إقرار

أنا الموقع أدناه مقدم الرسالة التي تحمل العنوان:

* طريقة كنتور نشط ثنائية المراحل لاجتزاء الأجسام من الصور بالرغم من تشوش الخلفية وتقعرات الحدود

* A Two-Phase Snake Method for Object Segmentation against Background Clutter and Boundary Concavities.

أقر بأن ما اشتملت عليه هذه الرسالة إنما هي نتاج جهدي الخاص، باستثناء ما تمت الإشارة إليه حيثما ورد، وإن هذه الرسالة ككل، أو أي جزء منها لم يقدم من قبل لنيل درجة أو لقب علمي أو بحثي لدى أية مؤسسة تعليمية أو بحثية أخرى.

DECLARATION

The work provided in this thesis, unless otherwise referenced, is the researcher's own work, and has not been submitted elsewhere for any other degree or qualification

Student's name: Nihal Mohd Saqer	اسم الطالب: نهال محمد صقر
Signature	التوقيع:
Date: 24/6/2015	التاريخ: ٢٤/٦/٢٠١٥م

بسم الله الرحمن الرحيم

Islamic University – Gaza

Deanery of Post Graduate Studies

Faculty of Information Technology

الجامعة الإسلامية – غزة

عمادة الدراسات العليا

كلية تكنولوجيا المعلومات

A Two-Phase Snake Method for Object Segmentation against Background Clutter and Boundary Concavities

Submitted by:

Nihal Mohammed Saqer

220100197

Supervised by:

Dr. Ashraf Alattar

A Thesis Submitted in Partial Fulfillment of Requirements for the Degree of Master in Information Technology

1436 H – May 2015

بسم الله الرحمن الرحيم

الجامعة الإسلامية – غزة
The Islamic University - Gaza

مكتب نائب الرئيس للبحث العلمي والدراسات العليا هاتف داخلي: 1150

الرقم ج س غ/35/ Ref

التاريخ 2015/06/02م Date

نتيجة الحكم على أطروحة ماجستير

بناءً على موافقة شئون البحث العلمي والدراسات العليا بالجامعة الإسلامية بغزة على تشكيل لجنة الحكم على أطروحة الباحثة/ **نهال محمد صالح صقر** لنيل درجة الماجستير في **كلية *تكنولوجيا المعلومات*** **برنامج تكنولوجيا المعلومات** وموضوعها:

طريقةُ كنتورٍ نشطٍ ثنائيةُ المراحلِ لاجتزاءِ الأجسامِ من الصور بالرغم من تشوش الخلفية وتقعرات الحدود

A Two-Phase Snake Method for Object Segmentation against Background Clutter and Boundary Concavities

وبعد المناقشة التي تمت اليوم الثلاثاء 15 شعبان 1436هـ، الموافق 2015/06/02م الساعة الحادية عشرة صباحاً، اجتمعت لجنة الحكم على الأطروحة والمكونة من:

د. أشرف محمد العطار	مشرفاً و رئيساً	....
د. ربحي سليمان بركة	مناقشاً داخلياً	....
د. سناء وفا الصايغ	مناقشاً خارجياً	....

وبعد المداولة أوصت اللجنة بمنح الباحثة درجة الماجستير في **كلية *تكنولوجيا المعلومات*/** برنامج **تكنولوجيا المعلومات.**

واللجنة إذ تمنحها هذه الدرجة فإنها توصيها بتقوى الله ولزوم طاعته وأن تسخر علمها في خدمة دينها ووطنها.

والله ولي التوفيق،،،

مساعد نائب الرئيس للبحث العلمي والدراسات العليا

أ.د. فؤاد علي العاجز

P.O. Box 108, Rimal, Gaza, Palestine fax: +970 (8) 286 0800 فاكس Tel: +970 (8) 286 0700 ص.ب. 108 الرمال، غزة، فلسطين هاتف
public@iugaza.edu.ps www.iugaza.edu.ps

Abstract

Various object segmentation methods have been proposed based on the classical Active Contour Model (Snake), which has been used extensively to locate object boundaries in images. However, these methods have limited capability in overcoming the problems of background clutter and boundary concavities. Background clutter has a noise, which obstruct the snake moving and determining the Object Of Interest (OOI), also the snake ability suffers to move into boundary concavities.

In this research, we propose a new snake method that can perform more efficiently in the presence of background clutter and boundary concavities. Our approach will use two- phase snake instead of a greedy snake algorithm, which works by computed energy functions on neighborhood around each snake point , then move to the position with the lowest energy, and it will use scale space continuation to increase the snake ability to find the OOI contour from cluttered background.

The first snake-phase try to converge on edges until stopping but that doesn't mean founding the OOI boundary. After that the second snake-phase starts its completing until the boundary of the object of interest is found.

The two- phase snake method is testing on a number of different images under most conditions. The experimental results errors also will be compared on the two- phase snake.

Performance of the new method will be evaluated in terms of accuracy and performance compared with existing methods, which are based on the classical model.

Keywords:

Background clutter, snake model, boundary concavities, object segmentation, greedy snake.

الملخص

تعتمد طرق اجتزاء الأجسام المختلفة على النموذج التقليدي للكنتور النشط ، والذي استخدم لتحديد موضع الحدود في الصور .

وهذه الطرق لم تستطع حل مشكلة تجاوز تشوشات الخلفية والتعمق بداخل تقعرات الحدود بالشكل المطلوب. في هذا البحث نقترح طريقة جديدة تستطيع التعامل مع المشكلتين السابقتين بشكل أكثر فعالية.

تعتمد فكرة العمل على استخدام الكنتور النشط الذي يعمل بخوارزمية greedy في مرحلتين ، ففي المرحلة الأولى يبدأ الكنتور عمله إلى أن يستقر (بشكل غير نهائي)، ثم تبدأ المرحلة الثانية بأن يحسب الكنتور تلقائياً مواضع جديدة لنقاطه ينتقل إليها ثم يواصل العمل مرة أخرى إلى أن يستقر عند حدود الجسم المطلوب .

أظهرت نتائج تقييم أداء هذه الطريقة من حيث نسبة الخطأ ومقارنتها بالطرق المماثلة تفوقاً واضحاً في الدقة في جميع الحالات التي تم اختبارها.

كلمات مفتاحية : الكنتور النشط – اجتزاء الأجسام – تقعرات الحدود – الخلفية المشوشة – الثعبان النشط.

Acknowledgement

*First, I thank **Allah** for guiding me and taking care of me all the time.*

*I would like to thank My Family especially **My Parents, My Mother and Father in law , and My Husband** for encouraging and supporting me all the time.*

*Also, I would like to take this opportunity to thank all My Teachers and My research supervisor, **Dr. Ashraf Alattar** for giving me the opportunity to work with him, guiding and helping me throughout this research and other courses.*

*I wish to express my considerable gratitude to my special **friends** and **colleges**, for there supports.*

Thank you all for being always there when I need you most.

Thank you for believing in me and supporting me.

Table of Contents

Table of Figures

List Of Tables

LIST OF ABBREVIATIONS

ROI	Region Of Interest
OOI	Object Of Interest
ACM	Active Contour Model
GAC	Geodesic Active Contours
GVF	Gradient Vector Flow
GGVF	Generalized Gradient Vector Flow
P1E	Phase 1 Error
P2E	Phase 2 Error
GUI	Graphical User Interface
TH	Threshold
RMSE	Root Mean Square Error

Chapter (1):

Introduction

1. CHAPTER 1: INTRODUCTION

Object segmentation is an important field in image segmentation, we choose this field because it is an exciting one, and this will be clear in our research.

1.1. Object Segmentation

Object segmentation is the process of extracting an Object Of Interest (OOI) from the rest of an image; the background. This is technically different from image segmentation, which refers to the process of dividing an image into regions or categories that correspond to different parts of the image. Both types of segmentation (object segmentation, and image segmentation) can be considered a type of pixel classification. In image segmentation every pixel in the image is classified to one of the different image regions, while in object segmentation it is classified as either object or background pixel.

However, in many cases, the objective of object segmentation is not to classify every single pixel of the OOI, but to only find the object's outer boundary. The object's boundary is a meaningful representation of the object, and once it is found it becomes a straightforward step to delete the background and extract the object.

1.1.1. Object Segmentation Applications & Importantance .

A segmentation could be used for object recognition, occlusion boundary estimation within motion, stereo systems, image compression, image editing, and image database look-up.

The object segmentation operation used in various systems such as, Graphics design applications, multimedia design applications, object-based video encoding systems, medical imaging systems, and security applications .

The object segmentation is very useful for tracking and recognition the object in a video, for example, pedestrian and highway traffic can be regularized using density evaluations obtained by segmenting people and vehicles. Using object segmentation, speeding and suspicious moving cars, road obstacles, strange activities can be detected.

The medical applications mostly require the segmentation for specified objects, such as specific organs or lesions. In medical imaging, the resulting contours after object segmentation can be used to create 3D reconstructions with the help of interpolation algorithms like Marching cubes. Segmentation of tumors from CT or MRI images can be critical in the treatment of cancer.

Security systems demand 3D segmentation algorithms that can quickly and reliably detect threats.

In salient object segmentation, image labelers annotate the saliency in an image by drawing pixel-accurate silhouettes of objects that they believed to be salient.

1.2. The Research Problem

Boundary methods have two main obstacles a) background clutter, and b) boundary concavities.

1.2.1. Background Clutter

Background clutter is the existence of various scene components in the background of the image. These components result in noise edges which obstruct the active contour and prevent it from reaching and converging on the interest object's boundary. Essentially, the contour falls in local minima problem before reaching the object's boundary, see Figure 1.5.

Figure 1.5: OOI (real image identified flower) on background clutter.

Variational methods solve this problem by means of energy minimization. Methods belonging to this class are generally divided into two types, the region-based and boundary-based approaches [1],[2] have advantages and disadvantages , such the gradient is very noise sensitive , and if the high frequency information in the image either is missing or is unreliable, boundary finding is more error.

1.2.2. Boundary Concavities

Boundary concavities is the existence of concavities in the object's boundary. Snake points cannot reach boundary concavities because they are restricted with internal snake forces, see Figure 1.6. If a curve is concave up (convex), the graph of the curve is bent upward, like an upright bowl. If a curve is concave down (or simply concave), then the graph of the curve is bent down, like a bridge.

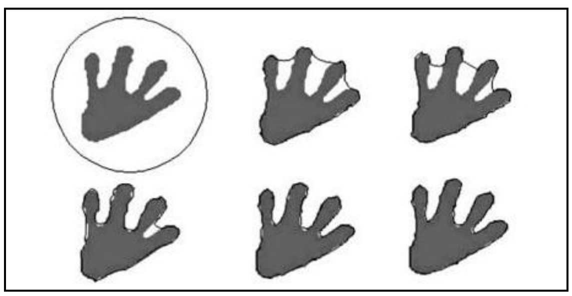


Figure 1.6: Object with several deep concavities.

Avoiding background clutter, boundary concavities many various approaches initialized by the active contours were pioneered with the classical snakes [3], the snake as close to the OOI as possible and in a way as to exclude background clutter from the snake's processing field.

Some research solutions have been proposed to handle our research problem " boundary concavities " such as the Gradient Vector Flow (GVF) snake proposed by Xu et al. in[4], [5],[6], Also others used balloon snake which behaves like a balloon is blown up. Which modifies the definition external forces (derived from gradient of the image) presented in traditional snake Kass [3]. A Geometric Active Contour, are typically implemented using level set techniques where the snake is embedded as the zero level set of a higher dimensional function, see [7], [5], [8], [9]. This implementation enables GAC snakes to handle boundary concavities and topology changes naturally. While the GAC model, the GVF model and the various other ACM variants show that the model can be modified to improve its performance against the concavity problem, the background clutter problem remains a challenge, which cannot be treated through modifications to the model [10], [11].

In other approaches, shape models have been used, such as in [7], [12], to identify the OOI and its edges, In addition to that it classifies all other edges as background clutter. While such methods may work for specific applications where prior knowledge about the OOI is available, obviously they cannot be used outside of their domain [2] [3].

In the related works section we present in more details some of the most common segmentation methods that have been proposed based on the classical snake model and non-snake-based.

As we shown ,snake-based methods are among the primary methods and are commonly used in applications such as object tracking, shape recognition, segmentation, medical and edge detection [15], [16], [17], [18],[19],[20],[21], however, performance of the snake model degrades significantly when the snake encounters challenges such as boundary concavities, object occlusion, and background clutter.

In this work, we develop a snake method depending on snake-based methods. This method will depend on employing snake method within a certain structure that ensures its effectiveness and accuracy of the output to object segmentation from cluttered background and boundary concavities.

The user of the application will input the image and some values, and the application will output the contour of OOI.

1.3. Problem Statement

Object segmentation methods based on the classical snake model have limited capability in overcoming the problems of background clutter and boundary concavities, because if they extract an Object Of Interest (OOI) from the rest of an image, cluttered background obstruct converging OOI , and if the OOI have concavity boundary, snake cant converge the OOI more efficiently.

1.4. Objectives

This research has a main and specific objectives through which we can achieve the solution to our problem.

1.4.1. Main Objective

To develop a snake-based object segmentation method that can perform time efficiently in the presence of background clutter and boundary concavity.

1.4.2. Specific Objectives

1. To design a new snake method

   A two- phase snake is designed, which have two stages, the first phase is a greedy snake and then adding the second phase after shifting the first control points snake to a new start position.

2. To implement the new method

   The implementation has a user interface which, the user can load image, specify a ROI and parameters through it, then start the two- phase snake.

3. To test the proposed method

   Testing the proposed method depends on using real and synthesized images also there are three test sets.

4. To evaluate the proposed method

   Evaluate the accuracy of our improved method by using the square root of the mean/average of the square of all the error.

5. To make comparative between the proposed method and others

   Making a comparison between the proposed method and greedy snake method depends on the results from the testing of the greedy snake and the proposed method.

1.5. Significance of the thesis

Enhance performance (in terms of time and accuracy) of object segmentation operation used in various systems such as:

- Graphics design applications

- Multimedia design applications
- Object-based video encoding systems
- Medical imaging systems
- Security applications

1.6. Scope and Limitations

This work is expected to be developed under some constraints and limitations such as:

- We will deal with only OOI which have a close boundary from cluttered background images.
- We will determine the ROI and the parameters before implementations.
- Our work depended on greedy snake algorithm, can't use other ACM methods for cluttered background and boundary concavities.

1.7. Research Methodology

The methodology of research had been followed in order to complete this research and achieve our goal presented as:

1) Literature Review

Firstly, a review had been done for current techniques used for handling the background cluttered and boundary concavities .

2) Study ACM methods

Hard study for ACM methods and it's types and consequences on the performance of object segmentation method.

3) Development our proposed solution

The proposed algorithm had handled background cluttered and boundary concavities problems efficiently.

4) Implementing the proposed algorithm

The proposed algorithm had been implemented and tested in Matlap.

5) Evaluating and comparing results

The evaluation of the system had been done using two metrics: errors and processing time. These two metrics had been used to compare the proposed algorithm with greedy snake.

In this chapter detailed information about object segmentation was presented. In addition, clutter background and boundary concavities were defined and the problem which affects the performance of extracting OOI had been clarified.

1.8. Thesis Organization

The remainder of the report is organized as follows. In the next chapter a related work , of a selection of papers dealing with active contours, is given. Thereafter the " proposed method " will be analyzed and explained in detail. This will be followed up by a chapter containing the results that were obtained when running the two – phase snake algorithm on the test images, with evaluation of the results. The final chapter will be the conclusion. In the appendix the reader will find the list of references.

Chapter (2):

Related Works

2. CHAPTER 2: Related Work

In this chapter, we present more details on the most common segmentation methods that have been proposed based on the classical snake model, and discuss their limitations with respect to the problems of background clutter and boundary concavities.

2.1. Object Segmentation Methods

A class of object segmentation methods that targets the boundary of the object is the edge-based Active Contour Methods (ACM, also known as "snake"), and the level-set method.

2.1.1. The Active Contour Model (ACM)

Active Contours Model (Snake), is a method depends on active curves or surfaces, and has been widely used in image segmentation field. The ACM separate an object from its surroundings by locating the object boundaries in the image. The ACM use a dynamically evolving curve to minimize an energy functional that presents a non-Euclidean length of the segmenting contour. The minimum occurs when the curve synchronize with points of high gradient in the image. Assuming that the object edges are characterized by relatively high intensity variations, the active contour then becomes stable when it reaches the object's boundary.

Active contours were pioneered with the classical snakes [3], followed by Geodesic Active Contour (GCA) [2], greedy snake [22], the balloon snake [23], and the Gradient Vector Flow (GVF) [4].

- **Traditional Snake**

Now we will talk about the traditional snake which is the basic element in our research.

In our research we developed snakes method for solving the problems. A snake is a list of points which are typically initialized outside the object of interest (OOI). Through an "energy minimization" process, the points iteratively converge towards the center stopping eventually at the OOI's boundary. In [15], all edges are the same because the snake can't distinguish between edges of the OOI and those of other image components, and thus the snake will converge on the first edge in it is way.

The active contour model is defined by an energy function, which is a weighted combination of *internal* and *external forces*. The internal forces emanate from the shape of the snake, while the external forces come from the image and/or from a higher-level image understanding process.

In Figure 2.1, Snake introduced by Kass [3], the idea started from making the initial closed curve converge to the desired object by minimizing the energy functional. The name “snake” is the reason of the appearance of the parametric curve, which changes during the iterative process.

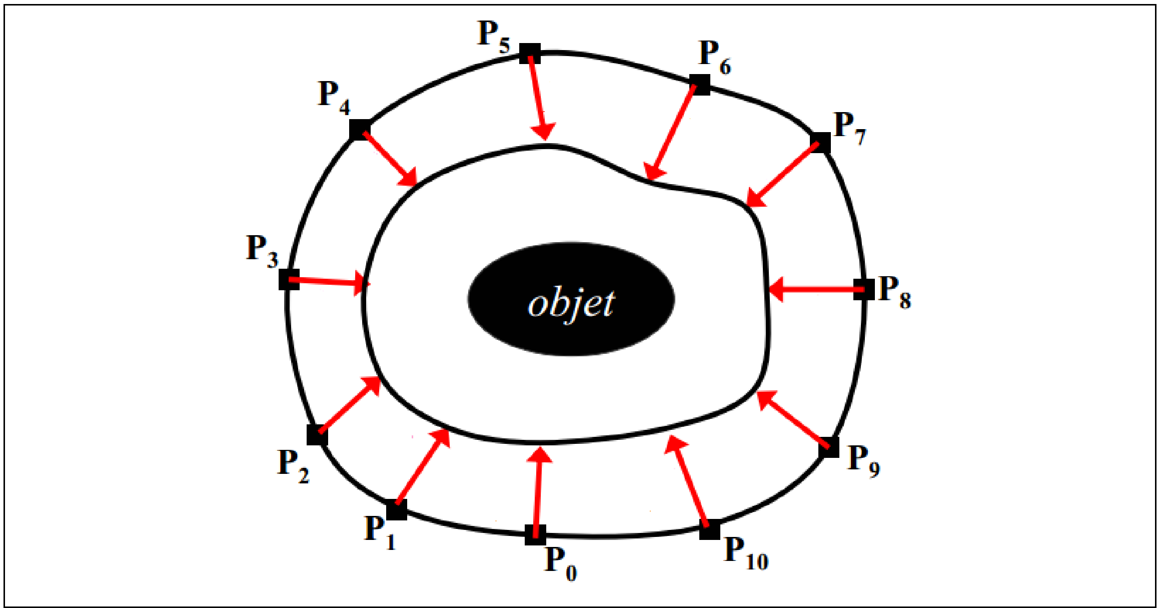


Figure 2.1: Active Contour Model (snakes).

The parametric form of a curve, which is in a vector form, is specified as $v(s) = (x(s), y(s))$ where $x(s), y(s)$ are x, y co-ordinates along the contour and s is from [0:1].

$$v(s) = \binom{x(s)}{y(s)} \qquad 2.1$$

Snake idea firstly depends on the continuous domain, and then it was transformed into a discrete model for implementation, the solution is found using techniques of variational calculus.

Firstly (snake in the continuous domain) [3],[24]: The energy functional is commonly defined using several additive terms, such as the following:

$$E_{snake} = \int_0^1 E_{int}(v(s)) + E_{ext}(v(s))ds \qquad 2.2$$

The energy functional is a combination of internal and external energy.

The internal energy E_{int} $(v(s))$ constrains shape of curve or encourages smoothness and the external energyE_{ext} $(v(s))$encourages matching to suitable image features as strong edges which, consists of both the image energy $E_{img}(v(s))$ and the external constraint forces E_{con} $(v(s))$, To simplify the analysis we will disregard the external constraint forces, so we get

$$E_{snake} = \int_0^1 E_{int}(v(s)) + E_{img}(v(s))ds \qquad 2.3$$

The snake will try to position itself in areas of low energy, often the snake should be attracted toward different image features as edges in the image. Because, the image has noise which obstructs the snake moving, the image can be smoothing to reduce the noise, the image can be convolved with a Gaussian kernel before computing the gradients that increase the capture range of the snake.

One possible way to extract edges is with Sobel filter, this filter combines a Gaussian kernel G with the derivative of the image$I$, as the image energy term is $E_{img} = I(x, y)$, where I is the image function. Using Gaussian kernelG_σ, This gives the following image energy term

$$E_{img} = -\nabla \| G_\sigma(x, y) * I(x, y) \| \qquad 2.4$$

Where $G_\sigma(x, y)$ is a two dimensional Gaussian with standard deviation σ. When strong edges in the image are blurred by the Gaussian the corresponding gradient is also smoothed which results in the snake coming under the influence of the gradient forces from a greater distance, hereby increasing the capture range of the snake. The negative sign reverses the energy so that sharp edges are mapped to areas of low energy.

In addition to the image energy, the snake is influenced by it's own internal energy, the internal energy of the snake is the component of the behavior function that describe the physical properties of our contour like smoothness or continuity and curvature as follows:

$$E_{int} = ½(\alpha(s)\|v_s(s)\|^2 + \beta(s)\|v_{ss}(s)\|^2) \qquad 2.5$$

The first order term $\|v_s(s)\|^2$ measures the continuity (elasticity) or smoothness so it's a function of the first derivative of our contour and, it is controlled by the coefficients $\alpha(s)$, while the second order $\|v_{ss}(s)\|^2$ measures the curvature energy, and is function of the second derivative of our contour, it is controlled by the coefficients $\beta(s)$.

The more the snake is stretched at a point $v(s)$ the greater the magnitude of the first order term, whereas the magnitude of the second order term will be greater in places where the curvature of the curve is high. It should be noted that if the snake is not under the influence of any image energy, and only moves to minimize its own internal energy. Then, for a closed curve, it will take the form of a circle that keeps shrinking and for the open curve the snake will position itself to form a straight line that's also shrinks.

Continuity force, which is the first term in the internal energy of the snake, is defined as the magnitude of the first derivative of the parametric curve. The first derivative of a parametric curve is given by

$$v_s(s) = \begin{pmatrix} x_s(s) \\ y_s(s) \end{pmatrix} \qquad 2.6$$

This derivative is the tangent vector to the curve at any point, and the direction of the tangent vector is the same direction of that curve at the point, see Figure 2.2. The magnitude/length of the tangent vector indicates the speed of the curve at the point. The speed of a parametric curve indicates the distance covered on the curve when s is incremented in equal steps. Therefore when the internal energy of the snake is being minimized, the speed of the curve will be diminished. This means that the continuity force helps keep the curve from deforming excessively. Thus the shrinking effect allows us to place the snake around the object of interest. However it might be necessary to adjust the parameter in order to prevent the continuity force from overcoming the image energy completely, as this would result in the snake shrinking into a single point.

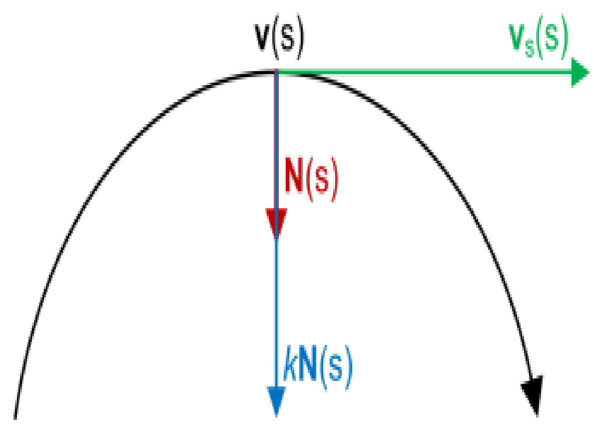


Figure 2.2: Illustration of a parametric curve v and some of the intrinsic vectors

The green vector is the tangent vector, the red vector is the unit principal normal vector and the blue vector shows the curvature multiplied with the unit principal normal vector.

The curvature force, which is the second derivative of the parametric curve $\|v_{ss}(s)\|$ is used to measure how much the snake curves at a point. Usually the curvature k is given by the second derivative of the curve with regards to the arc length .

$$v_u(s) = kN \qquad 2.7$$

where N is the unit principal normal vector which is illustrated in Figure 2.2. The magnitude of the curvature is given by

$$\|v_u(s)\| = \|kN\| = |k|\|N\| = |k|\,1 = |k| \qquad 2.8$$

Kass [3] assumed that the snake curve is parameterized by arc length, so $v_u(s) = v_{ss}(s)$, However we also have to adjust the parameter β that controls the influence of the curvature energy.

The following equation explains snake energy which has a combination of internal and external energy which gives the minimum energy.

$$E_{snake} = \int_0^1 ½(\alpha(s)\|v_s(S)\|^2 + \beta(s)\|v_{ss}(\mathbf{S})\|^2) + E_{img}(v(s))ds \qquad 2.9$$

An implementation of the traditional snake through a discrete formulation will explain in greedy snake.

- **Greedy Snake**

The greedy snake has a discrete nature, it depends on active contour model which, is an energy minimizing model where the energy of a spline is minimized under the influence of image forces and external constraints. It is important to note that the primary characteristic of the greedy snake is that it computes the movement of each snake point by looking at the neighborhood of pixels around the snake point and then moving the snake point to the position in the neighborhood which minimizes the energy term. The energy functional for the contour is defined as follows.

$$E *_{snake} = \int_0^1 E_{snake}(v(s))\, ds \qquad 2.10$$

$$E *_{snake} = \int_0^1 E_{int}\big(v(s)\big) + E_{img}\big(v(s)\big) + E_{con}\big(v(s)\big)\, ds \qquad 2.11$$

The image energy of the greedy snake is calculated as one possible way to extract edges with a Sobel filter, this filter combines a Gaussian kernel G with the derivative of the image I as follow, it is almost the same energy term as in the Kass. Snake, except that there is missisxzng a minus in front of the term.

$$E_{img} = \|\nabla\, [\, G_\sigma(x,y) * I(x,y)]\ \|^2 \qquad 2.12$$

The greedy snake algorithm works by computed energy functions on neighborhood around each snake point and then move to the position with the lowest energy.

Therefore the image energy in the neighborhood of the snake control point $v(s_i)$ has to be normalized in a manner which assigns large negative values to pixels with high gradient values, while assigning lower negative values to pixels with a lower gradient value, the gradient magnitudes are all in the interval [0,255], the normalization of the neighborhood is calculated as

$$Locationn(x,y) = \frac{(\min - mag\ (x,y))}{(\max - min\ )} \qquad 2.13$$

where $(min), (\ max)$ is the minimum and maximum gradient value in the neighborhood, and $mag(x, y)$ is the gradient magnitude of the current point. This normalization sets the highest gradient magnitude in the neighborhood to -1 and the lowest to 0, but if the neighborhood is nearly uniform in gradient magnitude we get large difference in the normalized values. So if all the gradient magnitude values are in the interval [46, 49] then the normalized values would be 0, -0.33, -0.66 and -1, which would suggest a strong edge even when there is none. To solve this problem the minimum value min is set to $max\ - 5$ if $max - min\ <\ 5$ [25].

The internal energy of the spline E_{int}, is divided in two terms E_{cont} and E_{curv}.

The continuity term of the greedy snake as it was clear in the Kass et al. the continuity term $\|v_s(s)\|^2$ has the effect of causing the curve to shrink upon itself. The greedy snake used a different way of calculating the continuity term. Thus reducing the shrinking effect and also making sure that the snake control points does not bunch up in places with high image energy. As its accounts for the spline continuity and should grow larger as the spline is stretched.

The continuity term in the greedy snake is calculated in the neighborhood of each snake control point as

$$\bar{d} - \|v(s_i) - v\ (s_{i-1})\| = \ \bar{d} - \ \sqrt{\sqrt{(x(s_i) - x\ (s_i) - 1)^2} + \ (y(s_i) - y\ (s_i) - 1)^2} \quad 2.14$$

where d is the average distance between all the points in the snake. After term has been calculated for each pixel in the neighborhood of a snake control point, the neighborhood is normalized as all the values are divided by the largest value in the neighborhood, which means that it only contains continuity term values between [0, 1].

The minimum energy will be achieved when

$$\bar{d} - \|v(s_i) - v\ (s_{i-1})\| = \ 0 \qquad 2.15$$

which will be true for points where the average distance equals the distance between the current and previous point on the snake. This new continuity term will therefore encourage the snake control points to be evenly spaced along the curve keeping the curve parameterized by arc length.

Finding a suitable value for this parameter $\alpha\ (s_i)$ is equally important in the greedy snake as in the Kass et al. If the parameter is set too high the snake will not really be able to evolve to fit the contour of an object because the snake curve will not change its length at all, see Figure 2.3.

In Williams and M. Shah [23], evaluate a range of different ways of calculating the curvature for a discrete parametric curve such as the greedy snake curve. Its accounts for the spline curvature.

$$\|v(si+1) - 2v(si) + v(si-1)\|^2 = 0 \qquad 2.16$$

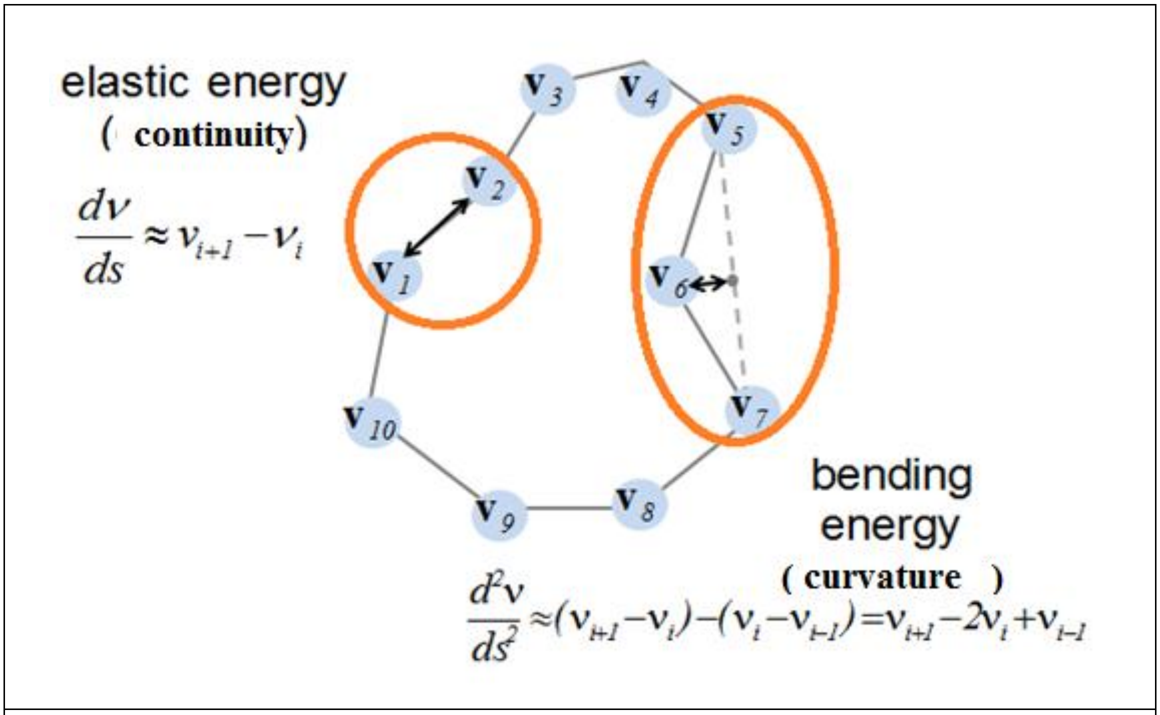


Figure 2.3: Elastic and curvature energy in greedy snake.

However the control points in the greedy snake a more likely to be evenly spaced than for the Kass et al. snake since the continuity term encourages even spacing of the control points. The influence of the curvature is controlled by the parameter β as in the Kass et al. Once the curvature has been calculated for each point in the neighborhood of the current snake control point, the values are normalized by dividing with the largest value which is in the range [0, 1].

For each point/pixel in the neighborhood of a snake control point $v(s_i)$ the three energy terms are calculated. Then the algorithm sums the energy terms to get the combined energy

$$E_{comb(x,y)} = \alpha(s_i)E_{ela}(x,y) + \beta(s_i)E_{curv}(x,y) + \gamma(s_i)\, E_{img}(x,y) \qquad 2.17$$

Where $E_{ela}(x,y)$ is the elasticity energy, $E_{curv}(x,y)$ is the curvature energy, $E_{img}(x,y)$ is the image energy and *(x,y)* are the indices to the points in the neighborhood. As we see that the greedy snake algorithm also uses a parameter to control the influence of the image energy.

Once the combined energy has been calculated for each neighborhood, the greedy algorithm choice and moves the snake control point to the position that has the minimum combined energy, so the name of the algorithm is derived from its behavior. This behavior is illustrated in Figure 2.4, the values in the squares represent the combined energy. Also pictured is the snake control point v(s_i) and the point before and after it. The red arrow shows to which point in the neighborhood the snake control point will move.

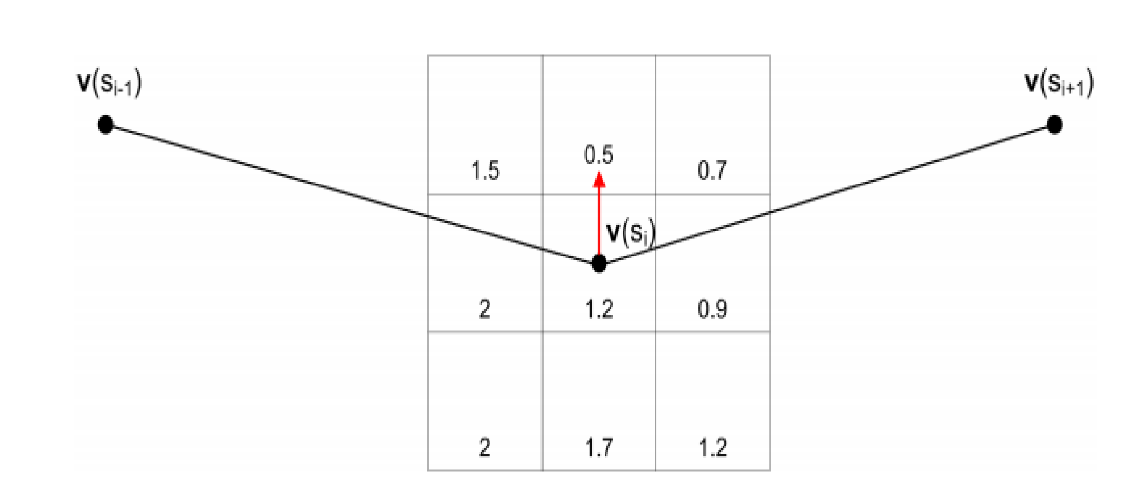


Figure 2.4: Illustrating the movement of a snake control point.

After all the control points along the snake have been moved to a new position the curvature is calculated a second time only for each control point along the snake and not for all the points in the neighborhood. That is to locate places where the curvature is high and then relax the parameter $\beta(s_i)$ for this control point i.e. setting $\beta(s_i) = 0$. Computing the curvature in the second time follows the following equation:

$$\left\| \frac{u_i}{\|u_i\|} - \frac{u_i+1}{\|u_i+1\|} \right\|^2 \qquad 2.18$$

where $ui = [x(s_i) - x(s_{i-1}), y(s_i) - y(s_{i-1})]$ and $ui + 1 = [x(s_{i+1}) - x(s_i), y(s_{i+1}) - y(s_i)]$

This equation gives a more accurate estimation of the curvature since we normalize by the magnitude of the vectors.

Once the new curvature has been calculated for all the snake control points the parameter is relaxed for control points where the following conditions hold true. First, the curvature of the control point $v(s_i)$ has to be larger than the curvature for its two neighbors $v(s_{i-1})$ and $v(s_{i+1})$. Second, the curvature has to be larger than a preset threshold value, Th_{curv}. Finally, the magnitude of the gradient at the control point also has to be above a certain threshold, Th_{grad}. If all these conditions are true then the control point is relaxed which means the value $\beta_{relax}(s_i)$ is set to 0.

The final step in the iteration of the greedy snake algorithm consists of checking whether the number of points that are still moving, $\beta_{relax}(s_i) <> 0$, in the iteration is below a threshold, Th_{mov}. This is used as a stopping criterion as the snake is *presumed* to have reached minimum energy when *most of* the control points have stopped moving.

- **An Active Contour Balloon Model**

The active contour model that Kass et al. introduced was developed to anew Active Contour Models calls the “Balloons" by Cohen [23]. The snake model balloons works on the same principles as the Kass et al. snake, but where the Kass et al. snake would shrink when not under the influence of image forces, the Cohen snake expands, such the snake bears some resemblance to a balloon being inflated in 2D, the expansive behavior is achieved by altering the values of $f_x(x, y)$ and $f_y(x, y)$ in equation:

$$f_x(x,y) = k_1\, n(s) - k\, \frac{\nabla p_x}{\|\nabla p_x\|} \;, f_y(x,y) = k_1\, n(s) - k\, \frac{\nabla p_y}{\|\nabla p_y\|} \qquad 2.19$$

where $n(s)$ is the unit principal normal vector to the curve at point $v(s)$ and $k1$ is the amplitude of this force, we have $\nabla p_x = \delta E_{ext}/\delta x$ *and* $\nabla p_y = \delta E_{ext}/\delta y$ while the magnitude of k determines the influence of the normalized external forces.

The changes to the original Kass et al. snake that the balloon snake introduces makes it possible to find the contours of an object by placing the initial snake inside the object instead of outside, while also providing more best results.

- **Gradient Vector Flow (GVF) Snake**

Gradient Vector Flow is one of the snake models that have been developed by Xu and Prince [4]. The Gradient Vector Flow snake was developed in order to increase the capture range and improve the snake's ability to move into boundary concavities. We know that, the capture range of the traditional snake Kass et al, is generally limited to the vicinity of the desired contour. Also, the traditional snake has problems with moving into concavity boundary.

The Gradient Vector Flow snake handles these problems by introducing a new external force. This new external force is a dense vector field derived from the image by minimizing energy functional in a variational framework. The minimization is achieved by solving a pair of decoupled linear partial differential equations which diffuses the gradient vectors of a gray-level or binary edge map computed from the image. This leads to the vector field being slowly varying in homogeneous regions, and at the same time being nearly equal to the gradient of the edge map in regions where the gradient of the edge map is large.

2.1.2. Level Set

The level-set based contour representations [7], have become a popular framework for image segmentation. They permit topological changes in the evolving contour and are also used to exploit various low level image properties such as edge information.

The level set method originally used as numerical technique for tracking interfaces and shapes. In the level set method, contours or surfaces are represented as the zero level set of a higher dimensional function, usually called a level set function, it is able to represent contours/surfaces with complex topology and change their topology in a natural way. Using the level set representation; the image segmentation problem can be formulated and solved in a principled way based on well-established mathematical theories, including calculus of variations and partial differential equations .

An advantage of the level set method is that numerical computations involving curves and surfaces can be performed on a fixed Cartesian grid without having to parameterize these objects.

2.2. Related Work

We divide the methods to: Snake-based or Non snake-based

2.2.1. Snake-based

Snake-based depends on active contour method for solving the problem such as:

Kass, Witkin, and Terzopoulos [3], have proposed Active contours or snakes are computer generated curves that move within the image to find object boundaries under the influence of internal and external forces.

The energy function is defined as

$$E_{snake} = \int_0^1 E_{snake}(v(s))\, ds \qquad 2.20$$

$$E_{snake} = \int_0^1 E_{int}\big(v(s)\big) + E_{img}\big(v(s)\big) + E_{con}\big(v(s)\big)\, ds \qquad 2.11$$

$E_{int}\big(v(s)\big)$: represents the internal energy of the spline due to bending,

$E_{img}\big(v(s)\big)$: represents the image forces,

$E_{con}\big(v(s)\big)$: represents the external constraint force

The sum of the image forces E_{img} and the external constraint forces E_{con} is also simply known as the external snake forces, denoted by E_{ext}. The internal energy of the snake

measures the continuity (elasticity) or smoothness of the first derivative of the contour and measures the curvature energy, which is the second derivative of the contour.

First Snake is placed near the contour of Region Of Interest (ROI), then during an iterative process due to various internal and external forces within the image, the Snake is attracted towards the target. These forces control the shape and location of the snake within the image.

An energy function is constructed which consist of internal and external forces to measure the appropriateness of the Contour of ROI, Minimize the energy function. (integral), which represents active contour's total energy, The internal forces are responsible for smoothness while the external forces guide the contours towards the contour of ROI.

Shortcoming of traditional snake is that, it requires user interaction, which consists of determining the curve around the detected object, the energy function often converge to minimum local energy, so snake should be placed usually near the boundary of ROI, such no external force acts on points which are far away from the boundary ,and convergence is dependent on initial position, so snake can't detect the OOI boundary from clutter background without user interaction.

Traditional snake algorithm is particularly sensitive to noise. More sensitive to the choice of its parameters and adaptively adjusts the parameters in an extremely complex process. The computational complexity of the algorithm is high. Snake fails to detect concave boundaries; such external force can't pull control points into boundary concavity. These issues can in principle, be solved by very high-level computations.

Williams and Shah [22], have introduced the greedy snake algorithm, their approach differs from the original Kass et al. This entails computing the movement of each snake point individually on the discrete indices of the image. The movement of each snake point is computed by looking at the neighborhood of pixels around the snake point and then moving the snake point to the position in the neighborhood which minimizes the energy term.

Williams and Shah also investigate efficient ways to approximate the curvature term when dealing with discrete curves. This leads to a new way of computing the curvature term, which is particularly appropriate for the greedy snake algorithm.

Laurent D.Cohen[23], have presented new snake it's behaves like a balloon which is blown up. Cohen snake would not shrink under the influence of image but it's expands. The expansion of the snake bears some resemblance to a balloon being inflated in 2D,

Snake balloon when it passes by edges; it is stopped if the contour is strong, or passes through if the contour is too weak. Thus, the initial snake needs not to be close to the solution (object) to converge. This approach modifies the definition external forces (derived from gradient of the image) presented in traditional snake (Kass *et al*).

A balloon model can solve some of the problems encountered with the snake model, but it cannot deal with boundary concavities problem.

Chenyang and Jerry [15], have developed a new external force for active contours, which they call gradient vector flow (GVF) and might form the basis for a new geometric snake, the gradient vector flow snake was developed in order to increase the capture range and improve the snakes ability to move into boundary concavities. The capture range of the traditional snake is generally limited to the vicinity of the desired contour, also the traditional snake has problems with moving into concave regions as an U-shaped object.

(GVF) are dense vector fields derived from images by minimizing an energy functional in a variational framework. The minimization is achieved by solving a pair of decoupled linear partial differential equations which diffuses the gradient vectors of a gray-level or binary edge map computed from the image.

Chenyang and Jerry [5], have developed GVF snake which is distinguished from the previous snake formulations in that its external forces cannot be written as the negative gradient of a potential function. Because of this, it cannot be formulated using the standard energy minimization framework; instead, it is specified directly from a force balance condition. In[4],[5], GVF snake has advantages like, it's insensitivity to initialization and its ability to move into boundary concavities, also there have some weakness as, the background clutter problem remains a challenge which cannot be treated through modifications to the model but rather requires ad hoc treatment , also very sensitive to parameters , slow and it is computationally expensive .

Although the GVF-snake has large capture range and considerable ability to handle boundary concavities, it is difficult to realize accurate segmentation when detecting complex shape objects with deep concavities, so it is considered as poor convergence to boundary concavities.

Zhang, Li and Bai [25], have proposed a novel improved scheme was based on the GVF-snake which has a large capture range and can deal with boundary concavities. However, when interesting object has deep concavities, traditional GVF-snake algorithm can't converge to true boundaries exactly. So they have introduced dynamic balloon force and tangential force to strengthen the static GVF force, and it is found that the capability of capturing concave edges enhanced effectively, such the balloon force is used to increase capture range and converging speed of snake, and the tangential force is used to make the snake converge to boundary concavities better.

Xu and Prince [26], have proposed GGVF snakes which have the ability of attracting the active contour toward object boundary from a sufficiently large distance and the ability of moving the contour into object boundary concavities. It was introduced recently to address problems in GVF. The external force fields derived from this new generalized GVF (GGVF) improve active contour convergence into long, thin boundary indentations, while maintaining other desirable properties of GVF, such as the extended capture range. The original GVF is a special case of GGVF. However, these parametric snake models still cannot automatically handle topological changes.

Shin, Alattar and Jang [27], [28], [29], have presented a snake-based scheme for efficiently detecting contours of objects with boundary concavities. The proposed method is composed of two steps. First, the object's boundary is detected using the proposed snake model. Second, snake points are optimized by inserting new points and deleting unnecessary points to better describe the object's boundary. The proposed algorithm can successfully extract objects with boundary concavities, and is insensitive to the number of initial snake points.

Shortcoming of snake-based method is solving the background clutter and boundary concavities efficiently.

2.2.2. Non-snake-based

Non Snake-based, which use methods different to active contour method for solving the problem such as:

- **Shape-based**

Ravikanth, James, and Baba [7], have used priori shape models to identify the object of interest (OOI) and its edges and therefore classify all other edges as background clutter. They adopt level set techniques to the problem of shape recovery. To isolate a shape from its background, they first consider a closed, nonintersecting, initial hyper surface placed inside (or outside) it. This hyper surface is then made to flow along it's gradient field . While such methods may work for specific applications where prior knowledge about the object of interest OOI is available, obviously they cannot be used outside of their domain, because not always possible to specify the topology of an object prior to its recovery, however the clutter background problem hadn't been solved.

Anucha, Wichian[30], have identified the ROI in a natural scene as a complex task because the content of natural images consists of the multiple non-uniform sub-regions. They have presented a novel Region of Interest (ROI) detection method to minimize the ROI in the images automatically by applying the geometric active contours at forces the variational level set function to be close to object boundaries. They have compared the efficiency of the proposed method with the method using the human segmentation of the images.

Riklin, Kiryati, Sochen [31], recently published a method allowing for a more general projective transformation of the object's shape. The image may contain a group of objects with similar shapes that can be transformed to each other by the allowed transformation (e.g., scale or rotation). In such case the prior shape encounters an ambiguity and therefore is not capable of discriminating between the objects of the group. Moreover, scenes with high level of clutter can have many local minima where the evolved contour may be trapped. In these cases there is a need for higher prior support in the segmentation process than just a shape model.

Ben, Aiger [2], have presented a new object segmentation method for segment an object in a very cluttered environment ,it is based on geodesic active contours (GAC) which combined shape and appearance priors, the appearance similarity measure is based on intensity differences, that means improve object segmentation in cluttered scenes and occlusions. They add a new prior, based on the appearance of the object to be segmented. This method enables the appearance pattern to be incorporated within the geodesic active contour framework with shape priors, seeking for the object whose boundaries lie on high image gradients and that fits the shape and appearance of a reference model.

Chan, Vese[23], have proposed a new model for active contours to detect objects in a given image, based on techniques of curve evolution, Mumford–Shah functional model uses the global information of the image as the stopping criterion for segmentation and level sets which is represent the curves or surfaces as the zero level set of a higher dimensional hyper surface. The problem becomes a “mean-curvature flow” from level sets, like evolving the active contour, which will stop on the desired boundary. However, the stopping term does not depend on the gradient of the image, as in the classical active contour models, but is instead related to a particular segmentation of the image. They have given a numerical algorithm using finite differences. This technique provides more accurate numerical implementations, also handle topological change very easily. The drawback of these methods appears when the contrast of a desired object or part of it, is low with respect to the background. The segmentation outcome then labels partially or the whole object as background, so it cannot solve background clutter.

Alban, Pierre[33], they present a new way of constraining the evolution of an active contour with respect to a set of fixed reference shapes. This approach is based on a description of shapes by the Legendre moments computed from their characteristic function. This provides a region-based representation that can handle arbitrary shape topologies. the new shape prior is based on a distance, in terms of descriptors, between the evolving curve and the reference shapes. Minimizing the corresponding shape energy leads to a geometric flow that does not rely on any particular representation of the contour and can be implemented with any contour evolution algorithm. Then introduce there prior into a two-class segmentation functional, showing its benefits on segmentation results in presence of severe occlusions and clutter.

As we see there is a shortcoming of non-snake-based method is solving the background clutter and boundary concavities efficiently.

- **Other methods**

Liyuan Li, et al. [34], have presented their work on foreground object detection through foreground and background classification under Bayesian framework. The object of interest called foreground often has homogeneous (constant or smooth) statistics. However, the remainder of the image (the background) is certainly not well approximated by a constant gray value. As a result, detecting the boundaries a seemingly easy task is hampered by the structure of the background, which ends up influencing the boundary more than the characteristics of the OOI, so this method cannot detect the OOI from clutter background, it's fails to deal with background clutter problem.

Xintong, Xiaohan and Chen [35], have proposed a foreground segmentation algorithm that does foreground extraction under different scales and refines the result by matting , This method can treating challenging images such cluttered background . First, the input image is filtered and resampled to 5 different resolutions. Then each of them is segmented by adaptive figure-ground classification and the best segmentation is automatically selected by an evaluation score that maximizes the difference between foreground and background. This segmentation is up sampled to the original size, and a corresponding trimap is built. Closed-form matting is employed to label the boundary region, and the result is refined by a final figure-ground classification.

In general, ACM model can be modified to improve its performance against the concavity problem and background clutter, but it remains a challenge, which can't be treated through modifications to the model.

2.3. Summary

As shown in this chapter most algorithms have limited capability in overcoming the problems of background clutter and boundary concavities. Even those who work on object segmentation problem, they failed to address the performance improvements especially in the cluttered background and boundary concavities.

This led us to think about developing a two- phase snake method that can work efficiently under most conditions .

Chapter (3):

Proposed Method

3. CHAPTER 3: Proposed Method

This chapter starts with a general introduction of what we add to solve the last challenges "Clutter Background and Boundary Concavity" using an active contour and how it works.
Our method follows the same approach of greedy snake, which aims to locate an object contour, under the influence of internal and external energy.

3.1. Method Characteristics

In our work we achieve our objectives in terms of the following characteristics before explaining the new method steps:

- **Two-phase approach**

Our approach uses two-phases, In phase1 a normal greedy snake is initialized at the borders of an ROI around the OOI. The snake starts to locate the OOI, under the influence of internal and external energy. Due to background clutter and boundary concavities, it is very unlikely that phase1 snake converge on the OOI's contour. Therefore, in our method we introduce a second phase in which the snake shifts to a new position calculated based on its final position of phase1. A center location amid the stable points of phase1 is calculated and each stable point is shifted to the middle distance between its current location and the center. Phase2 takes on from there, and the snake resumes operating again as a greedy snake, and moves toward the OOI's contour under the influence of the same energy forces.

The main advantage achieved from this proposed improved technique is the accuracy of determining the OOI which results in more accuracy in the use of the two- phase snake by reducing the influence of background clutter and boundary concavities.

- **Scale space continuation**

The scale space continuation technique is presented in the original paper by Kass et al. [3] as a way to improve the snake's ability to find the desired object contour even under the influence of noise, i.e. cluttered background. What's different in our method is that we apply this technique twice, in the two phases we proposed. That improves the snake's ability in locating the object contour from cluttered background. More details on this are given in Phase 1 subsection.

- **Handling boundary concavities**

Our method keeps the points in the snake equally spaced even at locations where the snake curve bends. When this requirement is enforced in the two-phase approach, its effect in handling boundary concavities becomes even stronger.

- **Pure object contour**

Active contours are used in the domain of image processing to locate the contour of an object. Trying to locate an object contour purely by running a low level image processing task such as Canny edge detection is not particularly successful. Often the edge is not continuous, i.e. there might be holes along the edge, and spurious edges can be presented because of noise, so we used Sobel edge detection [36], it is a discrete differentiation operator, and is based on convolving the image with a small, separable, and integer valued filter in horizontal and vertical direction and is therefore relatively inexpensive in terms of computations. On the other hand, the gradient approximation that it produces is relatively crude, in particular for high frequency variations in the image [36].

3.2. Method Steps

Our new method is depicted as a flow chart in Figure 3.1, It consists of preprocessing, which have input image, determines the ROI, edge smoothing by Gaussian, and edge detection using "Soble " filtering , then phase1 which is a greedy snake, and phase2 which implemented as a greedy snake after shifting the control points position .

Note that, the preprocessing (Gaussian, Soble filtering) is determined at start of phase2.

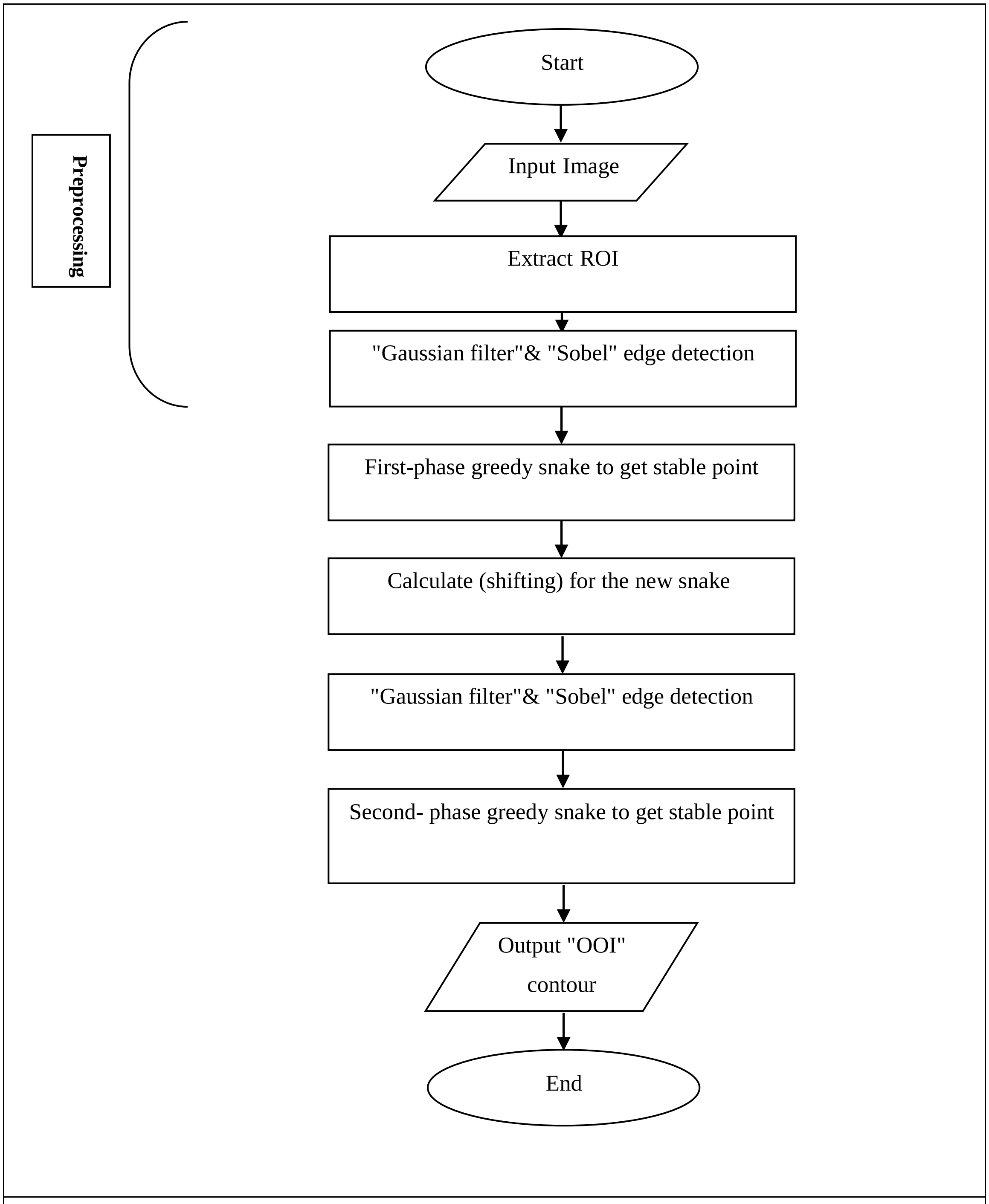


Figure 3.1: Steps of the new proposed method "A Two- Phase Snake".

Now: We'll explain ,in details, the consequences steps to resolve the problems which were mentioned in our research. The following steps explain this new mechanism.

3.2.1. Pre-Processing

- ***Defining the ROI***

After loading an image, the user must define an ROI around the OOI manually. The user choose Snake Circle which drone by the program or the user can either draw a circle by determines the center and the radius of the ROI, or place the control points one by one around the OOI, see figure (3.2). Each of these choices have its own advantages and disadvantages. When drawing a circle, the method will automatically initialize the snake points evenly spaced at the perimeter of the circle with the desired density. This relieves the user from having to set each snake point manually. However, in this manner some points will be closer to the actual object contour while others will be much far away. This disadvantage is eliminated when the user manually sets the points, however, the initialization process becomes cumbersome and time consuming. If the sets point is little, that influence of the snake operating .

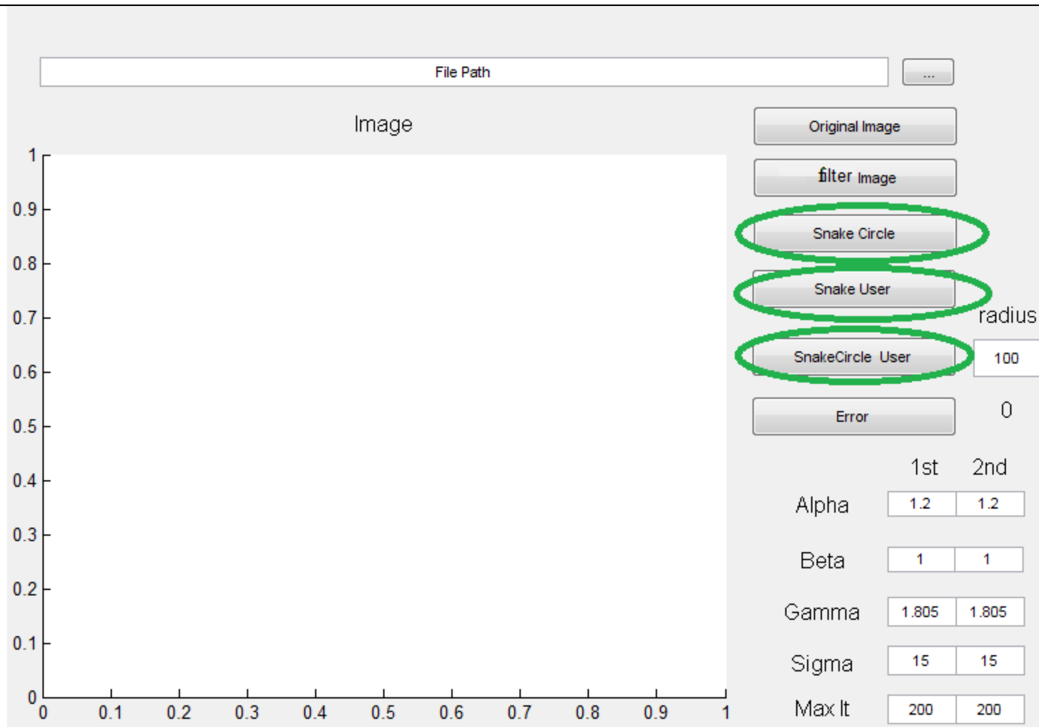


Figure 3.2: GUI testing .

So, to overcome this, we will have a function that will subdivide in equal length interval the line connecting two dots. By doing so the user will only have to define approximate shape around the object with a ten of point and the function will provide us with several subdivisions on each segment to have smooth lines.

To do subdivision in equal length, function first computes the average distance between all the snake control points add by the user. Then we iterates through all the snake control points while removing points in parts of the snake where the points are close together compared to the average distance, also inserts new points in parts of the snake where points are far apart compared to the average distance. When a new point is inserted, it is inserted in the middle of the line connecting the two points that are far apart.

- ***Edge Detection***

We used edge map, which is generated based on edge strength from the original image that helps in the solution. For approximating the directional gradients G_x and G_y of the image function $I(x, y)$, we use convolution with Sobel filters[36],[37] .

The operator uses two 3×3 kernels which are convolved with the original image $\boldsymbol{A}$ to calculate approximations of the derivatives, one for horizontal changes, and one for vertical.

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A \quad and \; G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A \qquad 3.1$$

Since the Sobel kernels can be decomposed as the products of an averaging and a differentiation kernel, they compute the gradient with smoothing. For example, G_x can be written as

$$\begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & +1 \end{bmatrix} \qquad 3.2$$

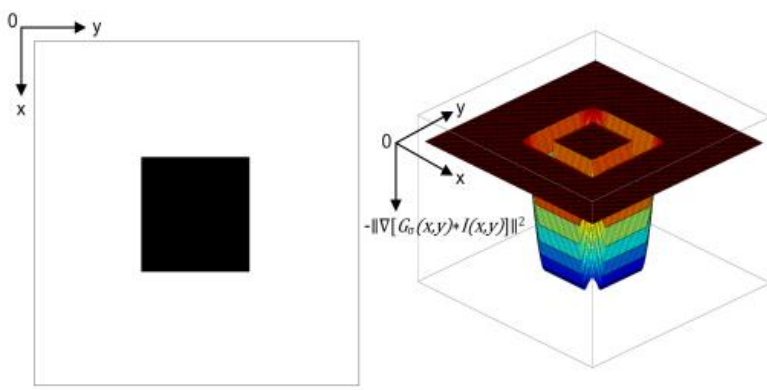
The gradient magnitude used in the image energy terms is then computed as :

$$G = \sqrt{G_x^2 + G_y^2} \qquad 3.3$$

Active contours try to improve on this by imposing desirable properties such as continuity and smoothness to the contour of the object. This means that the active contour approach adds a certain degree of prior knowledge for dealing with the problem of finding the object contour.

- ***Edge Smoothing***

Since all edge detection results are easily affected by image noise, it is essential to filter out the noise to prevent false detection caused by noise. In [38], to smooth the image, a Gaussian filter is applied to convolve with the image before computing the gradients. This step will slightly smooth the image to reduce the effects of obvious noise on the edge detector and increase the capture range of the snake. From the image energy term where $G_\sigma(x,y)$is a two dimensional Gaussian with standard deviation σ. When strong edges in the image are blurred by the Gaussian the corresponding gradient is also smoothed which results in the snake coming under the influence of the gradient forces from a greater distance, hereby increasing the capture range of the snake. In figure 3.3 the concept of the image energy is illustrated.



(a) (b)

Figure 3.3: Illustration of the image energy

Figure 3.3. (a) The original image, showing a black square on a white background. (b) The image energy derived from the image energy, shown as a 3D surface. also explain how the capture range is increased when first applying Gaussian convolution to the image, since the image energy would be focused only in the direct vicinity of the black square.

3.2.2. Phase I "greedy snake"

The first phase executes as a typical greedy snake algorithm. Note that, a set of parameters in snake energy : α is the continuity energy parameter , β is the curvature energy parameter , γ is the image energy parameter, σ is Gaussian parameter , and $maxItr$ must be initialized first. The greedy snake algorithm is briefly explained in "chapter 1" was used it to determine the first boundary of OOI, it started with calculated the image energy as

$$E_{img} = \|\nabla \left[G_{\sigma}(x, y) * I(x, y) \right] \|^2 \qquad 3.4$$

Based on scale space continuation, during this phase image energy is regenerated once every certain number of iterations with a decreased Gaussian sigma G_{σ}. Typically sigma starts initially at 15 and decreases at a rate of 4 every 15 iterations. These parameters can be adjusted according to the case depending on noise density. The movement of each snake point is computed by looking at the neighborhood of pixels around the snake point and then moving the snake point to the position in the neighborhood which minimizes the energy term. This phase continues until the number of unstable (moving) snake points drops below the threshold Th_{mov}. The contour resulting from this phase is used to generate a starting contour for the second phase.

As in Figure 3.4, show the Pseudo-code of the greedy snake algorithm, which starts with input s the image, determine ROI, preprocessing using Gaussian then Sobel filter, after that it computes the snake energy for all the control points neighborhood, and moving toward the minimum energy .

Algorithm 1 A GREEDY SNAKE

Input: : determine ROI, parameters α, β and γ

Output: Stop if only few points have moved

```
% n is the total number of snake control points
Index arithmetic for the snake control points is modulo n
Initialize the parameters  α, β and  γ
Do    % Main loop that moves the snake points to new locations
  for i = 1 to n  % The first and last point are the same in snake
                    Emin = infinity
         for j = 1 to m     % m is the neighborhood size
                    E(j) =  α Eela(j) + β Ecurv(j) + γ Eimg(j)
           if E(j) < Emin then     % Find location with min energy
                    Emin = E(j)
                    jmin = j
                    Move point v(i) to location jmin
             if jmin is not the current location then ptsmoved ++
  % The process below determines where to relax ?
        for i = 1 to n % Calculate exact curvature
           c(i) = || u(i) / || u(i) || - u(i+1) / || u(i+1)|| ||^2
        for i = 1 to n % Find where to relax ?
            if (c(i) > c(i-1) and c(i) > c(i+1)
               and c(i) > TH
               and mag(v(i)) > TH-mag
        then β(i) = 0     % Relax ? if all conditions true
           while ptsmoved > TH-moved
  % Stop if only few points have moved
```

Figure 3.4:Pseudo-code for the greedy snake algorithm[39].

3.2.3. Phase II

- Initializing contour for the second phase:

As explained earlier in our method's characteristics in section 3.1, to initialize the contour for the second phase the method calculates a center point amid the stable points, and moves those points to new positions accordingly. The center point $c(x_c, y_c)$ is calculated as the mean of the x and y coordinates of all the stable points.

Let $p_1(x_1, y_1), p_2(x_2, y_2), \ldots, p_n(x_n, y_n)$ be the stable control points. The coordinates x_c and y_c are calculated as follows:

$$x_c = \frac{1}{n}\ (x_1 + x_2 + \cdots + x_n) \qquad 3.5$$

$$y_c = \frac{1}{n}\ (y_1 + y_2 + \cdots + y_n) \qquad 3.6$$

$$c(x, y) = (x_c, y_c) \qquad 3.7$$

Where *c(x,y)* is the center position, and $p_i(x_i, y_i)$ is the position of the stable points in axes *x,y*.

The new position $p_i'(x_i', y_i')$ of each point shifts to the middle distance between the center and the point, and is calculated as follows:

$$\grave{x}_\mathrm{i} = x_\mathrm{i} + \frac{1}{2}(x_\mathrm{i} - x_c) \qquad 3.8$$

$$\grave{y}_\mathrm{i} = y_\mathrm{i} + \frac{1}{2}(y_\mathrm{i} - y_c) \qquad 3.9$$

After relocating the snake the greedy method resumes to find the final contour.

3.2.4. Termination criterion

When using the greedy method we have two stopping criteria.

The first one is if the numbers of points that move at each iterations fall under a certain threshold value. The other one being that the number of iterations reaches another threshold value. That mean the final step in the iteration of the greedy snake algorithm consists of checking wither the number of points moved in the iteration is below the threshold. This is used as a stopping criterion as the snake is presumed to have reached minimum energy when most of the control points have stopped moving. Note that, this idea is implementing on greedy snake in phase one and two.

Now the results we have, when the second phase-snake stable, are often the contour of the OOI, as we will see in the testing (see chapter4).

3.3. How the new method handles background clutter and boundary concavities

3.3.1. Background clutter

One of the problems that might prevent the snake from correctly finding the contours of an object is background clutter which is the presence of noise in the image. If the image contains noise edges the initial snake curve has to be placed closer to the object contour than otherwise the noise obstruct the active contour and prevent it from reaching and converging on the interest object's boundary.

As noise in the image might results in the snake stopping short before coming in contact with the image energy of the desired contour in the image. To prevent the snake from getting stuck in an undesired local minima created by noise, a technique known as scale space continuation can be applied. This technique was presented in the original paper by Kass et al. [3], as a way to improve the snake's ability to find the desired object contour even under the influence of noise.

We use this technique twice, in the first phase when implementing greedy snake initialization from outside and away from the object contour.
In the second phase, when creating a new greedy snake points from the center of the first snake, this helps us to reduce the noise of the image dramatically helps more from reaching the edges of the desired object, especially in both cases we use the value of σ from high to low, it will change when performing number of iteration, We'll explain the idea as follows:

The technique is rather straight forward. The idea is to use a large value for σ when first computing the image energy as we see in (chapter4).

Then let the snake iterate while slowly reducing the value of σ .When first using larger values for σ the noise is mostly smoothed and the snake can more easily pass over noise areas. The large amount of blurring also increases the capture range of the snake by making the influence of strong gradients stretch further away from the area in which they reside. However the large amount of blurring used in the beginning results in the snake doing a poor job of localizing the edge. Therefore σ is decreased and the snake then slowly adapts to the finer details of the contour. In order to use scale space continuation effectively the image energy is submitted to Min-Max normalization.

$$norm(E_{img}) = \frac{(E_{img} - \min(E_{img}))}{(\max(E_{img}) - \min(E_{img}))} * (0 - (-1)) + (-1) \qquad 3.10$$

Where norm(E_{img}) is the normalized image energy, min(E_{img}) the minimum value of the image energy and max(E_{img}) the maximum value of the image energy. After the Min-Max normalization the maximum values for the normalized image energy takes on the value 0 while the minimum is -1. The normalization makes sure that the range for the image energy stays constant even when varying σ. If this normalization of the image energy was omitted the image energy would generally be lower when using large values of σ.

3.3.2. Boundary concavities

Concavities in the boundary of an object pose a challenge to active contour (snake) methods. In this research, we present a two- phase snake for efficiently detecting contours of objects with boundary concavities. Our modifications to the internal term, the first internal energy term is the continuity energy term. The main role of the continuity energy term is to make even spacing between the snake points by minimizing the difference between the average distance and the distance between neighboring snake points, however the control points in the greedy snake are more likely to be evenly spaced.   This is suitable for objects with concavities because the distance between snake points should be relaxed in boundary concavities.

For curvature energy, the goal is to control the smoothness of the contour between neighboring snake points. The way in which this term is formulated affects the ability of snake points to progress into boundary concavities, such in greedy snake to computing the curvature for each of the points in the neighborhood of the snake point used the formula $(v(si + 1) - 2v(si) + v(si - 1))^2$, note that if we have magnitude a part of the curve where the last term would be greater than zero.   Since the curve actually does bend in Figure 3.5(a) we achieve the desired results. In Figure 3.5 (b) the points in the curve are not equally spaced and even though the curve does not bend in this part we notice that the term $(v(si+1) - 2v(si) + v(si-1))^2$ would be greater than zero., since the continuity term encourages even spacing of the control points, such when implementing the idea in the two phase, it's can successfully extract objects with boundary concavities, and is insensitive to the number of initial snake points.

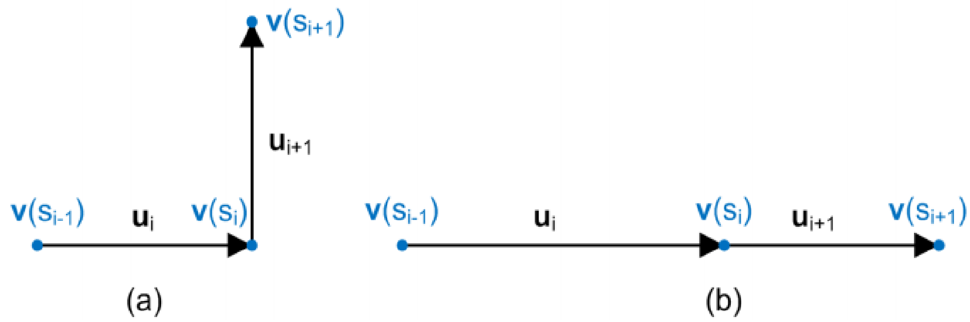


Figure 3.5.: Illustration showing two different parts of the snake curve. (a) In this part the curve is bending and the points are equally spaced since the two vectors u_i and u_i+1 have equal length. (b) In this part of the curve the true curvature is zero while the two Vectors u_i and u_i+1have different lengths.

3.4. Summary

In this chapter a two- phase snake method was presented and discussed in details. It was adding a second phase to the greedy algorithm(first phase) after shifting the snake in the first phase to a new start position, that ensures converging the OOI . The method introduced many choices to the user for defining the ROI manually. Other improvement was using scale space continuation on both of the two snake phases that, was handling clutter background more efficiently. Our method keeps the points in the snake equally spaced in the two –phase ,which was handling boundary concavities stronger .

Chapter (4):

Evaluation and Results

4. CHAPTER 4: Evaluation and Results

We implemented our method using MATLAB. The implementation includes a user interface through which the user load an image, specifies a ROI, specifies the set of parameters, and starts the snake. The snake performs all stages, preprocessing, phase1 and phase2, and generates the final contour superimposed on the image.

To test our method we compiled three test sets which include both real and synthesized images. Image Set 1 contained images with cluttered background only, no boundary concavities. Image Set 2 has images with boundary concavities only, no background clutter. While in Image Set 3 the images contained both background clutter and boundary concavities. This arrangement ensures a variety of test cases. They were run using different inputs and parameters. The following subsections present the results for each of these sets. In all test experiments, the ROI was specified as a circle and the initial snake contained 50 control points equally spaced around the perimeter of the circle. The interface shows accuracy indicated by an error measure.

To quantify the accuracy of our improved two- phase snake method, we use the square root of the mean/average of the square of all of the error (RMSE). The use of RMSE makes an excellent general purpose error metric for numerical predictions. Defined correct object of interest as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \qquad 4.1$$

where y_i is the new position of the points and $\hat{y}_i$ is the correctly position of the point on the boundary of OOI, and n is the number of the error points position.

In the following there are subsections present the results in three sections:

4.1. Test Set 1: Images with Cluttered Background only (no boundary concavities)

Images in this test set are selected to examine the performance of our method when only the background clutter problem is present. The image in Figure 4.1, is a sample of this set showing a guava fruit on a cluttered background of leaves and branches.

Figure 4.1: The original image "Test Set 1" .

A high degree of noise has been explained outside guava by using Sobel filter in Figure 4.2.

Figure 4.2: Sobel filter of the original image "Test Set 1".

Figure 4.3, 4.4 show the result of phase1 and phase2, respectively, of our method on the image of Figure 4.1. It took the snake 43 iterations to reach the contour in Figure 4.3, based on the parameter values: $\alpha = 1.4$, $\beta = 0.9$, $\gamma = 1.5$, $\sigma = 21$. Clearly, first phase snake has not found the correct contour of the guava fruit.

For phase2 as in Figure 4.4, the parameters were $\alpha = 2.2$, $\beta = 0.4$, $\gamma = 1.7$, $\sigma = 18$, neighborhood size 5, and the number of iterations was 92. Clearly, the second phase snake could find the true contour of the guava fruit accurately. The first phase snake could not find the contour because of the noise while the second phase snake avoided the noise by shifting to a new location.

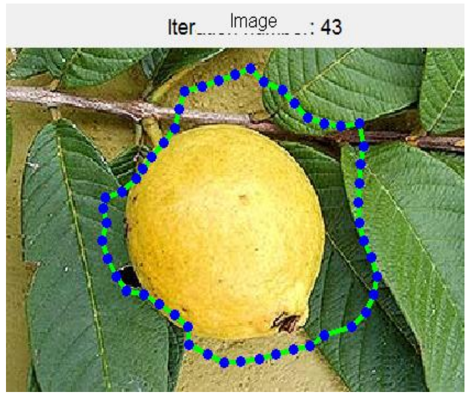


Figure 4.3: The result of first phase-snake" Test Set 1".

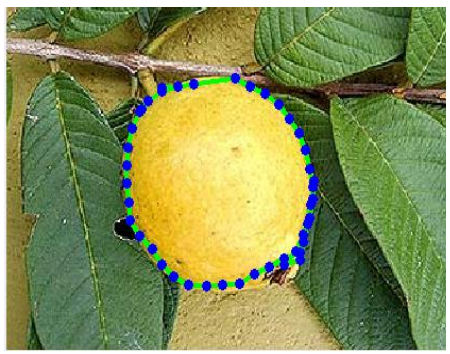


Figure 4.4: The result of second phase-snake" Test Set 1". .

4.2. Test Set 2: Images with Boundary Concavities only (no background clutter)

In this set, images are selected to examine the performance of our method when only the boundary concavity problem is present.

The synthesized image in Figure 4.5, is a sample of this set showing an object with many deep boundary concavities. Kass's traditional snake, the greedy snake and many other snakes fail to reach inside the concavities unless the control points were initially placed inside the concavity. If we wish to have a correctly segment object with boundary concavities using a snake we must make sure that the initial snake is placed inside the concavity. In this way the snake will be caught by the image energy and stick to the contour.

Figure 4.6, and Figure 4.7, show the result of phase1 and phase2, respectively, of our method on the image of Figure 4.5 . It took the snake 88 iterations to reach the contour in Figure 4.6, based on the parameter values $\alpha = 1.9$; $\beta = 1.3$; $\gamma = 1.8$; $s = 5$; $\sigma = 15$. The snake ran 88 iterations which is minimum than the maximum number of iterations, Clearly, first phase snake has found the correct contour of the OOI.

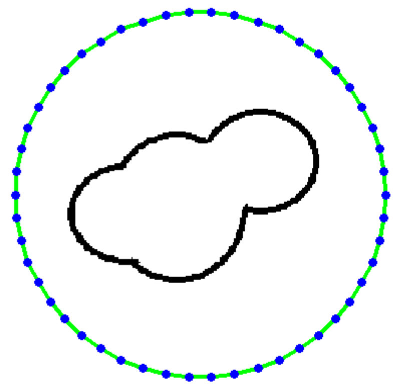


Figure 4.5: The original image "Test Set 2 ".

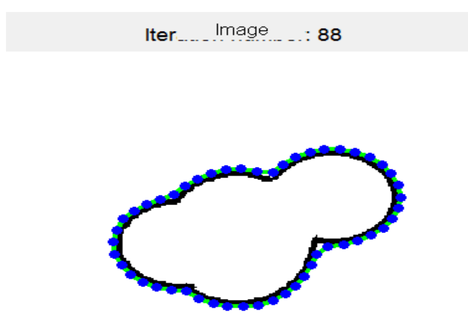


Figure 4.6: The result of first phase-snake "Test Set 2 ".

For phase2 see Figure 4.7, the parameters were $\alpha = 2.5$; $\beta = 0.6$; $\gamma = 1.1$; $s = 5$; $\sigma = 15$; $maxItr = 200$ and the number of iterations was 126. Clearly, the snakes in both phases could find the contour of the object. However, the contour of the second phase snake is more accurate with less errors.

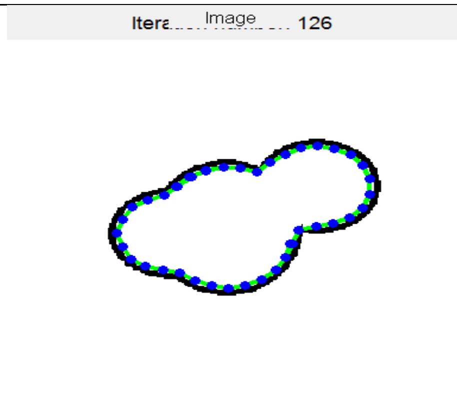


Figure 4.7: The result of second phase-snake" Test Set 2 ".

4.3. Test Set 3: Images with Both Cluttered Background and Boundary Concavities

In this set, images are selected to examine the performance of our method when both cluttered background and boundary concavity problems are present.

The real image in Figure 4.8, is a sample of this set showing an flower cluttered background of leaves and multiple deep boundary concavities and other details .

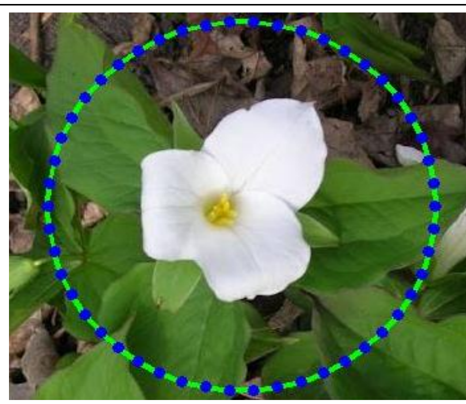


Figure 4.8: The original image " Test Set 3 ".

Figure 4.9, and Figure 4.10, show the result of phase1 and phase2, respectively, of our method on the image of Figure 4.8. It took the snake 34 iterations to reach the contour in Figure 4.9 based on the parameter values $\alpha = 1.5$; $\beta = 1$; $\gamma = 1.5$; $s = 5$; $\sigma = 15$ Clearly, the snake fails in phase1 because it could not overcome the background clutter, let alone to begin handling boundary concavities. However, when the snake shifts and performs phase2, it succeeds, as shown in Figure 4.10, in finding the object's boundary and reaching inside the concavities. The final state of the second phase was reached after 52 iterations, and the parameters were $\alpha = 1.5$; $\beta = 1$; $\gamma = 1.5$; s = 5; $\sigma = 15$; $maxItr = 200$.

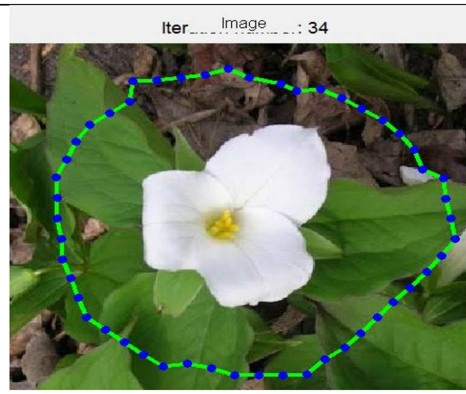


Figure 4.9: The result of first phase-snake" Test Set 3 ".

Clearly, the contour of the second phase snake is more accurate with less errors.

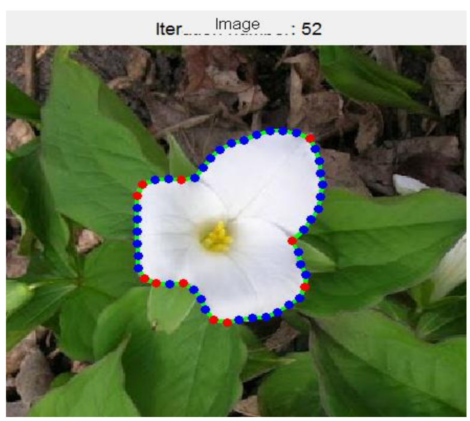


Figure 4.10: The result of second phase-snake" Test Set 3 ".

We observe also, that greedy snake hasn't found the contour because of the noise, but the second phase-snake found the contour although the noise in the image, see Sobel filter in Figure 4.11.

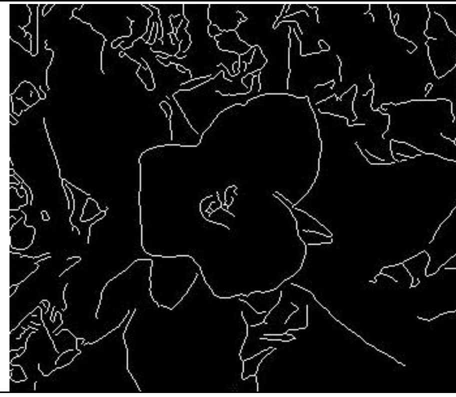
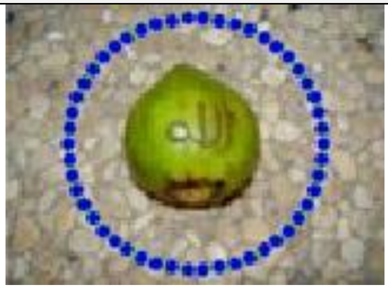
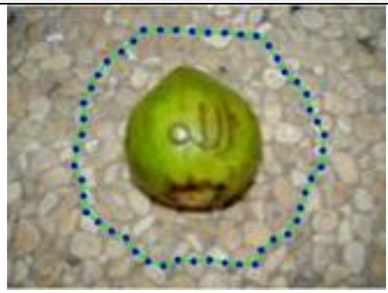
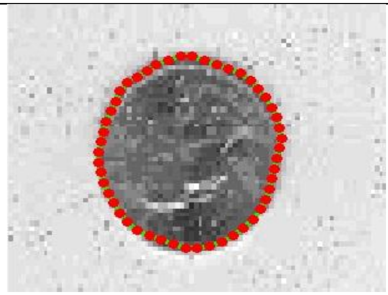
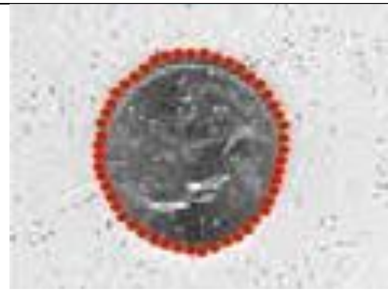
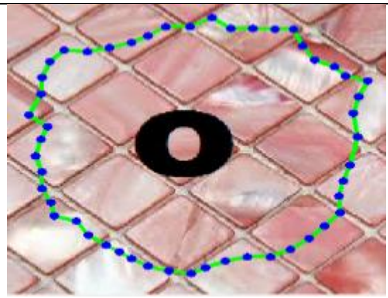
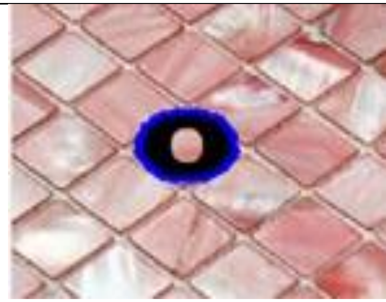


Figure 4.11: Sobel filter "Test Set 3 ".

Overall results

The following 3 tables summarize the results of multiple samples from each test set. Table 4.1, test set 1 explains the performance of our method when only background cluttered. Table 4.2, test set 2 explains the performance of our method when only boundary concavities, and Table 4.2, test set 2 explains the performance of our method when background cluttered and boundary concavities. In general, the first-phase snake fails to find the object's contour when either problem or both are present. On the other hand, the second-phase snake finds the boundary and reaches inside concavities with high accuracy as indicated by low error values.

Table 4.1: Test set 1 - Images with Cluttered Background only (no Boundary Concavities).

No.	Image	Edge Map	Phase1"Greedy Snake"	Error	Time	Phase2	Error	Time
1				36.2288	05.12		0	11.39
			α:1.6, β:1.0, γ:1.5, σ: 15, iteration:66			α:2.3, β:1.0, γ:1.8, σ: 15, iteration:74		
2				2.6067	04.11		0	09.01
			α: 0.5, β:1.1, γ:0.5, σ: 15, iteration:44			α: 2.5, β:1.5, γ: 1.8 , σ: 15, iteration:61		
3				74.5789	00.04		0	00.06
			α: 1.2, β:1.0, γ:1.85 , σ: 21, iteration:32			α: 1.9, β:1.0, γ:1.8 , σ: 15, iteration:53		

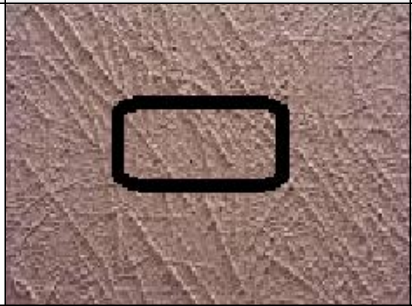
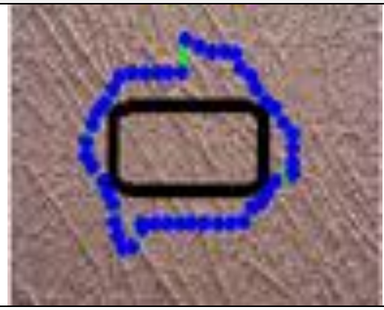
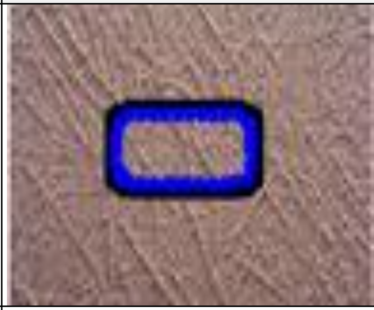
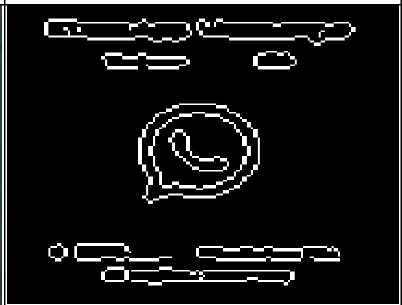
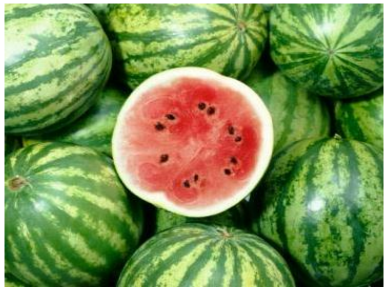
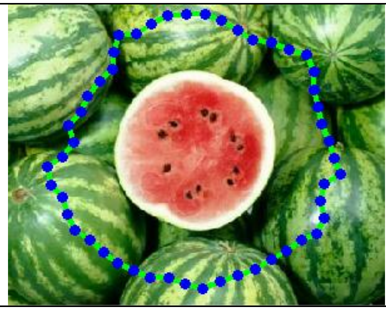
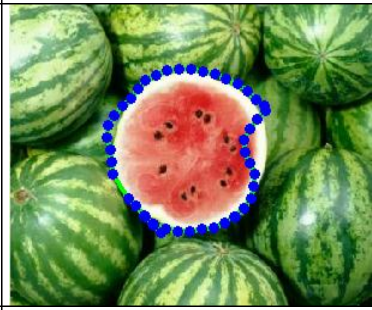
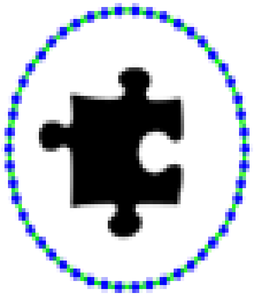
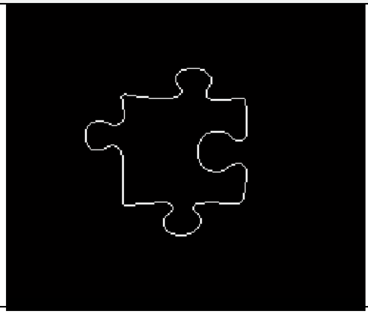
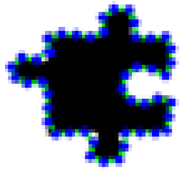
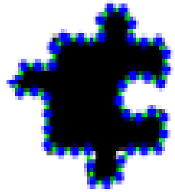
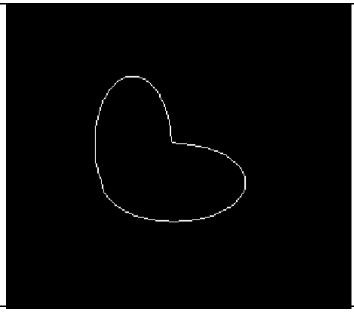
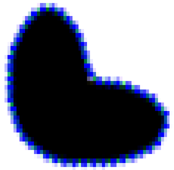
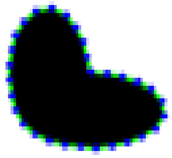
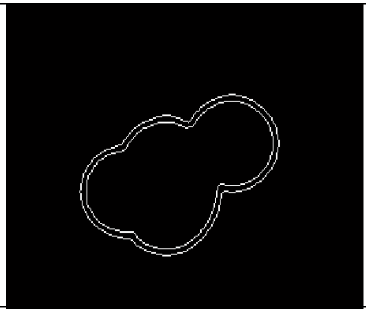
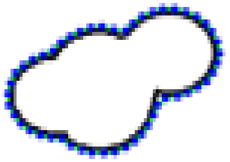
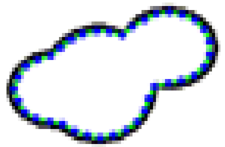
No.	Image	Edge Map	Phase1"Greedy Snake"	Error	Time	Phase2	Error	Time
4				13.5204	03.01		0	13.73
			α: 1.2, β:1.0, γ:1.7 , σ: 21, iteration:38			α: 2.2, β:1.2, γ:1.8 , σ: 15, iteration:138		
5				24.7831	00.02		2.8845	04.44
			α: 1.8, β:1.4, γ:1.6 , σ: 18, iteration:22			α: 2.3, β:1.0, γ:1.1 , σ: 15, iteration:41		
6				52.7941	03.21		14.4376	11.45
			α: 2.5, β:1.5, γ:2.5 , σ: 15, iteration:46			α: 2.5, β:1.3, γ:2.5 , σ: 25, iteration:115		

Table 4.2: Test set 2- Images with Boundary Concavities only (no cluttered background).

No.	Image	Edge Map	Phase1"Greedy Snake"	Error	Time	Phase2	Error	Time
1				**5.1093**	**07.34**		**2.6067**	**14.11**
			α: 1.5, β:1.1, γ:1.5 , σ: 15, iteration:103			**α: 1.5, β:1.15, γ:1.1 , σ: 15, iteration:129**		
2				**4.5608**	**05.22**		**0**	**09.47**
			α: 1.9, β:1.3, γ:1.8 , σ: 15, iteration:89			**α: 2.5, β: 0.5, γ:0.6 , σ: 15, iteration:74**		
3				**5.7291**	**04.00**		**0**	**09.15**
			α: 1.9, β:1.3, γ:1.8 , σ: 15, iteration:88			**α: 2.5, β: 0.6, γ:1.1 , σ: 15, iteration:126**		

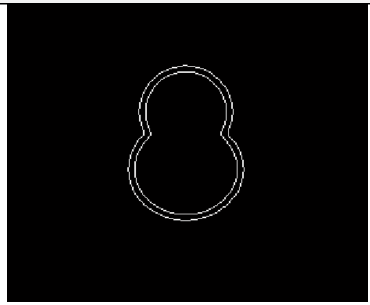
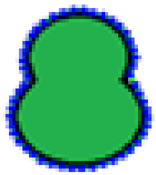
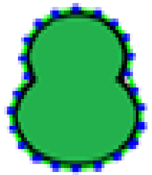
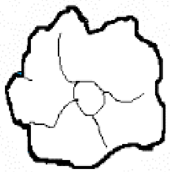
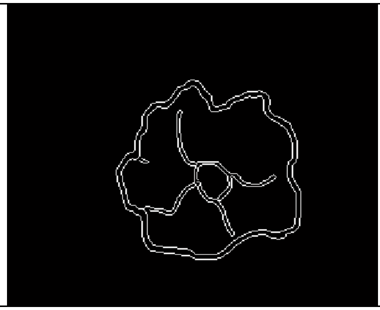
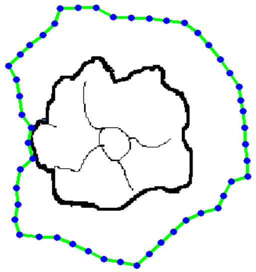
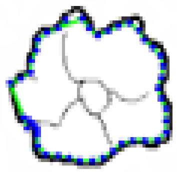
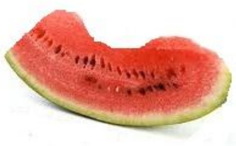
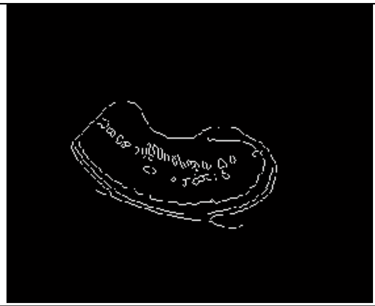
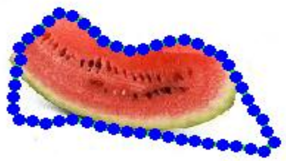
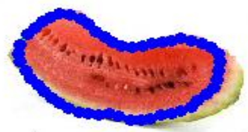
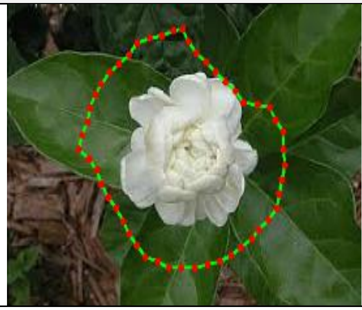
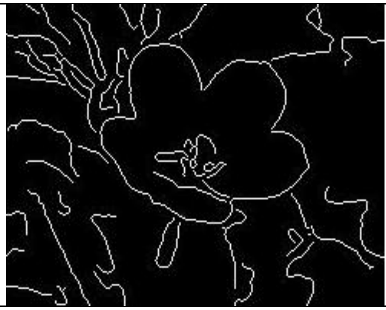
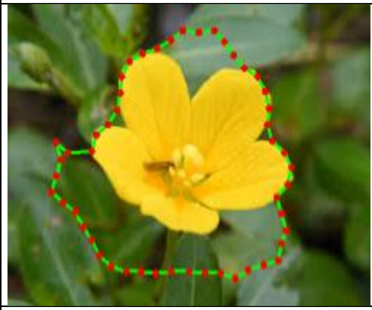
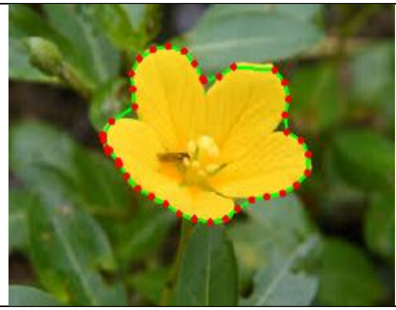
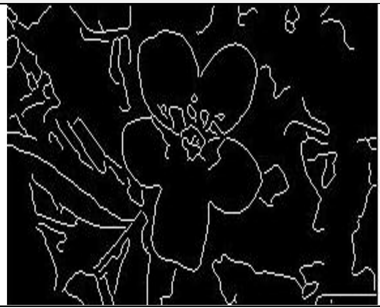
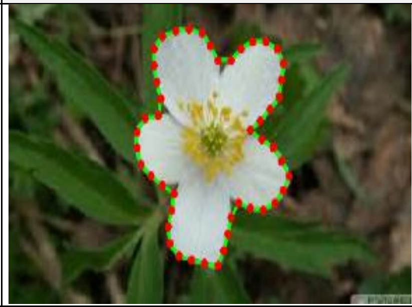
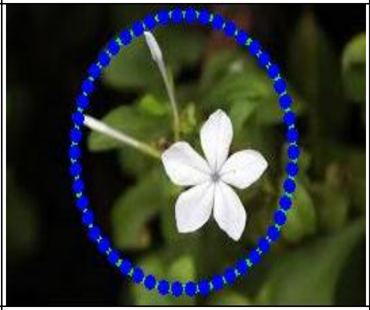
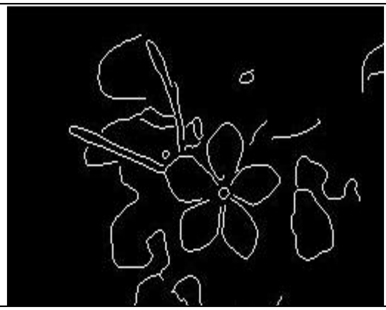
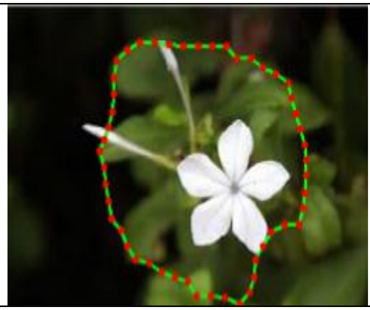
No.	Image	Edge Map	Phase1"Greedy Snake"	Error	Time	Phase2	Error	Time
4				3.8700	07.11		4.6062	16.20
			α: 1.9, β:1.3, γ:1.8 , σ: 15, iteration:120			α: 2.6, β: 0.5, γ:1.1 , σ: 15, iteration:169		
5				46.1225	04.50		6.6359	08.22
			α: 1.2, β:1.0, γ:1.805 , σ: 15, iteration:58			α: 1.45, β:1.15, γ:1.8 , σ: 15, iteration:102		
6				27.2626	06.20		15.3173	10.25
			α: 1.6, β:1.5, γ:1.5 , σ: 15, iteration:96			α: 1.9, β:1.3, γ:1.9 , σ: 21, iteration:105		

Table 4.3: Test set 3- Images with Cluttered Background and Boundary Concavities.

No.	Image	Edge Map	Phase1"Greedy Snake"	Error	Time	Phase2	Error	Time
1				24.8969	04.00		0	10.07
			α:1.1, β:1.3, γ:1.1, σ: 15, iteration:63			α:1.5, β:1.4, γ:1.5, σ: 15, iteration:47		
2				0	02.09		0	03.11
			α:1.5, β:1.3, γ:1.5, σ: 15, iteration:46			α:1.5, β:1.4, γ:1.5, σ: 15, iteration:52		
3				29.19	03.20		7.3688	06.45
			α:1.6, β:1.5, γ:1.6, σ: 15, iteration:59			α:1.6, β:1.2, γ:1.55, σ: 15, iteration:120		

No.	Image	Edge Map	Phase1"Greedy Snake"	Error	Time	Phase2	Error	Time
4				**4.7187**	**02.45**		**4.2883**	**04.20**
			α:1.8, β:1.2, γ:1.8, σ: 15, iteration:55			**α:1.8, β:1.48, γ:1.8, σ: 15, iteration:98**		
5				**47.248**	**02.00**		**5.9404**	**10.20**
			α:1.5, β:1.25, γ:1.5, σ: 15, iteration:30			**α:1.5, β:1.35, γ:1.5, σ: 15, iteration:119**		
6				**21.5084**	**02.33**		**45.7179**	**08.05**
			α:1.6, β:1.5, γ:1.5, σ: 15, iteration:42			**α:2.0, β:0.7, γ:2.0, σ: 21, iteration:92**		

4.4. Discussion

Our experimental results show that snakes can be used to segment a variety of different images. It was also established that scale space continuation greatly reduced the influence of cluttered background images.

Note that the image has a cluttered background only (no boundary concavities) see (Table 4.1), when implementing the first phase- snake, the snakes initiation was far away from the OOI, it hasn't found the contour except a few images, (see image 2 from Table 4.1) , that is because of cluttered background , such noise edges obstruct the snake to evolve toward the OOI. The second phase starts after shifting the first phase, but after implementing it, it has found the contour, see images (1: 6) from (Table 4.1) so the result in the second phase is more accurate generally , see Figure 4.12.

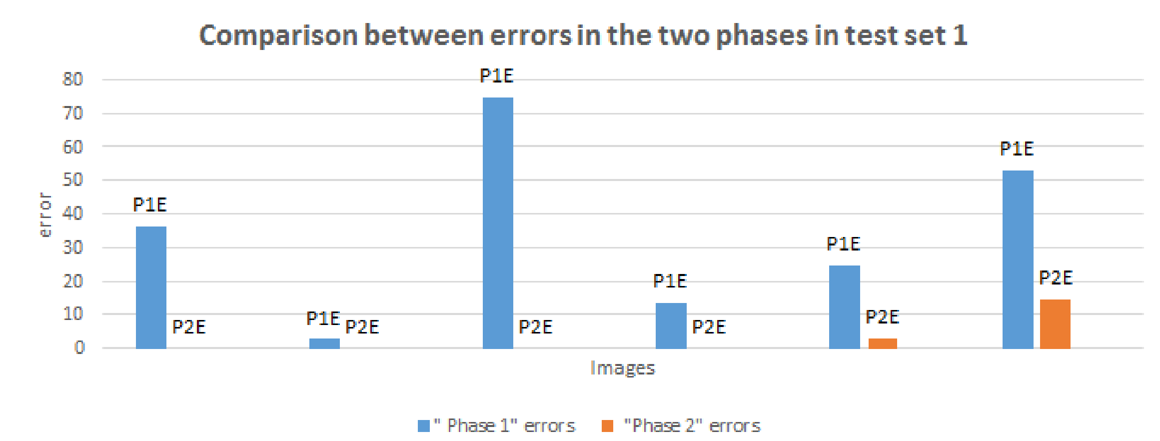


Figure 4.12: Comparison between errors in a cluttered background only (no boundary concavities) in test set 1 .

In Addition to that, using images with multi size OOI in our test has found the contour of OOI. That means, the best results are given after implementing the two- phase snake. We also recommend that, if images have a boundary concavities (no cluttered background) see (Table 4.2), the snake s initiation was far away the OOI, when implementing the first phase and after implementing the second phase in many images , we observe that it has found the contour in the two phases, see images (1:4) from (Table 4.2) , but the result in the second phase is more accurate , see Figure 4.13.

Note that images 5, 6 from (Table 4.2), when implementing the first phase hasn't found the contour, but after implementing the second phase, it has found the contour. In Addition to that, using images with multi size OOI in our test has found the contour of OOI.

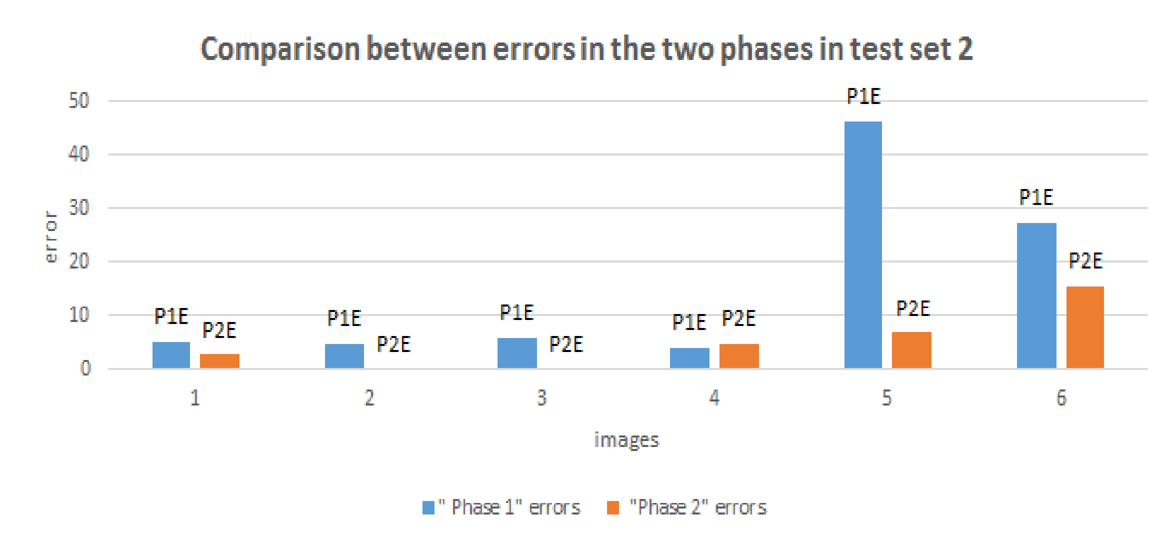


Figure 4.13: Comparison between errors in boundary concavities only (no a cluttered background) in test set 2.

In images which have a cluttered background and boundary concavities see (Table 4.3), the snake's initiation was far away from OOI, when implementing the first phase, it hasn't found the contour except few images see images 2,4 from (Table 4.3) , but after implementing the second phase, it has found the contour, such the result after the second phase is more accurate in most images, so the result after implementing the two- phase snake is more accurate generally , specially images with big size OOI as image 6 from (Table 4.3), hasn't found the contour of OOI as satisfactory, see Figure 4.14.

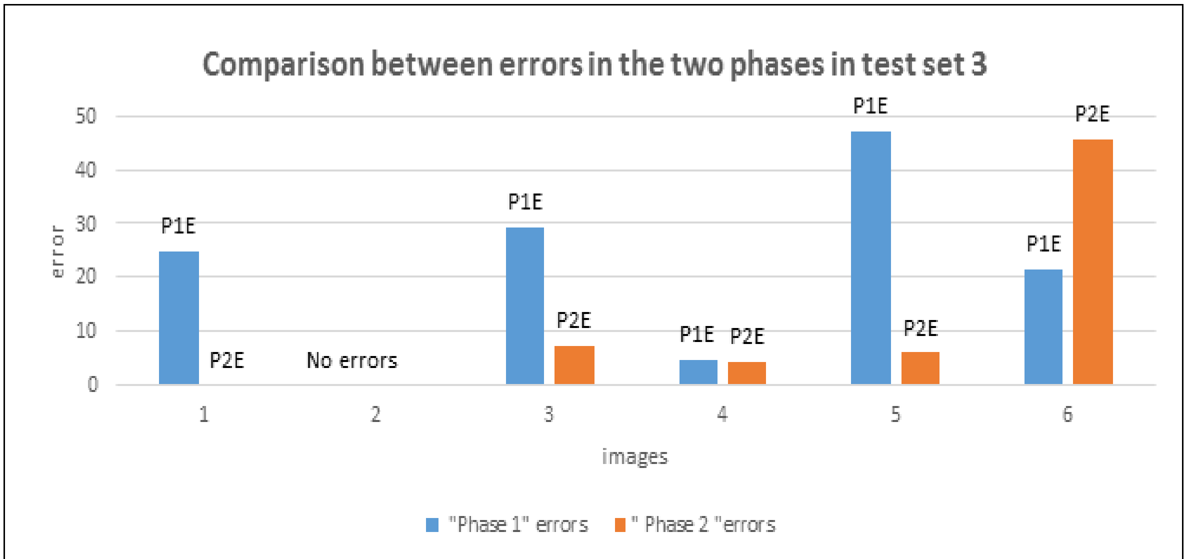


Figure 4.14: Comparison between errors in boundary concavities and a cluttered background in test set 3.

From the error result, observing that the last greedy snake was succeed in a few images to solve the two problems (background clutter and boundary concavities), but in our new idea, by using snake in two phases, the result be more accurate and more efficiently.

In addition to that, shortcoming of traditional snake is that, it requires user interaction, which consists of determining the curve around the detected object ,but in our proposed method the user can determine initial position as he see, that can't obstruct determine the OOI, this properties add more efficiently.

Also, the image has small gabs in the contour edge, see edge map from (Table 4.1, 4.2, 4.3), the two- phase snake overcome the gabs and find the OOI contour.

- **The weakness**

Few weakness in the new snake idea, the method requires a balance between how close or far away to initialize the ROI around the OOI, see image 6 in (Table 3: Test set 3).

On the other hand when comparing between phase1

And the two- phase snake according to the time per second, it will be clear that the time during the two- phase snake is longer than the time during phase1 see Figure 4.15, 4.16, 4.17.

This it consider a weak point even though it will give a better result which is determining the OOI.

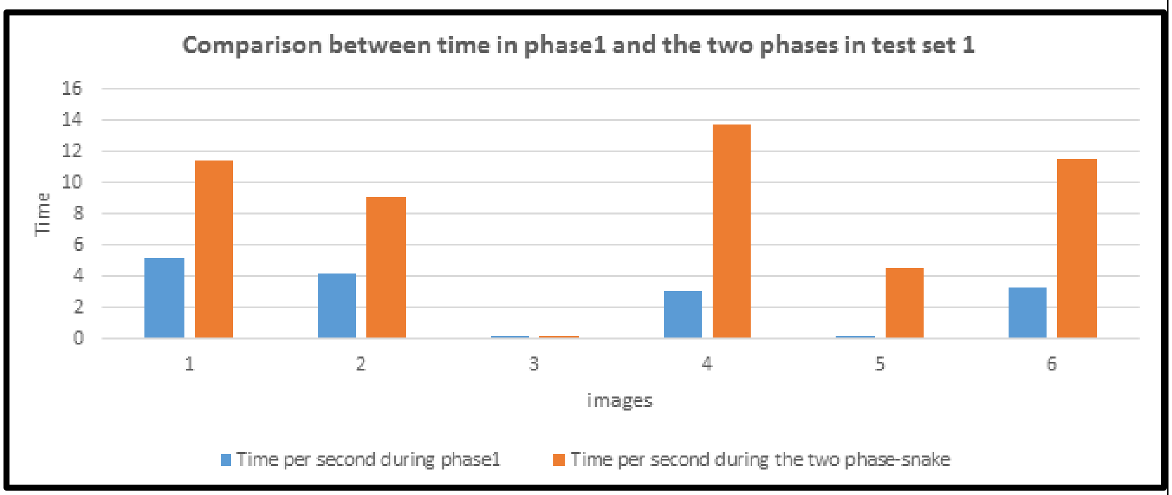


Figure 4.15: Comparison between time in cluttered background only in test set 1.

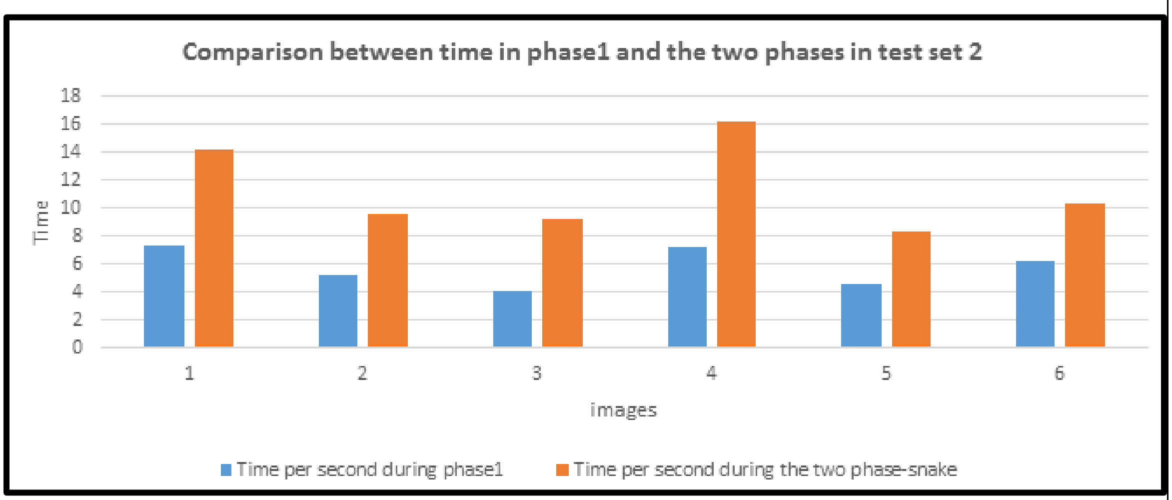


Figure 4.16: Comparison between time in boundary concavities only in test set 2.

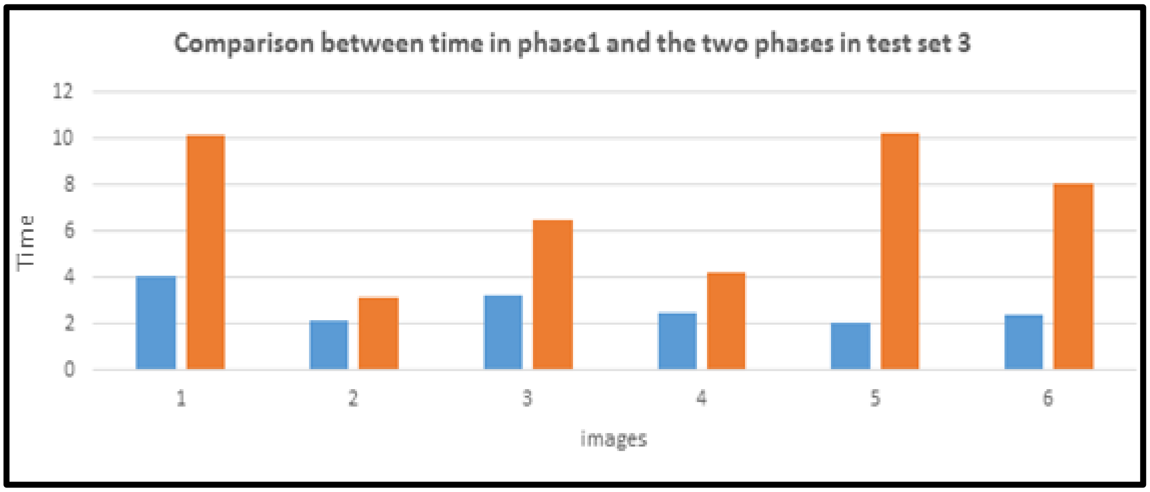


Figure 4.17: Comparison between time in boundary concavities and a cluttered background in test set 3.

4.5. Summary

In this chapter the experiments were presented and discussed in details. The experiments were done using the two-phase snake method . They were tested on a number of different images with combinations of clutter and concavity problems.

To test our method we compiled three test sets which include both real and synthesized images. Image Set 1 contained images with cluttered background only, no boundary concavities. Image Set 2 had images with boundary concavities only, no back ground clutter. While in Image Set 3 the images contained both background clutter and boundary concavities.

The experimental results explained that the two-phase active contour could achieve the research objectives under most conditions .

Chapter (5):

Conclusion and Future Work

5. CHAPTER 5: Conclusion and Future Work

5.1. Conclusion

The main goal of this proposed method was to develop a segmentation snake-based method that could perform more efficiently in the presence of background clutter and boundary concavity.

The active contour method that was chosen for implementing and comparison was the greedy snake. The primary improvement introduced in our method was adding a second phase to the greedy algorithm after shifting the snake to a new start position. Another improvement was using scale space continuation on both of the two snake phases.

In addition to that, the user can determine the ROI manually and the snake initiation is far away to the OOI contour, but this will be very sensitive when selecting points of the snake, overcoming this by insert or delete points using function computes the average distance between the snake control points. These improvements contributed to the better performance of our method.

The two-phase snake method was tested on a number of different images with combinations of clutter and concavity problems. In the experimental results section it was established that under most conditions our two-phase active contour could achieve the research objectives.

The experimental results also show that, it is possible to closely compare the performance of each snake-phase, the difference between the result when using the first phase and the two phases was more clearly, such the best result were given after implementing the two phases .

So the goal of developing a segmentation snake-based method has been reached by using two phase-snake.

On the other hand, other weaknesses can be mentioned the fact that, the parameters of the snake has to be adjusted manually. Another few weakness in the new snake idea, the method requires a balance between how close or far away to initialize the ROI around the OOI.

5.2. Future Work

- We have throughout the project deal of many different size objects, except the ones which have a very big size comparing with background so that, one possible extension could be to define a balance ratio between how close or far away to initialize the ROI around the OOI.
- We have throughout the project only been considering snakes that form closed curves, some images have gabs in the OOI contour edge map, in our research we overcome small gabs. The extension could be to extend implemented a program to deal with big gabs (open curve snakes).
- Another extension is to find contours in the images that weren't found in our research such as RMI etc ...

6. Bibliography

[1] Amit,Lawarence,staib and James " Deformable Boundary Finding in Medical Images by Integrating Gradient and Region Information" ,*IEEE Transactions On Medical Imaging* ,1996 .

[2] Rami Ben-Ariand Dror Aiger" Geodesic Active Contours with Combined Shape and Appearance Priors", 2005.

[3] Kass, M., A. Witkin, and D. Terzopoulos. "Snake: Active Contour Models." Int'l J. Computer Vision, vol. 1, no.4, pp. 321-331, 1987.

[4] C. Xu and J.L. Prince, "Gradient Vector Flow: A New External Force for Snakes", IEEE Conf. on Comp. Vis. Patt. Recog. (CVPR), Los Alamitos: Comp. Soc. Press, pp. 66-71, June 1997.

[5] Chenyang Xu and Jerry L. Prince , "Snakes, Shape, and Gradient Vector Flow", IEEE Transactions on Image Processing, 1998.

[6] Xu and Prince" Gradient Vector Flow: A New External Force for Snakes"IEEE Proc. Conf. on Comp. Vis. Patt. Recog. (CVPR'97).

[7] Ravikanth, James, and Baba,"Shape modeling with front propagation: A level set approach," *IEEE Trans. Pat. Anal. Mach. Intell.,* vol. 17, no. 2, pp. 158-175, June 1995.

[8] Seema and Poonam ," An Improved Region Based Active Contour Model for Medical Image Segmentation" *International Journal of Signal Processing, Image Processing and Pattern Recognition* Vol. 8, No. 1 (2015)

[9] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: A level set approach," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 2, pp. 158–175, 1995.

[10] F. Shih and K. Zhang, "Locating object contours in complex background using improved snakes," *Computer Vision Graphics and Image Understanding*, vol. 105, no. 2, pp. 93–98, 2007.

[11] X. Bresson, S. Esedoglu, P. Vandergheynst, J. Thiran, and S. Osher,"Fast global minimization of the active contour/snake model," *Journal of Mathematical Imaging* and Vision, vol. 28, no. 2, pp. 151–167, 2007.

[12] C. Xu and J. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 359–369, 1998.

[13] B. Nagarajan, and P. Balasubramanie " Cluttered Background Removal in Static Images with Mild Occlusions " *International Journal of Recent Trends in Engineering, Vol. 1, No. 2*, May 2009 .

[14] http://en.wikipedia.org/wiki/Active_contour_model, visited on 25th May, 2014.

[15] Alattar A. M., "A Multi-Level Edge-Based Stereo Correspondence Method for Snake-Based Object Segmentation", Graduate School of PaiChai University, pp. 13-29, December 2008.

[16] Y. Wong, P. Yuen, and C. Tong, “Segmented snake for contour detection,” *Pattern Recognition*, vol. 31, no. 11, pp. 1669–1679, November 1998.

[17] Grigorios Tsagkatakis and Andreas Savakis, “Online Distance Metric Learning for Object Tracking” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 21, no. 12, 1810-1821, 2011.

[18] Ning, Zhang, " Joint Registration and Active Contour Segmentation for Object Tracking", *College of Information Engineering, Northwest A&F University, Yangling, Shaanxi, China*, vol. 10, no. 2, 2012.

[19] C. Davatzikos and J. L. Prince." An active contour model for mapping the cortex", *IEEE Trans. on Medical Imaging*, 14(1):65–80, Mar. 1995.

[20] T. McInerney and D. Terzopoulos, "T-snakes: Topology adaptive snakes" Medical Image Analysis,vol.4, pp. 73-91, 2000.

[21] Cheng, SunRes. J." Medical Image Segmentation with Improved Gradient Vector Flow ", *Appl. Sci. Eng. Technol*., 4(20): 3951-3957, 2012

[22] D. Williams and M. Shah, "A fast algorithm for active contours and curvature estimation", Computer Vision, Graphics, and Image Processing: Image Understanding 55, 1992, 14-26

[23] D. COHEN ,"On Active Contour Models ,balloon" Laurent,1998

[24] Staib, L., Duncan, J."Boundary finding with parametrically deformable models" *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14(11) (1992) 1061–1075

[25] Mengmeng Zhang, Qianqian Li, Lei Li, Peirui Bai" An Improved Algorithm Based on the GVF-Snake for Effective Concavity Edge" *Journal of Software Engineering and ApplicationsVol.6 No.4*(2013), Article ID:30236,5 pages.

[26] C. Xu and J. Prince, “Generalized gradient vector flow external force for active contours”, Signal Processing, vol. 71, pp. 131-139, 1998.

[27] Shin-Hyoung,Kim,Alattar and Jong Whan Jang "A Snake-Based Segmentation Algorithm for Objects with Boundary Concavities" *IEEE International Conference on Multimedia and Expo*, 2006.

[28] Shin-Hyoung Kim, Ashraf M. Alattar and Jong Whan Jang "Accurate Contour Detection Based on Snakes for objects with boundary concavities", (2006).

[29] Shin-Hyoung,Kim, Alattar and Jong Whan Jang "Snake-based contour detection for objects with boundary concavities", Opt. Eng. 47(3), 037002 (April 02, 2008). ; http://dx.doi.org/10.1117/1.2894148 .

[30] Anucha , Wichian, " Automatic Region of Interest Detection in Natural Images "Graduate School of Information Technology in Business Siam University, 2011 .

[31] Riklin-Raviv, T., Kiryati, N., Sochen, N." Prior-based segmentation and shape registration in the presence of projective distortion". IJCV 72 (2007) .

[32] Chan, T., Vese, L.:" Active contours without edges" IEEE Transactions on Image Processing 10 (2001) 266{277}.

[33] Alban , Pierre, and Fabrice Heitz " Multi-Reference Shape Priors for Active Contours" Author manuscript, published in "International Journal of Computer Vision 81 (2009)

[34] Liyuan Li, Weimin Huang, Irene Yu-Hua Gu, and Qi Tian," Statistical Modeling of Complex Backgrounds for Foreground Object Detection " IEEE Transaction On Image Processing VOL. 13, NO. 11, November 2004 .

[35] Xintong, Xiaohan, Chen" Foreground Segmentation Based On Multi-Resolution And Matting" 2012.

[36] http://en.wikipedia.org/wiki/Sobel_operator, visited on 25th May, 2014.

[37] J. F. Canny, "A computational approach to edge detection." *IEEE Trans. Pattern Anal.* Mach. Intelligence PAMI-8 (1986), 679-698

[38] http://en.wikipedia.org/wiki/Gaussian_filterr, visited on 25th May, 2014.

[39] Liyan Zhang, " Active Contour Model, Snake", Department of Computer Science, University of Nevada, Reno, 2008.