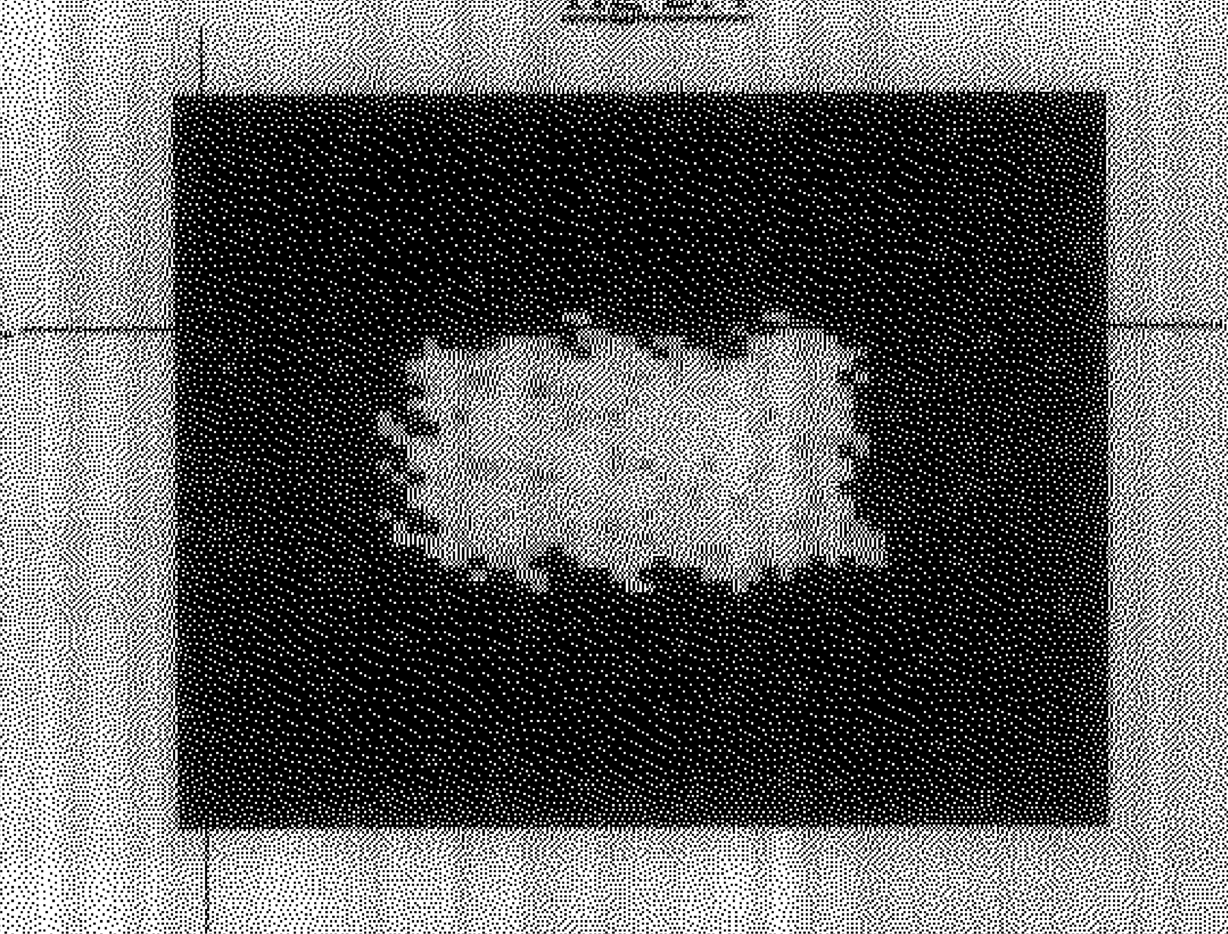


fig 4.4

Figures show the synthetic objects, which are tracked by the proposed algorithm

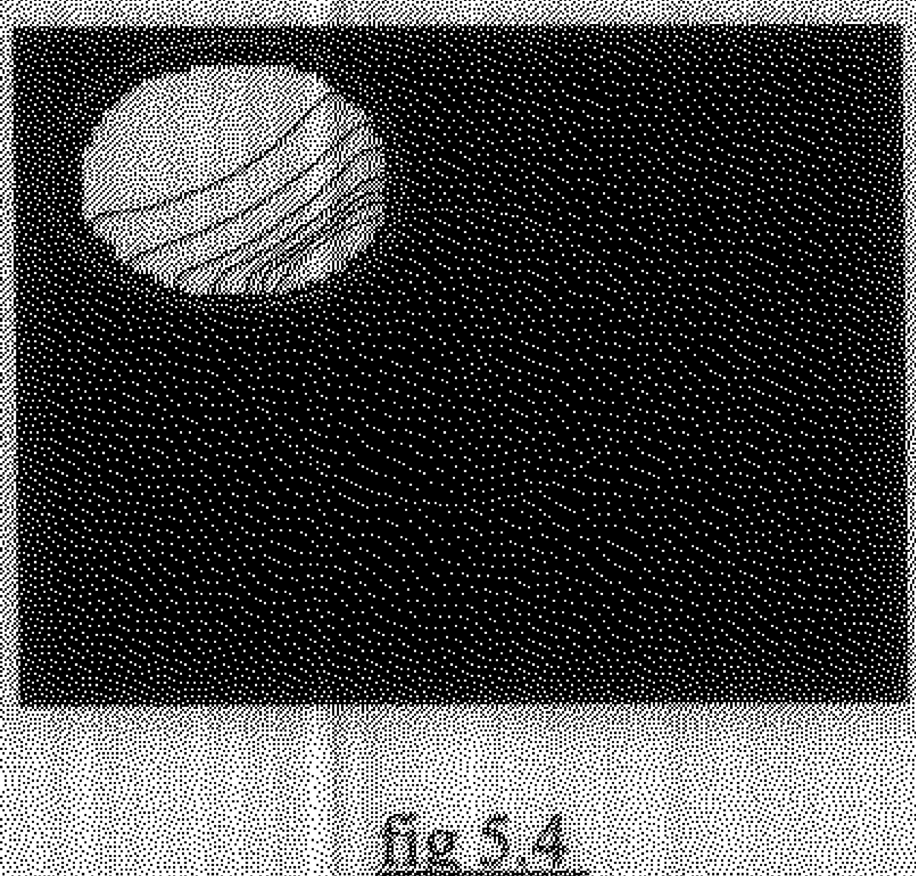


fig 5.4

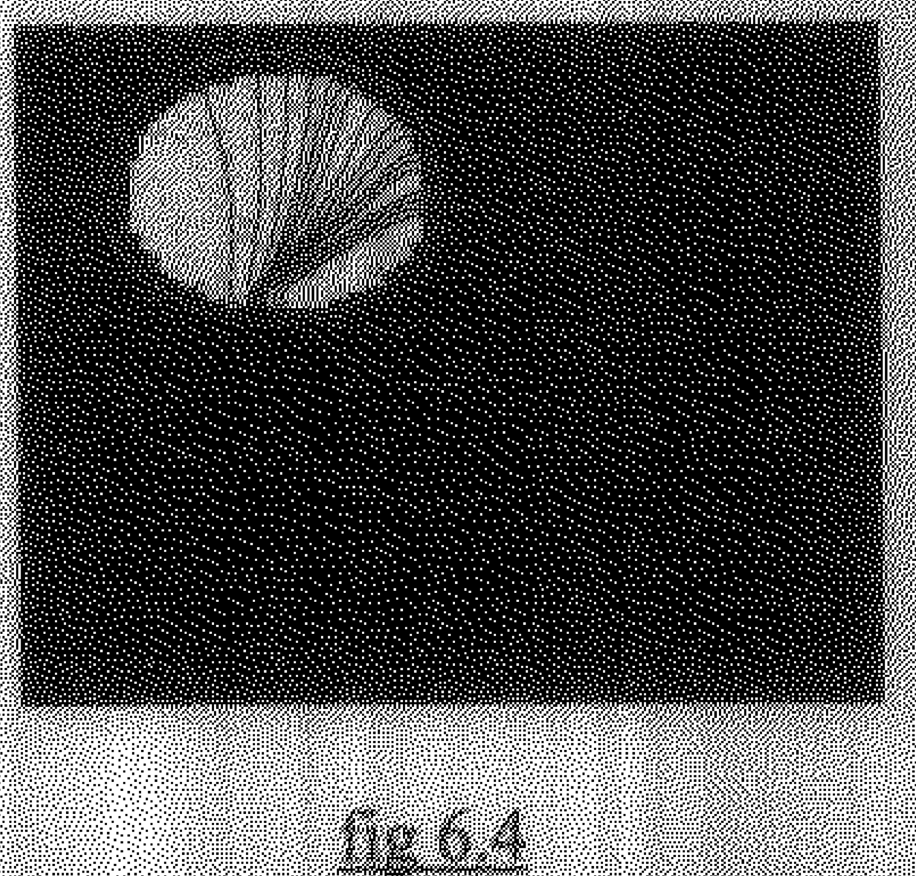
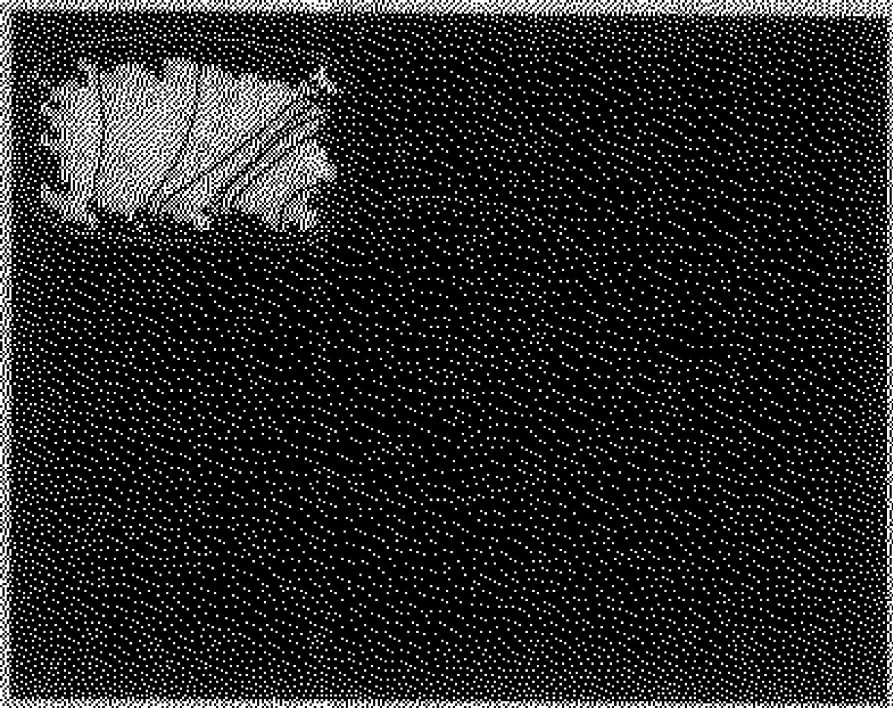


fig 6.4

Steps in segmentation problem for recognition of circular object. Fig 5.4 shows recognition in image without noise and fig 6.4 shows recognition in noisy image.

For the synthetic circular and rectangular objects shown in figs. 1.4, 2.4, 3.4, 4.4 the algorithm is used for segmentation purpose. The values of the weighting coefficient are taken as α =5, β=2, w_{10}=0.0001, w_{20}=0.005, w_{21}=0.005, w_{22}=0.005, w_{30}=0.01, w_{31}=0.01, w_{32}=0.01. Elapsed time required for all the images without noise is 7.7960sec. For noisy images it becomes 7.7970sec. Salt and pepper type noise is taken for experiment. The cyan colored contour shows the initial contour. The red colored contour shows some of intermediate steps of the initial contour, which is gradually deforming searching for high stress region. The green colored contour gives the final segmentation result. In fig 7.4 and 8.4 segmentation of rectangular object is shown. Note that identical weights as in fig 1.4 and 2.4 are used for synthetic rectangular images (fig 3.4 and fig 4.4). The rectangular edges are deliberately distorted to demonstrate that even for an irregular contour the proposed algorithm work successfully.



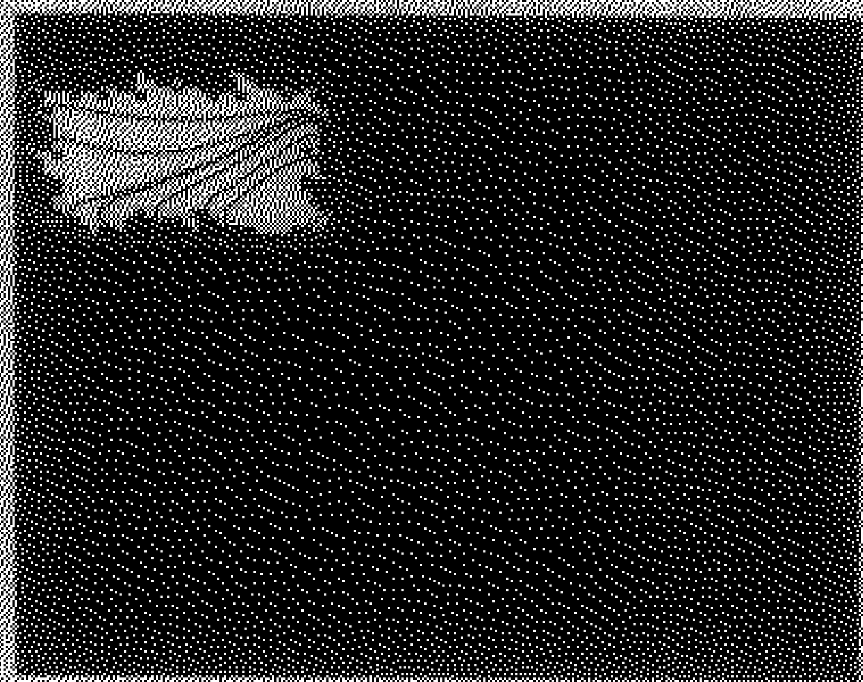
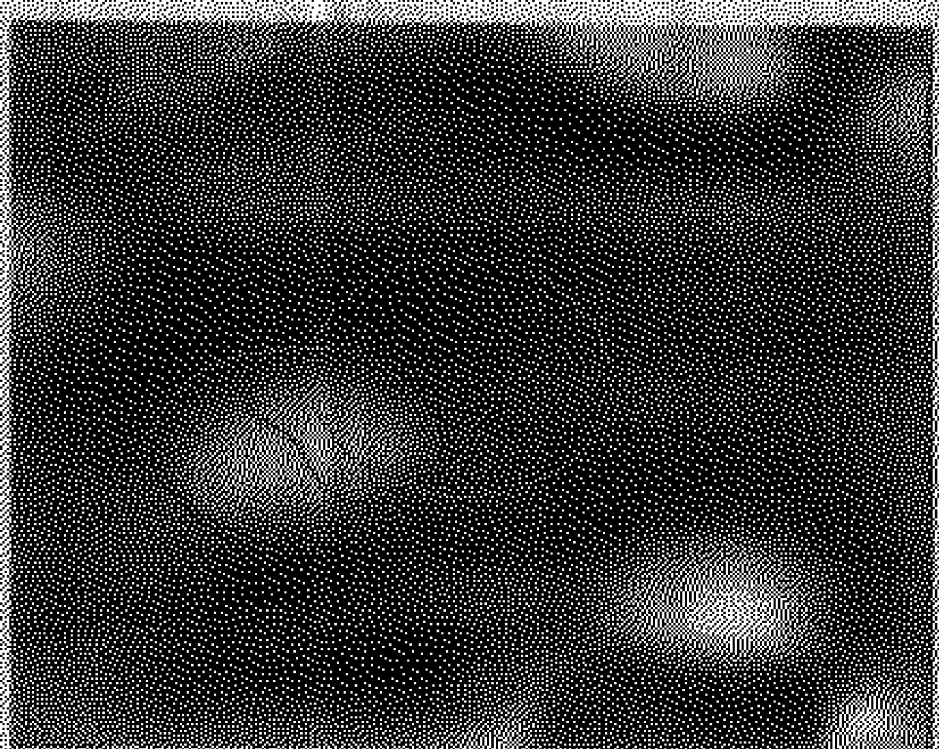


fig 7.4 fig 8.4

Steps in segmentation problem for recognition of rectangular object, fig 7.4 shows recognition in noisy image and fig 8.4 shows recognition in image with out noise.

Fig 9.4 and 10.4 show the identification of white blood cell in biological images. The values of weighting coefficient terms are taken $\alpha=5$, $\beta=2$, $w_{10}=0.001$, $w_{20}=0.01$, $w_{21}=0.01$, $w_{22}=0.01$, $w_{30}=0.05$, $w_{31}=0.05$, $w_{32}=0.05$. Color notations of contour are same as that in previous figures.



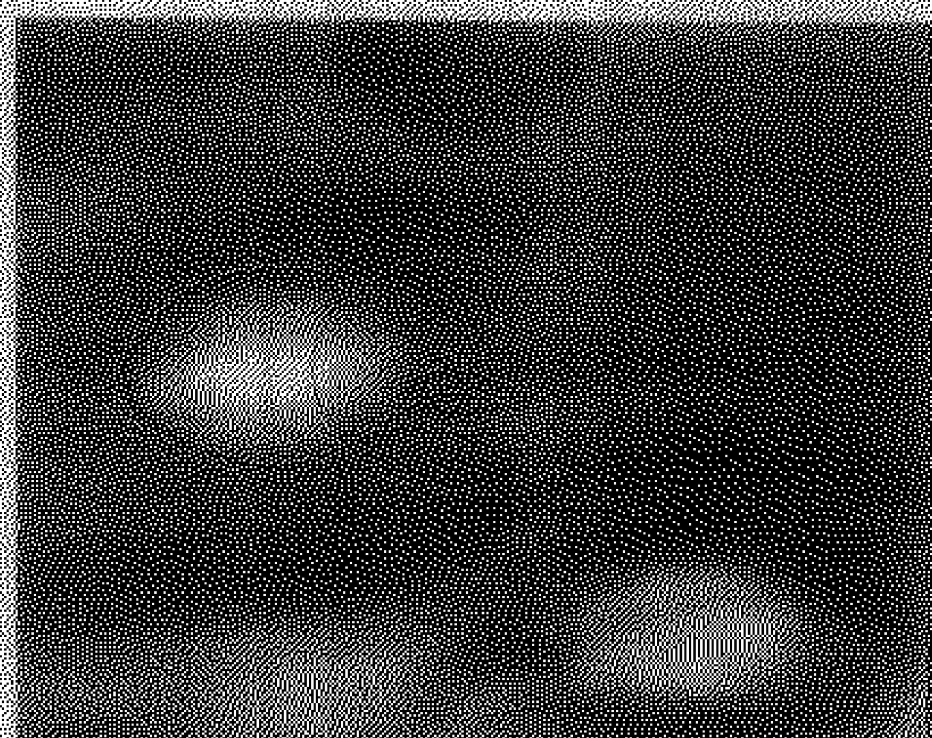


fig 9.4 fig 10.4

Steps of identifications of white blood cell in biological image. Both figures show the procedure for two different white blood cells.

In case of tracking, algorithm is first tested for a table tennis ball. The tracking is done for 30 continuous frames. The performance of the tracker is good. In case of table tennis ball tracking the values of the weighting coefficient are taken as $\alpha=26$, $\beta=2$, $w_{10}=0.01$, $w_{20}=0.002$, $w_{21}=0.002$, $w_{22}=0.002$, $w_{30}=0.008$, $w_{31}=0.008$, $w_{32}=0.008$. Fig 11.4 to fig 14.4 gives some intermediate views of the tracking scenario. In each frame 150 iterations are required to encircle the object. Elapsed time required for all the images is 4.2650sec. Color notations of contour are same as that of previous figures except the first frame where the cyan contour represents the initial contour. In all other cases the converged contour of the immediate last frame is used as the initial contour for the current frame.

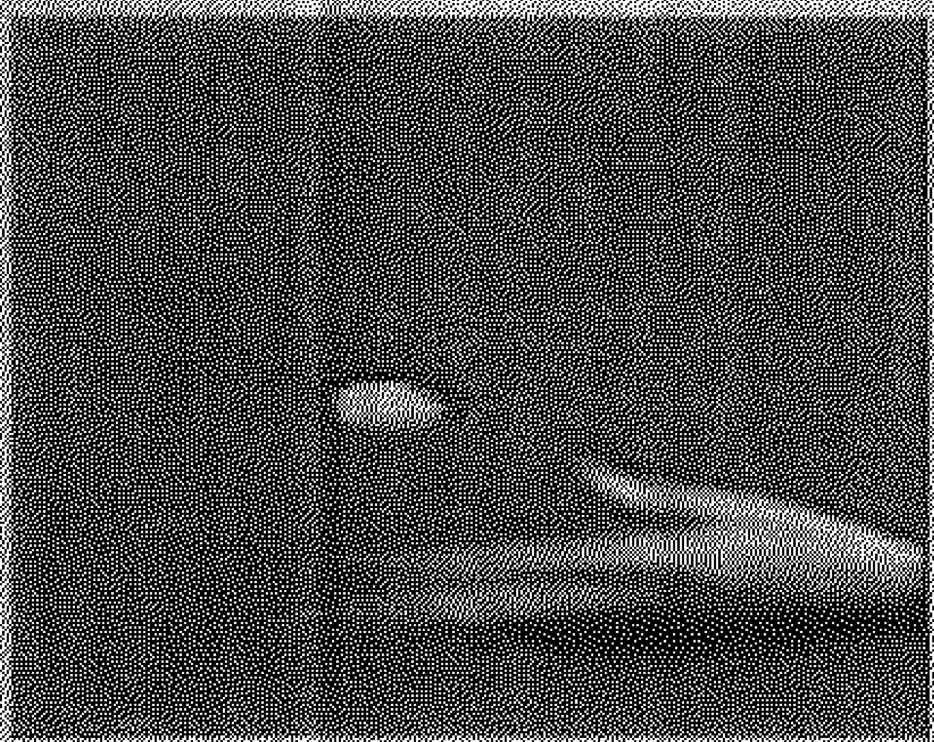


fig 11.4

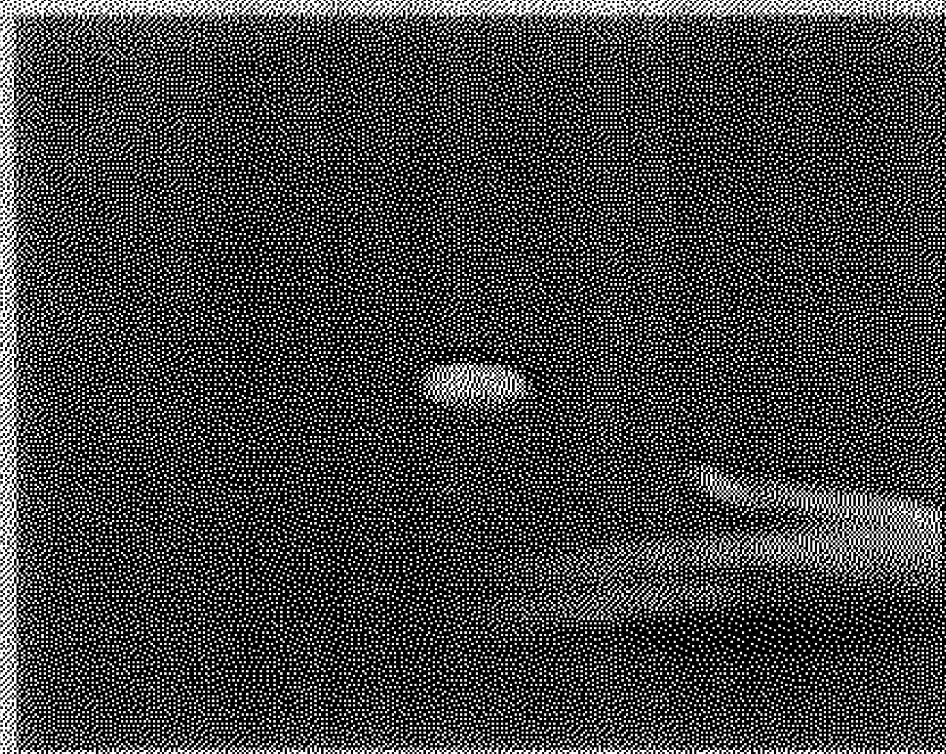


fig 12.4

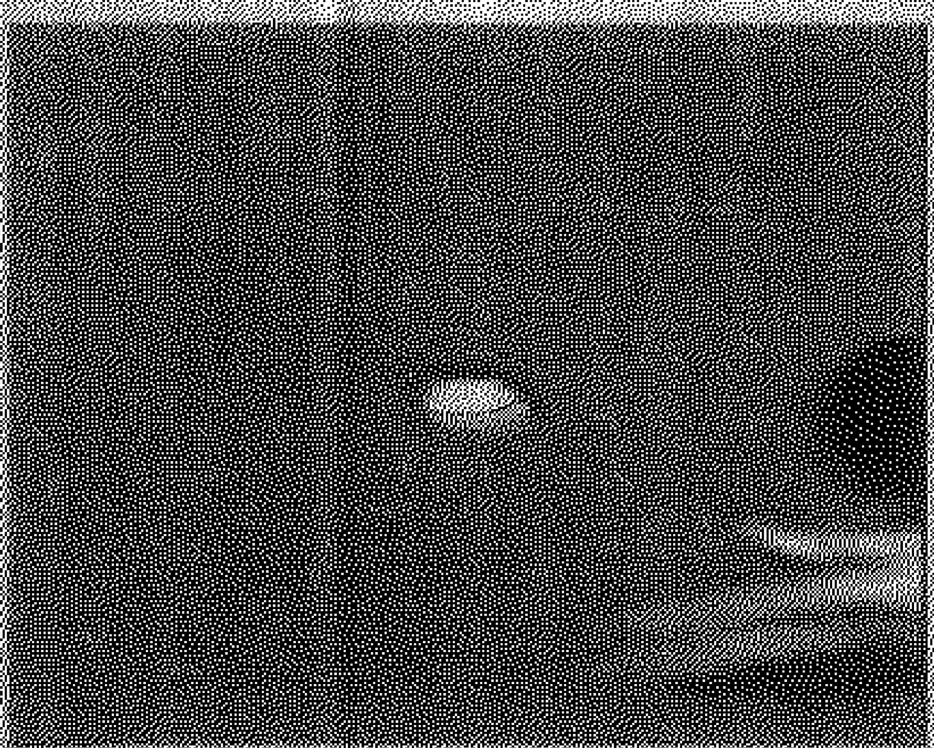


fig 13.4

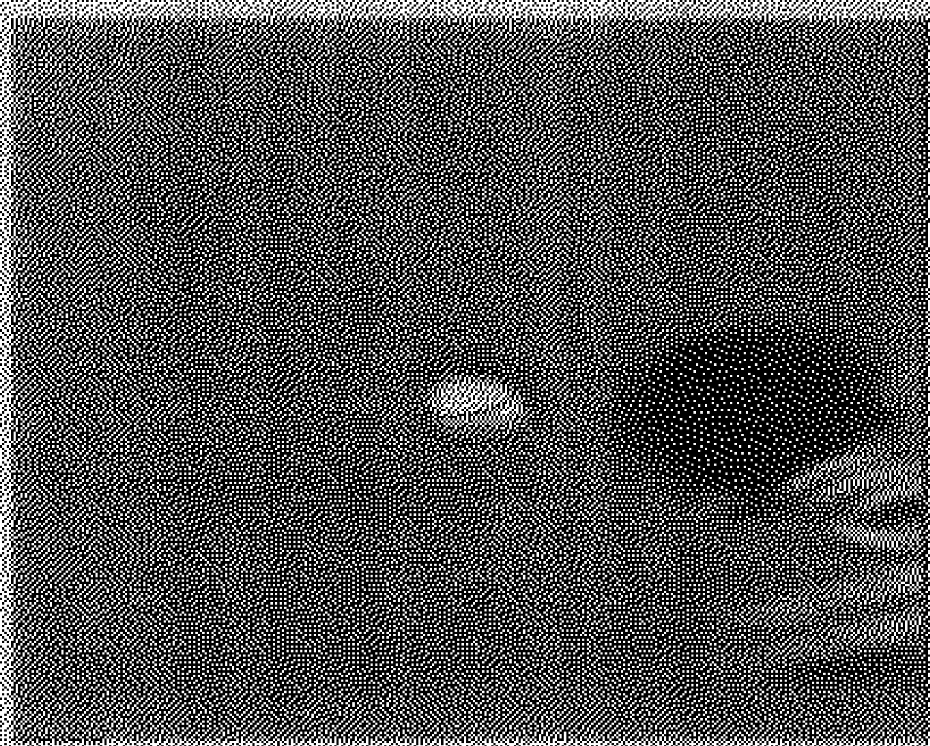


fig 14.4

Figures show steps of table tennis ball tracking. Fig 11.4 (frame no 1), fig 12.4 (frame no 5), fig 13.4 (frame no15), fig 14.4 (frame no 27).

Another example of tracking is shown in fig 15.4 to 20.4. A white blood cell is tracked in the movie. 25 frames are taken for the test. The weighting coefficients are taken as $\alpha=5, \beta=2, w_{10}=0.001, w_{20}=0.01, w_{21}=0.01, w_{22}=0.01, w_{30}=0.05, \quad w_{31}=0.05, w_{32}=0.05$. For all the frames the shape of the blood cell is captured correctly. In each frame 150 iterations are taken to capture the object.

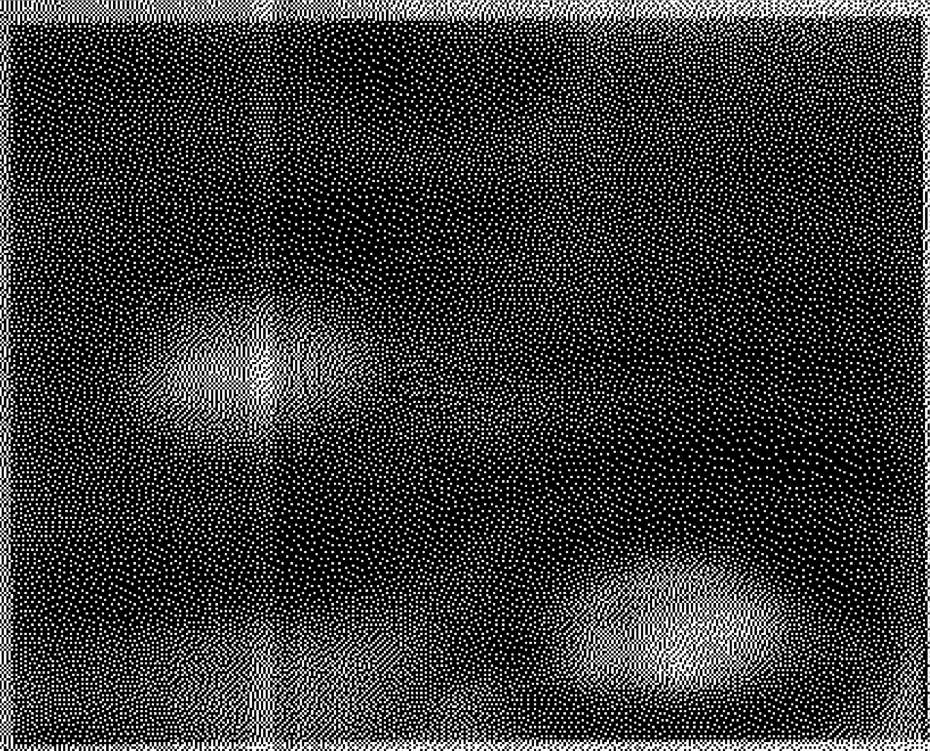


fig 15.4

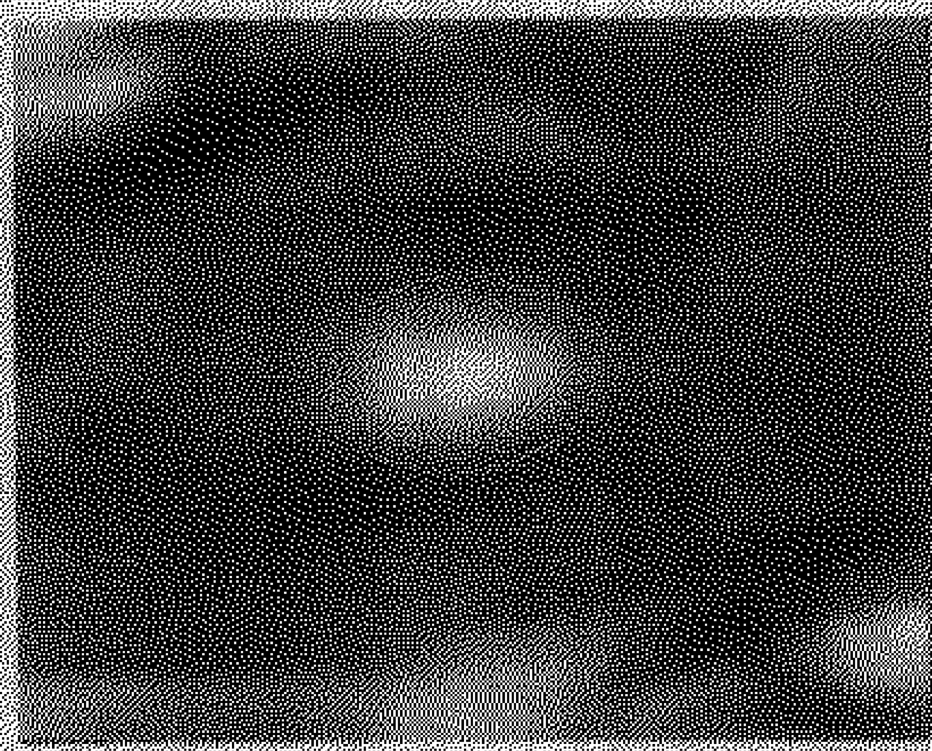


fig 16.4

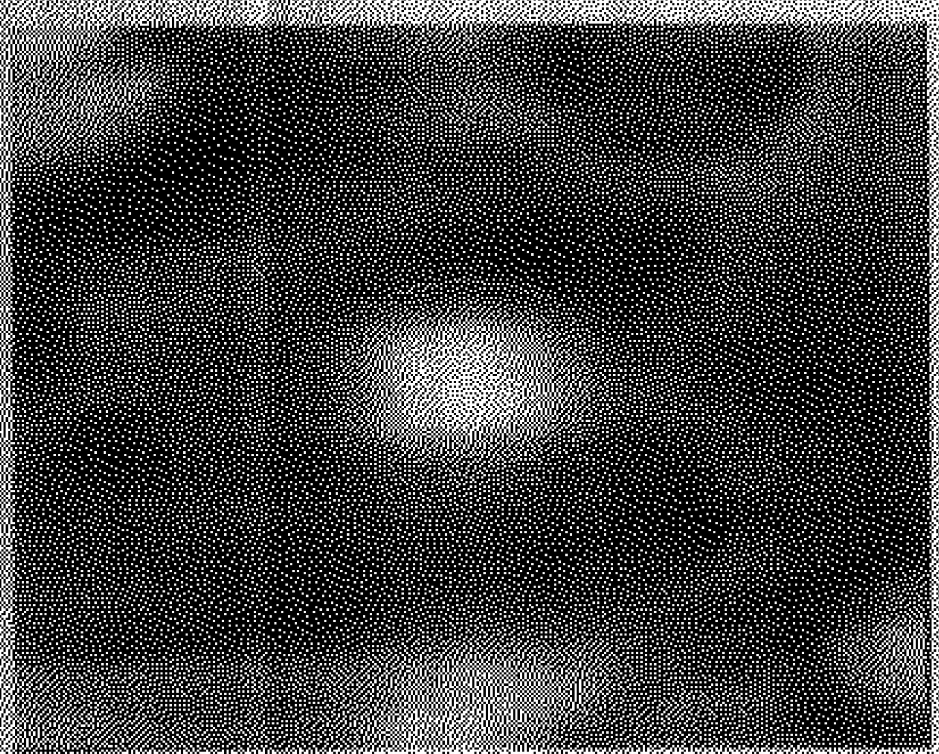


fig 17.4

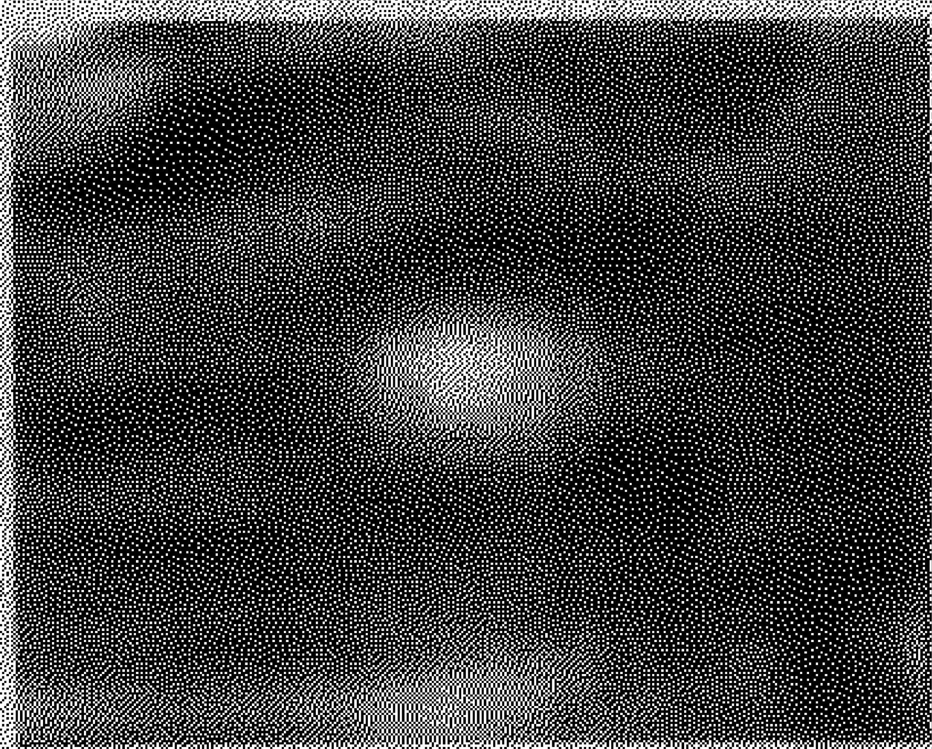


fig 18.4

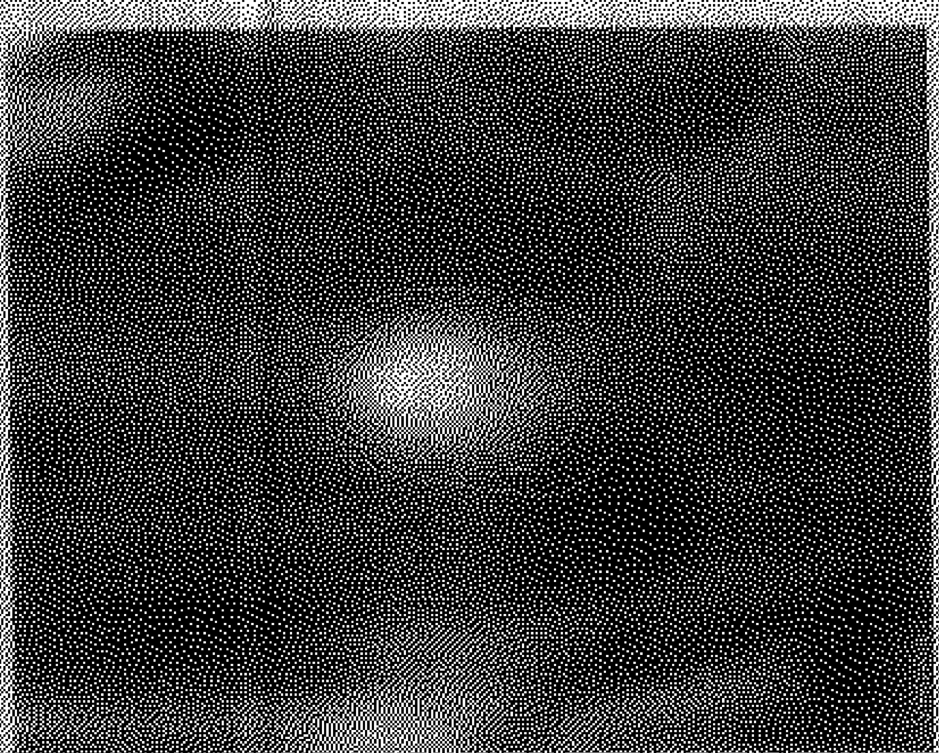


fig 19.4

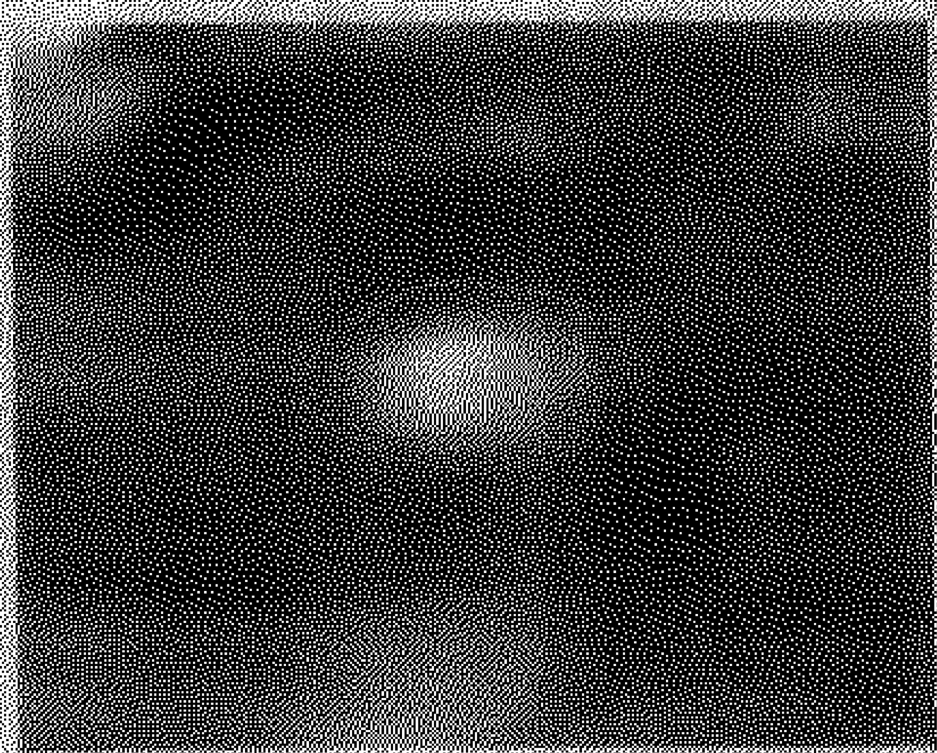


fig 20.4

Figures show steps of white blood cell tracking. Fig 15.4 (frame no 1), fig 16.4 (frame no5), fig 17.4 (frame no 10), fig 18.4 (frame no 15), fig 19.4 (frame no 20), fig 20.4 (frame no 25)

5 Conclusion

The model is developed depending upon the idea of region-based technique. The influence of derivative operator can be extended by adding influence of neighborhood intensities to each pixel intensity in the image plane. In this model Gaussian smoothing partially does the job. But it will be better if any image plane is approximated by any standard surface like B-spline surface and the technique is applied on that surface.

Our proposed algorithm encircles the target object accurately but it cannot detect the object boundary perfectly. Future work should improve this discrepancy.

6 References

[1] M. Kass, A. Witkin and D.Terzopoulos, "*Snake: Active contour model*", Int. J. Computer Vision, Vol. 1, pp. 321-331, 1987.

[2] C. Xu and J.L. Prince,"*Generalized Gradient Vector Flow External Forces for Active Contours*", Signal Processing, An International Journal, 71(2), pp. 131-139, December 1998.

[3] Linguan, Sun-Yang and San Lersen, "*Deformable surface spine model*", Multimedia Video and Image Processing , CRC press London, New York Page 223-227.

[4] F.M. Ansia, M.G. Penedo, C. Marino, J. Lopez and A. Mosquera, "*Morphological Analysis with Active Nets*", ICPR 2000:1486-1489.

[5] Remi Ronfard, "*Region-Based Strategies for Active contour model*", Int. J. Computer Vision, 13:2, 229-251(1994).

[6] D.P. Mukherjee, N. Ray and S.T. Acton, "*Tracking of Rolling Leukocytes through Frame Segmentation*", communicated to IEEE Trans. Image Processing, 2003.

[7] I. M. Gelfand, S.V. Fomin,"*Calculus of variations*", 1st ed., Prentice-hall, New Jersey,1963.

[8] B.S.Grewal "*Higher engineering mathametics*", 32th ed, Khanna Publishers,1995.

[9] "*Numerical recipes in C*", 2nd ed, Cambridge university Press, March 1993.

7 Appendix

Calculus of variation deals with functional. The term functional can be explained by an example given below[8].

Let problem is to find out a curve through two points (X_1,Y_1) and (X_2,Y_2) whose length is minimum. It is same as determining the curve Y=Y(X) for which $Y(X_1)=Y_1$, $Y(X_2)=Y_2$ such that $\int_{X_1}^{X_2}\sqrt{(1+Y'^2)}dX$ is a minimum.

In general terms, we wise to find the curve Y=Y(X) where $Y(X_1)=Y_1$ and $Y(X_2)=Y_2$ such that for a given function $f(X,Y,Y')$,

$$\int_{X_1}^{X_2} f(X,Y,Y')dx \qquad ..(1.7)$$

is a stationery value or an extremum.

An integral such as (1.7), which assumes a definite value for functions of the type Y=Y(X) is called a functional.

In differential calculus, we deal with the problems of maxima and minima of functions. The calculus of variations is however, concerned with maximizing or minimizing functional.

The equation (8.3) is a functional as it is a function of variable contour defined in (1.2). In (7.3) distance function is defined as,

$$d_{RR'}(c(s))=\left[\iint_{c(s)}\varepsilon_p(X,Y)dXdY-\iint_{c(s)}dXdY\right].$$

Changing the surface integral to line integral by applying Green's [6] theorem we get,

$$d_{RR'}(c(s))=\oint(PdX+QdY)-\frac{1}{2}\oint(-YdX+XdY)$$

where $\quad P=-\frac{1}{2}\int_0^Y\varepsilon_p(X,t)dt$ and $Q=\frac{1}{2}\int_0^X\varepsilon_p(t,Y)dt$

Appling the change equation (8.3) becomes.

$$\varepsilon=\int_0^1\left\{\frac{1}{2}\alpha\left[\left(\frac{\partial X}{\partial s}\right)^2+\left(\frac{\partial Y}{\partial s}\right)^2\right]+\frac{1}{2}\beta\left[\left(\frac{\partial^2 X}{\partial s^2}\right)^2+\left(\frac{\partial^2 Y}{\partial s^2}\right)^2\right]\right\}ds$$
$$-\left[\oint(PdX+QdY)-\frac{1}{2}\oint(-YdX+XdY)\right]$$

Convert the line integral part into parametric form the functional can be expressed as:

$$\varepsilon=\int_0^1\left\{\frac{1}{2}\alpha\left[\left(\frac{\partial X}{\partial s}\right)^2+\left(\frac{\partial Y}{\partial s}\right)^2\right]+\frac{1}{2}\beta\left[\left(\frac{\partial^2 X}{\partial s^2}\right)^2+\left(\frac{\partial^2 Y}{\partial s^2}\right)^2\right]\right\}ds$$
$$-\left[\int_0^1\left(P\frac{dX}{ds}+Q\frac{dY}{ds}\right)ds-\frac{1}{2}\int_0^1\left(-Y\frac{dX}{ds}+X\frac{dY}{ds}\right)ds\right]$$

By rearranging the terms. we get

$$\varepsilon=\int_0^1\left\{\frac{1}{2}\alpha\left[\left(\frac{\partial X}{\partial s}\right)^2+\left(\frac{\partial Y}{\partial s}\right)^2\right]+\frac{1}{2}\beta\left[\left(\frac{\partial^2 X}{\partial s^2}\right)^2+\left(\frac{\partial^2 Y}{\partial s^2}\right)^2\right]\right\}ds$$
$$-\left[\int_0^1\left(P\frac{dX}{ds}+\frac{1}{2}Y\frac{dX}{ds}+Q\frac{dY}{ds}-\frac{1}{2}X\frac{dY}{ds}\right)ds\right] \quad ..(2.7)$$

Let the (X(s).Y(s)) be the the contour which make ε an extremum. Let

$X(s.\xi_1)=X(s)+\xi_1 h(s)$ ..(3.7)

$Y(s.\xi_2)=Y(s)+\xi_2 h(s)$ ..(4.7)

be a neighboring contour. The energy ε for the contour defined by (3.7) and (4.7) is

$$
\begin{aligned}
\varepsilon(\xi_1,\xi_2) = & \int_0^1 \frac{1}{2}\alpha\left[\left(\frac{dX(s)}{ds}+\xi_1\frac{dh(s)}{ds}\right)^2+\left(\frac{dY(s)}{ds}+\xi_2\frac{dh(s)}{ds}\right)^2\right]ds \\
& + \int_0^1 \frac{1}{2}\beta\left[\left(\frac{d^2X(s)}{ds^2}+\xi_1\frac{d^2h(s)}{ds^2}\right)^2+\left(\frac{d^2Y(s)}{ds^2}+\xi_2\frac{d^2h(s)}{ds^2}\right)^2\right]ds \\
& - \int_0^1\left[P(X(s)+\xi_1h(s),Y(s)+\xi_2h(s))\left(\frac{dX(s)}{ds}+\xi_1\frac{dh(s)}{ds}\right)\right]ds \\
& - \int_0^1\left[\frac{1}{2}(Y(s)+\xi_2h(s))\left(\frac{dX(s)}{ds}+\xi_1\frac{dh(s)}{ds}\right)\right]ds \\
& - \int_0^1\left[Q(X(s)+\xi_1h(s),Y(s)+\xi_2h(s))\left(\frac{dY(s)}{ds}+\xi_2\frac{dh(s)}{ds}\right)\right]ds \\
& + \int_0^1\left[\frac{1}{2}(X(s)+\xi_1h(s))\left(\frac{dY(s)}{ds}+\xi_2\frac{dh(s)}{ds}\right)\right]ds \qquad ..(5.7)
\end{aligned}
$$

This being a function of ξ_1 and ξ_2 is maximum or minimum for $\xi_1=0$ and $\xi_2=0$, when $\frac{\partial\varepsilon}{\partial\xi_1}=0$ and $\frac{\partial\varepsilon}{\partial\xi_2}=0$ at $\xi_1=0$ and $\xi_2=0$.

Differentiating (5.7) with respect to ξ_1, we get,

$$
\begin{aligned}
\frac{\partial\varepsilon}{\partial\xi_1} == & \int_0^1\alpha\left[\frac{dX(s)}{ds}+\xi_1\frac{dh(s)}{ds}\right]\frac{dh(s)}{ds}ds+\int_0^1\beta\left[\frac{d^2X(s)}{ds^2}+\xi_1\frac{d^2h(s)}{ds^2}\right]\frac{d^2h(s)}{ds^2}ds \\
& - \int_0^1\left[\frac{\partial P}{\partial X}h(s)\left(\frac{dX(s)}{ds}+\xi_1\frac{dh(s)}{ds}\right)+P\frac{dh(s)}{ds}\right]ds-\int_0^1\left[\frac{1}{2}(Y(s)-\xi_2h(s))\frac{dh(s)}{ds}\right]ds \\
& - \int_0^1\left[\frac{\partial Q}{\partial X}h(s)\left(\frac{dY(s)}{ds}+\xi_2\frac{dh(s)}{ds}\right)\right]ds+\int_0^1\left[\frac{1}{2}h(s)\left(\frac{dY(s)}{ds}+\xi_2\frac{dh(s)}{ds}\right)\right]ds \qquad ..(6.7)
\end{aligned}
$$

Making $\frac{\partial \varepsilon}{\partial \xi_1} = 0$ at $\xi_1=0$ and $\xi_2=0$,

$$\int_0^1 \alpha \frac{dX(s)}{ds}\frac{dh(s)}{ds}ds + \int_0^1 \beta \frac{d^2X(s)}{ds^2}\frac{d^2h(s)}{ds^2}ds - \int_0^1 \left(\frac{\partial P}{\partial X}h(s)\frac{dX(s)}{ds} + P\frac{dh(s)}{ds}\right)ds - \int_0^1 \frac{1}{2}Y(s)\frac{dh(s)}{ds}ds$$
$$- \int_0^1 \frac{\partial Q}{\partial X}h(s)\frac{dY(s)}{ds}ds + \int_0^1 \frac{1}{2}h(s)\frac{dY(s)}{ds}ds = 0. \ (7.7)$$

A basic formula of differential calculus is:

$\int d(uv) = \int udv + \int vdu$ Which implies $\int vdu = uv - \int udv$. (8.7)

Implying the result of (8.7) different terms of equation (7.7) is evaluated below:

$$\int_0^1 \alpha \frac{dX(s)}{ds}\frac{dh(s)}{ds}ds = \int_0^1 \alpha \frac{dX(s)}{ds}dh(s)$$

Let u= $h(s)$ and v= $\frac{dX(s)}{ds}$ then as per equation (8.7) above integral become:

$$\int_0^1 \alpha \frac{dX(s)}{ds}\frac{dh(s)}{ds}ds = \alpha\left[h(s)\frac{dX(s)}{ds}\right]_0^1 - \alpha\int_0^1 h(s)\frac{d^2X(s)}{ds^2}ds$$

As s=0,s=1 represent same point, first term of integral is zero, it can be written as:

$$\int_0^1 \alpha \frac{dX(s)}{ds}\frac{dh(s)}{ds}ds = -\alpha\int_0^1 h(s)\frac{d^2X(s)}{ds^2}ds. \ (9.7)$$

Similarly,

$$\int_0^1 \beta \frac{d^2X(s)}{ds^2}\frac{d^2h(s)}{ds^2}ds = \int_0^1 \beta \frac{d^2X(s)}{ds^2}d\left(\frac{dh(s)}{ds}\right)$$

Let u= $\frac{dh(s)}{ds}$ and v= $\frac{d^2X(s)}{ds^2}$ then as per equation (8.7) above integral become

$$= \beta\left[\frac{dh(s)}{ds}\frac{d^2X(s)}{ds^2}\right]_0^1 - \beta\int_0^1 \frac{dh(s)}{ds}\frac{d^3X(s)}{ds^3}ds$$

$$= -\beta \int_0^1 \frac{dh(s)}{ds}\frac{d^3X(s)}{ds^3}ds$$ (As s=0 and s=1 are same point, first term of integral is zero)

$$= -\beta \int_0^1 \frac{d^3X(s)}{ds^3}dh(s)$$

$$= -\beta\left[h(s)\frac{d^3X(s)}{ds^3}\right]_0^1 + \beta\int_0^1 h(s)\frac{d^4X(s)}{ds^4}ds$$ (With help of (8.7))

$$= \beta\int_0^1 h(s)\frac{d^4X(s)}{ds^4}ds$$ (As s=0 and s=1 are same point, first term of integral is zero)

Result of derivation is:

$$\int_0^1 \beta\frac{d^2X(s)}{ds^2}\frac{d^2h(s)}{ds^2}ds = \beta\int_0^1 h(s)\frac{d^4X(s)}{ds^4}ds \qquad ..(10.7)$$

Now we are going to simplify another two integrals given below:

$$\int_0^1 P\frac{dh(s)}{ds}ds,\ \int_0^1 \frac{1}{2}Y(s)\frac{dh(s)}{ds}ds$$

comparing the integral with (9.7) it can be directly written:

$$\int_0^1 P\frac{dh(s)}{ds}ds = -\int_0^1 h(s)\frac{dP}{ds}ds = -\int_0^1 h(s)\left(\frac{\partial P}{\partial X}\frac{dX}{ds}+\frac{\partial P}{\partial Y}\frac{dY}{ds}\right)ds \qquad ..(11.7)$$

$$\int_0^1 \frac{1}{2}Y(s)\frac{dh(s)}{ds}ds = -\int_0^1 \frac{1}{2}h(s)\frac{dY(s)}{ds}ds \qquad ..(12.7)$$

With the help of the equations (9.7), (10.7), (11.7), (12.7) the equation (7.7)can be written as:

$$-\alpha\int h(s)\frac{d^2X(s)}{ds^2}ds + \beta\int_0^1 h(s)\frac{d^4X(s)}{ds^4}ds - \int_0^1\frac{\partial P}{\partial X}h(s)\frac{dX(s)}{ds}ds + \int_0^1 h(s)\left(\frac{\partial P}{\partial X}\frac{dX}{ds}+\frac{\partial P}{\partial Y}\frac{dY}{ds}\right)ds$$
$$+\int\frac{1}{2}h(s)\frac{dY(s)}{ds}ds - \int_0^1\frac{\partial Q}{\partial X}h(s)\frac{dY(s)}{ds}ds + \int_0^1\frac{1}{2}h(s)\frac{dY(s)}{ds}ds = 0$$

By simplification the equation becomes:

$$\int_0^1\left(-\alpha\frac{d^2X(s)}{ds^2} + \beta\frac{d^4X(s)}{ds^4} + \frac{\partial P}{\partial Y}\frac{dY(s)}{ds} + \frac{dY(s)}{ds} - \frac{\partial Q}{\partial X}\frac{dY(s)}{ds}\right)h(s)ds = 0 \qquad ..(13.7)$$

As $h(s)$ is an arbitrary function, from above equation it can be written as:

$$-\alpha \frac{d^2X(s)}{ds^2}+\beta \frac{d^4X(s)}{ds^4}+\frac{\partial P}{\partial Y}\frac{dY(s)}{ds}+\frac{dY(s)}{ds}-\frac{\partial Q}{\partial X}\frac{dY(s)}{ds}=0 \qquad ..(14.7)$$

Now $\frac{\partial P}{\partial Y}=\frac{\partial}{\partial Y}\left(-\frac{1}{2}\int_0^Y \varepsilon_p(X,t)dt\right)$

Apply the Leibnitz's rule,

$$\frac{\partial P}{\partial Y}=-\frac{1}{2}\varepsilon_p(X,Y)$$

In similar way

$$-\frac{\partial Q}{\partial X}=-\frac{\partial}{\partial X}\left(\frac{1}{2}\int_0^X \varepsilon_p(t,Y)dt\right)$$

Apply the Leibnitz's rule.

$$-\frac{\partial Q}{\partial Y}=-\frac{1}{2}\varepsilon_p(X,Y)$$

Putting the values of $\frac{\partial P}{\partial Y}$ and $-\frac{\partial Q}{\partial X}$ in equation (14.7)

$$\left(-\varepsilon_p(X,Y)+1\right)\frac{dY}{ds}-\alpha\frac{d^2X(s)}{ds^2}+\beta\frac{d^4X(s)}{ds^4}=0 \qquad ..(15.7)$$

Writing this in terms of discrete points

$$\left(-\varepsilon_p(X_i,Y_i)+1\right)\left(Y_{i+1}-Y_i\right)-\alpha\left(X_{i+1}-2X_i+X_{i-1}\right)+\beta\left(X_{i+2}-4X_{i-1}+6X_i-4X_{i-1}+X_{i-2}\right)=0$$

$$\left(-\varepsilon_p(X_i,Y_i)+1\right)\left(Y_{i+1}-Y_i\right)+\beta X_{i-2}+\left(-\alpha-4\beta\right)X_{i-1}+\left(2\alpha+6\beta\right)X_i+\left(-\alpha-4\beta\right)X_{i+1}$$

$$+\beta X_{i+2}=0 \qquad ..(16.7)$$

If contour have n number of points, then n no of equations come out putting i=1,2,...,n in (16.7). The equation can expressed in matrix notation as:

$$\left(-\begin{bmatrix} \varepsilon_p(X_1,Y_1) & 0 & . & . & 0 \\ 0 & \varepsilon_p(X_2,Y_2) & . & . & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ 0 & 0 & . & . & \varepsilon_p(X_n,Y_n) \end{bmatrix} + \mathbf{I}\right)\left(\begin{bmatrix} Y_2 \\ Y_3 \\ . \\ Y_n \\ Y_1 \end{bmatrix} - \begin{bmatrix} Y_1 \\ Y_2 \\ . \\ Y_{n-1} \\ Y_n \end{bmatrix}\right) +$$

$$\begin{bmatrix} (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & 0 & 0 & . & . & 0 & \beta & (-\alpha-4\beta) \\ (-\alpha-4\beta) & (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & 0 & . & . & 0 & 0 & \beta \\ \beta & (-\alpha-4\beta) & (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & . & . & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . & . & . \\ (-\alpha-4\beta) & \beta & 0 & 0 & 0. & . & . & \beta & (-\alpha-4\beta) & (2\alpha+6\beta) \end{bmatrix} \mathrm{X}$$

$$\begin{bmatrix} X_1 \\ X_2 \\ . \\ X_{n-1} \\ X_n \end{bmatrix} = 0 \qquad ..(17.7)$$

I is an identity matrix of same dimensionality with $\mathbf{E_p}$. The equation can be written in a vector notation form:

$$\left(-\mathbf{E_p}+\mathbf{I}\right)\left(\bar{Y}'-\bar{Y}\right)-\mathbf{A}\bar{Y}=0 \qquad ..(18.7)$$

where,

$$\mathbf{E_p} = \begin{bmatrix} \varepsilon_p(X_1,Y_1) & 0 & . & . & 0 \\ 0 & \varepsilon_p(X_2,Y_2) & . & . & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ 0 & 0 & . & . & \varepsilon_p(X_n,Y_n) \end{bmatrix}$$

$$\bar{Y}'=\begin{bmatrix} Y_2 \\ Y_3 \\ . \\ Y_n \\ Y_1 \end{bmatrix} \text{ and } \bar{Y}=\begin{bmatrix} Y_1 \\ Y_2 \\ . \\ Y_{n-1} \\ Y_n \end{bmatrix} \text{and } \bar{X}=\begin{bmatrix} X_1 \\ X_2 \\ . \\ X_{n-1} \\ X_n \end{bmatrix}$$

$$\text{and } \mathbf{A}=\begin{bmatrix} (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & 0 & 0 & . & . & 0 & \beta & (-\alpha-4\beta) \\ (-\alpha-4\beta) & (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & 0 & . & . & 0 & 0 & \beta \\ \beta & (-\alpha-4\beta) & (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & . & . & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . & . & . \\ (-\alpha-4\beta) & \beta & 0 & 0 & 0. & . & . & \beta & (-\alpha-4\beta) & (2\alpha+6\beta) \end{bmatrix}$$

Equation (18.7) is the extrema condition of energy function (8.3). Gradient decent method is used to find out the contour for which energy function is minimum. It is a numerical method. In this method extrem value of a function is evaluated by finding out the values of its variables for which function is extremum. Initially the condition for minima is evaluated in arbitrary current position. It is obviously not satisfy extrema condition. Then, this evaluated extrema condition is subtracted from variables to get the new values of the variables, which are more close to minima. We applied this procedure iteratively applied until the minima condition comes out correctly. In gradient decent method numerical equation for finding out X coordinates of contour is:

$$\bar{X}^{t+1}-\bar{X}^{t}=-\left[\left(-\mathbf{E}_{\mathbf{p}}-\mathbf{I}\right)\left(\bar{Y}'^{t}-\bar{Y}^{t}\right)+\mathbf{A}\bar{X}^{t}\right] \qquad ..(19.7)$$

where $\bar{X}^{t+1}$ denotes X coordinate values at t+1th iteration and $\bar{X}^{t}$ denotes X coordinate values at t th iteration. But equation (18.7) is numerically unstable [9]. Corresponding numerical stable equation is.

$$\bar{X}^{t+1}=(\mathbf{I}+\mathbf{A})^{-1}\left(\left(-\mathbf{E}_{\mathbf{p}}-\mathbf{I}\right)\left(\bar{Y}'^{t}-\bar{Y}^{t}\right)+\bar{X}^{t}\right)\Delta t \qquad ..(20.7)$$

where Δt is the time step by which amount of movement of contour is controlled for numerical stability. Equation (19.7) modifies the X coordinates of contour.

Differentiating (5.7) with respect to ξ_2, we get.

$$\frac{\partial\varepsilon}{\partial\xi_2}==\int_0^1\alpha\left[\frac{dY(s)}{ds}+\xi_2\frac{dh(s)}{ds}\right]\frac{dh(s)}{ds}ds+\int_0^1\beta\left[\frac{d^2Y(s)}{ds^2}+\xi_2\frac{d^2h(s)}{ds^2}\right]\frac{d^2h(s)}{ds^2}ds$$
$$-\int_0^1\left[\frac{\partial P}{\partial Y}h(s)\left(\frac{dX(s)}{ds}+\xi_1\frac{dh(s)}{ds}\right)\right]ds-\int_0^1\left[\frac{1}{2}h(s)\left(\frac{dX(s)}{ds}+\xi_1\frac{dh(s)}{ds}\right)\right]ds$$
$$-\int_0^1\left[\frac{\partial Q}{\partial Y}h(s)\left(\frac{dY(s)}{ds}+\xi_2\frac{dh(s)}{ds}\right)+Q\frac{dh(s)}{ds}\right]ds+\int_0^1\left[\frac{1}{2}(X(s)+\xi_1h(s))\frac{dh(s)}{ds}\right]ds \quad ..(21.7)$$

Making $\frac{\partial\varepsilon}{\partial\xi_2}=0$ at $\xi_1=0$ and $\xi_2=0$.

$$\int_0^1\alpha\frac{dY(s)}{ds}\frac{dh(s)}{ds}ds+\int_0^1\beta\frac{d^2Y(s)}{ds^2}\frac{d^2h(s)}{ds^2}ds-\int_0^1\frac{\partial P}{\partial Y}h(s)\frac{dX(s)}{ds}ds-\int_0^1\frac{1}{2}h(s)\frac{dX(s)}{ds}ds$$
$$-\int_0^1\left(\frac{\partial Q}{\partial Y}h(s)\frac{dY(s)}{ds}+Q\frac{dh(s)}{ds}\right)ds+\int_0^1\frac{1}{2}X(s)\frac{dh(s)}{ds}ds=0 \quad ..(22.7)$$

Implying the result of (8.7) different terms of equation (21.7) is evaluated below:

$$\int_0^1\alpha\frac{dY(s)}{ds}\frac{dh(s)}{ds}ds=\int_0^1\alpha\frac{dY(s)}{ds}dh(s)$$

let u= $h(s)$ and v= $\frac{dY(s)}{ds}$ then as per equation(21.7)the integral become become:

$$\int_0^1\alpha\frac{dY(s)}{ds}\frac{dh(s)}{ds}ds=\alpha\left[h(s)\frac{dY(s)}{ds}\right]_0^1-\alpha\int_0^1h(s)\frac{d^2Y(s)}{ds^2}ds$$

As s=0.s=1 represent same point so first term of integral is zero.it can be written as:

$$\int_0^1\alpha\frac{dY(s)}{ds}\frac{dh(s)}{ds}ds=-\alpha\int_0^1h(s)\frac{d^2Y(s)}{ds^2}ds \quad ..(23.7)$$

Similarly.

$$\int_0^1\beta\frac{d^2Y(s)}{ds^2}\frac{d^2h(s)}{ds^2}ds=\int_0^1\beta\frac{d^2Y(s)}{ds^2}d\left(\frac{dh(s)}{ds}\right)$$

Let u= $\frac{dh(s)}{ds}$ and v= $\frac{d^2Y(s)}{ds^2}$ then as per equation (8.7) above integral become:

$$= \beta \left[\frac{dh(s)}{ds} \frac{d^2 Y(s)}{ds^2} \right]_0^1 - \beta \int_0^1 \frac{dh(s)}{ds} \frac{d^3 Y(s)}{ds^3} ds$$

$$= - \beta \int_0^1 \frac{dh(s)}{ds} \frac{d^3 Y(s)}{ds^3} ds$$ (As s=0.s=1 represent same point so first term,of integral is zero)

$$= - \beta \int_0^1 \frac{d^3 Y(s)}{ds^3} dh(s)$$

$$= - \beta \left[h(s) \frac{d^3 Y(s)}{ds^3} \right]_0^1 + \beta \int_0^1 h(s) \frac{d^4 Y(s)}{ds^4} ds$$ (with help of equation (8.7))

$$= \beta \int_0^1 h(s) \frac{d^4 Y(s)}{ds^4} ds$$ As s=0.s=1 represent same point so first term of integral is zero

Result of derivation is:

$$\int_0^1 \beta \frac{d^2 Y(s)}{ds^2} \frac{d^2 h(s)}{ds^2} ds = \beta \int_0^1 h(s) \frac{d^4 Y(s)}{ds^4} ds \qquad ..(24.7)$$

Now we are going to simplify another two integrals given below:

$$\int_0^1 Q \frac{dh(s)}{ds} ds \,.\, \int_0^1 \frac{1}{2} X(s) \frac{dh(s)}{ds} ds$$

comparing the integral with (22.7) it can be directly written:

$$\int_0^1 Q \frac{dh(s)}{ds} ds = - \int_0^1 h(s) \frac{dQ}{ds} ds = - \int_0^1 h(s) \left(\frac{\partial Q}{\partial X} \frac{dX}{ds} + \frac{\partial Q}{\partial Y} \frac{dY}{ds} \right) ds \qquad ..(25.7)$$

$$\int_0^1 \frac{1}{2} X(s) \frac{dh(s)}{ds} ds = - \int_0^1 \frac{1}{2} h(s) \frac{dX(s)}{ds} ds \qquad ..(26.7)$$

With the help of the equations (23.7). (24.7). (25.7). (26.7) the equation (22.7)can be written as:

$$- \alpha \int_0^1 h(s) \frac{d^2 Y(s)}{ds^2} ds + \beta \int_0^1 h(s) \frac{d^4 Y(s)}{ds^4} ds - \int_0^1 \frac{\partial P}{\partial Y} h(s) \frac{dX(s)}{ds} ds - \int_0^1 \frac{1}{2} h(s) \frac{dX(s)}{ds} ds - \int_0^1 \frac{\partial Q}{\partial Y} h(s) \frac{dY(s)}{ds} ds +$$
$$\int_0^1 h(s) \left(\frac{\partial Q}{\partial X} \frac{dX}{ds} + \frac{\partial Q}{\partial Y} \frac{dY}{ds} \right) ds - \int_0^1 \frac{1}{2} h(s) \frac{dX(s)}{ds} ds = 0$$

By simplification the equation becomes:

$$\int_0^1 \left(-\alpha \frac{d^2 X(s)}{ds^2} + \beta \frac{d^4 X(s)}{ds^4} - \frac{\partial P}{\partial Y}\frac{dY(s)}{ds} - \frac{dX(s)}{ds} + \frac{\partial Q}{\partial X}\frac{dX(s)}{ds} \right) h(s)ds = 0 \qquad ..(27.7)$$

As $h(s)$ is an arbitrary function, from above equation it can be written

$$-\alpha \frac{d^2 X(s)}{ds^2} + \beta \frac{d^4 X(s)}{ds^4} - \frac{\partial P}{\partial Y}\frac{dY(s)}{ds} - \frac{dX(s)}{ds} + \frac{\partial Q}{\partial X}\frac{dX(s)}{ds} = 0 \qquad ..(28.7)$$

Now $-\frac{\partial P}{\partial Y} = -\frac{\partial}{\partial Y}\left(-\frac{1}{2}\int_0^Y \varepsilon_p (X,t)dt \right)$

Apply the Leibnitz's rule.

$$\frac{\partial P}{\partial Y} = \frac{1}{2}\varepsilon_p (X,Y)$$

Now $\frac{\partial Q}{\partial X} = \frac{\partial}{\partial X}\left(\frac{1}{2}\int_0^X \varepsilon_p (t,Y)dt \right)$

Apply the Leibnitz's rule.

$$\frac{\partial Q}{\partial Y} = \frac{1}{2}\varepsilon_p (X,Y)$$

so putting the values of $-\frac{\partial P}{\partial Y}$ and $\frac{\partial Q}{\partial X}$ in equation (27.7)

$$(\varepsilon_p (X,Y) - 1)\frac{dX(s)}{ds} - \alpha \frac{d^2 Y(s)}{ds^2} + \beta \frac{d^4 Y(s)}{ds^4} = 0$$

Writing this in terms of discrete points

$$(\varepsilon_p (X_i,Y_i) - 1)(X_{i+1} - X_i) - \alpha (Y_{i-1} - 2Y_i + Y_{i+1}) + \beta (Y_{i+2} - 4Y_{i+1} + 6Y_i - 4Y_{i-1} + Y_{i-2}) = 0$$

$$(\varepsilon_p (X_i,Y_i) - 1)(X_{i+1} - X_i) + \beta Y_{i-2} + (-\alpha - 4\beta)Y_{i-1} + (2\alpha + 6\beta)Y_i + (-\alpha - 4\beta)Y_{i+1}$$

$$+ \beta Y_{i+2} = 0 \qquad ..(29.7)$$

If contour have n number of points, then n no of equations come out putting i=1,2,...n in (28.7).The equation can expressed in matrix notation as:

$$\left(\begin{bmatrix} \varepsilon_p(X_1,Y_1) & 0 & . & . & 0 \\ 0 & \varepsilon_p(X_2,Y_2) & . & . & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ 0 & 0 & . & . & \varepsilon_p(X_n,Y_n) \end{bmatrix} - \mathbf{I}\right)\left(\begin{bmatrix} X_2 \\ X_3 \\ . \\ X_n \\ X_1 \end{bmatrix} - \begin{bmatrix} X_1 \\ X_2 \\ . \\ X_{n-1} \\ X_n \end{bmatrix}\right) +$$

$$\begin{bmatrix} (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & 0 & 0 & . & . & 0 & \beta & (-\alpha-4\beta) \\ (-\alpha-4\beta) & (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & 0 & . & . & 0 & 0 & \beta \\ \beta & (-\alpha-4\beta) & (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & . & . & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . & . & . \\ (-\alpha-4\beta) & \beta & 0 & 0 & 0. & . & . & \beta & (-\alpha-4\beta) & (2\alpha+6\beta) \end{bmatrix} \mathrm{X}$$

$$\begin{bmatrix} Y_1 \\ Y_2 \\ . \\ Y_{n-1} \\ Y_n \end{bmatrix} = 0 \qquad ..(30.7)$$

The equation can be written in vector form as:

$$\mathbf{E}_p - \mathbf{I}\big)\big(\bar{X}' - \bar{X}\big) + \mathbf{A}\bar{Y} = 0 \qquad ..(31.7)$$

where,

E_r is the energy matrix and

$$\mathbf{E}_p = \begin{bmatrix} \varepsilon_p(X_1,Y_1) & 0 & . & . & 0 \\ 0 & \varepsilon_p(X_2,Y_2) & . & . & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ 0 & 0 & . & . & \varepsilon_p(X_n,Y_n) \end{bmatrix}$$

$$\bar{X}' = \begin{bmatrix} X_2 \\ X_3 \\ . \\ X_n \\ X_1 \end{bmatrix} \text{ and } \bar{Y} = \begin{bmatrix} Y_1 \\ Y_2 \\ . \\ Y_{n-1} \\ Y_n \end{bmatrix} \text{ and } \bar{X} = \begin{bmatrix} X_1 \\ X_2 \\ . \\ X_{n-1} \\ X_n \end{bmatrix}$$

and

$$\mathbf{A} = \begin{bmatrix} (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & 0 & 0 & . & . & 0 & \beta & (-\alpha-4\beta) \\ (-\alpha-4\beta) & (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & 0 & . & . & 0 & 0 & \beta \\ \beta & (-\alpha-4\beta) & (2\alpha+6\beta) & (-\alpha-4\beta) & \beta & . & . & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . & . & . \\ (-\alpha-4\beta) & \beta & 0 & 0 & 0. & . & . & \beta & (-\alpha-4\beta) & (2\alpha+6\beta) \end{bmatrix}$$

Equation (29.7) is the extrema condition of energy function (8.3). Corresponding numerical equation in gradient decent method for finding out Y coordinates of contour is:

$$\bar{Y}^{t+1} - \bar{Y}^{t} = -\left[\left(\mathbf{E}_p - \mathbf{I}\right)\left(\bar{X}'' - \bar{X}'\right) + \mathbf{A}\bar{Y}^{t}\right]$$

where $\bar{Y}^{t+1}$ denotes Y coordinate values at t+1th iteration and $\bar{Y}^{t}$ denotes Y coordinate values at t th iteration. But equation (18.7) is numerically unstable [9]. Corresponding numerical stable equation is.

$$\bar{Y}^{t+1} = (\mathbf{I}+\mathbf{A})^{-1}\left(\left(\mathbf{E}_p - \mathbf{I}\right)\left(\bar{X}'' - \bar{X}'\right) + \bar{Y}^{t}\right)\Delta t \qquad \ldots(32.7)$$

where Δt is the time step by which amount of movement of contour is controlled for numerical stability. Equation (30.7) modifies the Y coordinates of contour.