نموذج رقم (1)

إقـــــرار

أنا الموقع أدناه مقدم الرسالة التي تحمل العنوان:

An Adaptive Active Contour Model For Building Extraction From Aerial Images

أقر بأن ما اشتملت عليه هذه الرسالة إنما هو 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 أي مؤسسة تعليمية أو بحثية أخرى.
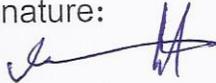
DECLARATION

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: Mohammad AbdelHameed Ondah

Signature:

Date:

24/2/2016

اسم الطالب: محمد عبد الحميد محمد عودة

التوقيع:

التاريخ: 2016 /2/24

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

An Adaptive Active Contour Model for Building

Extraction from Aerial Images

Submitted by:

Mohammad Abdel Hameed Oudah
120110151

Supervised by:

Dr. Ashraf Alattar

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

Jan 2016

بسم الله الرحمن الرحيم

الجامعة الإسلامية – غزة
The Islamic University - Gaza

مكتب نائب الرئيس للبحث العلمي والدراسات العليا   هاتف داخلي: 1150

الرقم: س/غ/35/........... Ref
التاريخ: 2016/01/30 Date............

نتيجة الحكم على أطروحة ماجستير

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

نموذج كنتور نشط محسن لاجتزاء المباني من الصور الجوية

An Adaptive Active Contour Model for Building Extraction from Aerial Images

وبعد المناقشة العلنية التي تمت اليوم السبت 20 ربيع الآخر 1437هـ، الموافق 2016/01/30م الساعة الواحدة ظهراً بمبنى طيبة، اجتمعت لجنة الحكم على الأطروحة والمكونة من:

د. أشرف محمد العطار	مشرفاً و رئيساً
د. أشرف يونس مغاري	مناقشاً داخلياً
د. سامي عبد الله سلامة	مناقشاً خارجياً

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

واللجنة إذ تمنحه هذه الدرجة فإنها توصيه بتقوى الله ولزوم طاعته وأن يسخر علمه في خدمة دينه ووطنه.

والله ولي التوفيق ،،،

نائب الرئيس لشئون البحث العلمي والدراسات العليا

أ.د. عبدالرؤوف علي المناعمة

إهـــداء

إلى أبـي الغالي

إلى أمـي الغالية

إلى عائلتي الغالية إخواني وأخواتي

إلى زوجتي وأبنائي أنس ألما ولما

إلى روح عمي الغالي أبوخالد نسيم نعيم

إلى روح شهـــداء فلسطـين والأمـة الإسـلامية

إلى أصدقائي الأعزاء وزملائي في العمل والدراسة

Abstract

Building extraction from aerial images is one of the recent topics of remote sensing that used in many applications such as urban planning, disaster management, military planning, and Geographic Information Systems (GIS).

One of the most used approaches in building extraction is the Active Contour Model (ACM) or snakes for its ability to extract contours of structured and unstructured shapes of objects. However, using the traditional ACM snake model in building extraction and other fields faces the problem of extracting contours of concavity regions, because snake points cannot converge inside narrow concavity regions during its movement.

In this research we proposed to solve extraction contours of concavity regions problem by adapting coefficients of ACM forces during snake iterations by adding a concavity index to indicate that snake points stop in a concave region or not. Then adapt these coefficients in term of concavity index value, to allow snake converge inside concavity region.

Our adaptive model was tested on different sets of sub aerial images of buildings that contain concavity regions, we show the results and evaluate these results using two evaluation methods, the first evaluation method done in terms of accuracy, precision and recall. And the second evaluation method evaluates the Error Distance Ratio (ER_d) which is the average ratio of distance between each snake point and the true edge map point (by pixels), the result values is compared with the GVF snake model, which is an important improved ACM model that solved the concavity contour extraction problem. In addition, we test and compare the execution time of our adaptive ACM model and GVF model.

Keywords: Building Extraction, ACM Model, Snake Model, GVF Model.

ملخص

استخراج المباني من الصور الجوية هو أحد الموضوعات الحديثة في مجال الاستشعار عن بعد لما له من استخدامات في العديد من التطبيقات مثل التخطيط الحضري، إدارة الكوارث، التخطيطات العسكرية، ونظم المعلومات الجغرافية.

وتعتبر طريقة الكنتور النشط (ACM) أحد أكثر الطرق استخداماً في عملية استخراج المباني من الصور الجوية لقدرتها العالية على استخراج حدود الأشكال المنتظمة والغير منتظمة. بالرغم من ذلك استخدام هذه الطريقة في استخراج المباني من الصور الجوية تواجه مشكلة تحديد حدود المباني في المناطق المقعرة داخل المبنى أو التي تحتوي تجاويف ضيقة، لعدم قدرة نقاط الكنتور النشط على الدخول داخل المناطق المقعرة.

في هذا البحث قمنا باقتراح حل لمشكلة تحديد حدود المناطق المقعرة بتكييف عوامل القوة المؤثرة في حركة الكنتور بإضافة معامل جديد سمي بمؤشر التقعر أو (concIndex) ليدل على وقوف نقطة الكنتور في منطقة مقعرة. ويتم إعادة احتساب معاملات القوة لجميع النقاط خلال حركة الكنتور النشط بناء على مؤشر التقعر ليسمح بدخول الكنتور إلى داخل المناطق المقعرة.

تم إجراء اختبار للطريقة المقترحة باستخدام مجموعات مختلفة لصور جوية لمباني تحتوي مناطق مقعرة، بطريقتي اختبار مختلفتين. الطريقة الأولى تعني بإيجاد دقة النتائج من حيث اكتشاف نقاط الحدود الصحيحة، والطريقة الثانية تعني باحتساب نسبة الخطأ في بعد كل نقطة من النقاط المكتشفة عن النقطة الصحيحة، وقمنا بعد ذلك بمقارنة نتائج الاختبارين بطريقة الـ GVF باعتبارها أحد أهم الطرق التي طورت طريقة الـACM التقليدية لمعالجة مشكلة اكتشاف حدود المناطق المقعرة في الأشكال.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACM	Active Contour Model
CNN	Centroid Neural Network
ConIndex	Concavity Index
DP	Dynamic Programming
$\textbf{\textit{ER}}_{\textit{d}}$	Error Distance Ratio
GA	Genetic Algorithm
GUI	Graphical User Interface
GIS	Geographic Information Systems
GVF	Gradient Vector Flow
ROI	Region of Interest
SVM	Support Vector Machines

1 Introduction

Building extraction is the process of defining the correct contour of buildings using extracted features with decreasing human support. Figure 1 shows an extracted buildings example from an aerial image.

Building extraction from aerial images is an important problem of remote sensing field and it became a research topic for its important value in many applications such as urban planning, disaster management, military planning, geographic information systems (GIS) and other applications.

Figure 1: Building extraction from aerial image

Many approaches have been proposed for building extraction from aerial images, these approaches can be generally categorized to automatic and semi-automatic approaches, both use different image characteristics to extract building objects.

Many challenges hinder object extraction from aerial images such as image quality, resolution, noise, and lighting conditions [1]. Moreover, buildings are usually close to each other, and many other objects are close proximity such as trees, vehicles, parking areas, power lines, shadows, and buildings have differentiated structures. These challenges make it is difficult to perform a full automatic and accurate building extraction process and it became a research area and researchers have improved different approaches to overcome these challenges. The proposed approaches used different image segmentation techniques. One of the most used techniques is Active Contour Model (ACM).

Active Contour Model (ACM) is one of the most used techniques in building extraction and contour detection for its ability to detect irregular shapes. ACM (also called snake) was defined by Kass et al. as an energy minimizing, deformable spline influenced by constraint and image forces that pull it towards object contours [2]. It is deformed due to external forces that attract it towards salient features of the image and internal forces that try to preserve the condition of smoothness in the shape of the curve.

ACM or Snake model is used for object tracking, shape recognition, segmentation, edge detection, stereo matching. Traditional ACM which defined by Kass have multiple shortcomings, one of these shortcomings is that contour extraction is strongly dependent on the initial value of the snake, and it is very sensitive to noise and weak edges lead to leakage, also it cannot converge inside deep narrow concavities. Therefore, using ACM for building extraction faces all the mentioned challenges as shown in Figure 2
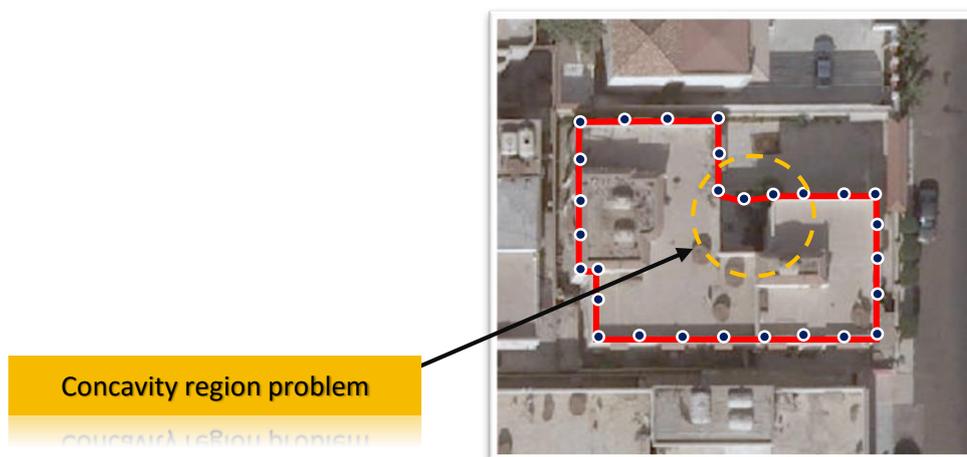
Figure 2: ACM concavity regions problem

The different developed approaches for automatic building extraction concentrated on detecting, locating buildings, and give an initial shape for the building contour. Some other researches make some additional operations of fining the detected contour.

In our thesis, we intend to use ACM to solve extracting contours of buildings that have concavity regions and modify the ACM Model to allow snake converge inside concavity regions by adapting internal and external energies, Figure 2 shows a case of the problem that we intend to solve that is building includes a concavity region.

1.1 Problem Statement

According to the mentioned characteristics of traditional ACM model and the controlling forces of snake movement, snake model cannot converge inside deep concavity regions, therefore it cannot extract accurate contour concavity regions in different shapes. And so on, applying ACM for building extraction from aerial images have the problem of concavity contours extraction, where many buildings contains deep narrow concave regions as shown in Figure 2.

1.2 Research Objectives

Our research aims to achieve a main objective and a set of sub objectives that when achieved enables the achievement of the main objective.

1.2.1 Main Objective

The main objective of this research is to make improvements on ACM model for enabling snake converging inside narrow concave regions while extracting building contour.

1.2.2 Sub objectives

1. Understanding building extraction state of Art.
2. Detect reasons of ACM failure for detecting concave regions.
3. Propose suitable solution for ACM problem.
4. Design ACM modifications
5. Implementation of ACM modified Design.
6. Make Experiments of our proposed modifications.
7. Test the results of our new modifications on ACM model.
8. Compare our adaptive ACM model results with GVF model because it is one of the most used improved ACM model for extracting contours of concavities.

1.3 Scope and limitation

1. This study does not cover the problem of attached buildings extraction such as shown in Figure 3, it covers only separated buildings.

2. This study does not cover extraction of buildings with weak edges or uncompleted edges caused by trees, shadows, or high intensity.

3. We use in our study existing approaches for building extraction preprocessing and don't propose any preprocessing steps.

4. Input aerial image must be 15 cm per pixel resolution at least and should be in good lighting and weather conditions.

5. This study depends on contour extraction for sub image only from complete aerial image (image contains one building have narrow concavities) as shown in Figure 4, and this sub image must be extracted by user manually.

6. The proposed building extraction process depend on 2D top view extraction and do not operate with 3D model.

7. The proposed solution was tested on building extraction applications only, and not tested on other object extraction.

Figure 3: Attached buildings example
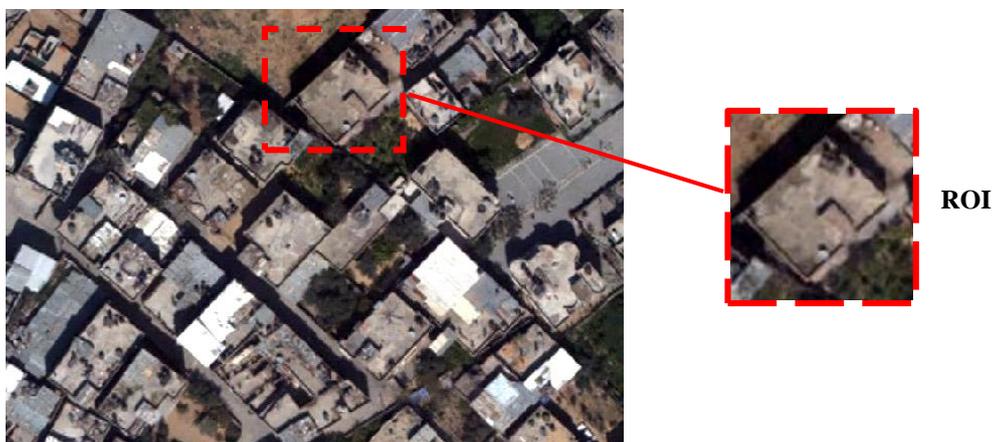
Figure 4: Proposed method sample area and region of interest (ROI)

1.4 Significance

Our thesis provides an adaptive ACM model with an ability to overcome the challenge of extracting concavity regions contours without adding new forces to the traditional snake model in order to save processing time and computations. By achieving our objectives, our improvements should support many applications that depends on buildings extraction, such as urban planning, military purposes, GIS applications and others.

1.5 Research Format

This report is organized as follows, in chapter 2, we present the theoretical background of building extraction field, then in chapter 3 we review some selected related works for building extraction using ACM model and using other approaches. After that, we present our improved ACM method, and in chapter 5 we present the implementation and testing of our proposed method using two evaluation methods. The final chapter is the conclusion and future works.

2 Theoretical Background

In this chapter, we present the theoretical background of building extraction technique, in sec. 2.1, we present what is building extraction and how it is done and we present some proposed approaches for building contour extraction. Then in sec. 2.2, we review ACM model which is one of the most used techniques in building extraction, finally we discuss some other improved ACM models that overcomes traditional ACM shortcomings.

2.1 Building Extraction from Aerial images

Automatic extraction of objects in aerial images is a hot topic in the field of remote sensing applications. New remote sensing applications were recently found after the availability of high-resolution, obvious and detailed aerial images, because it increased the abilities of many analysis applications such as urban planning, military strategies, geographical and environmental analysis, disaster management and Geographic Information Systems (GIS).

Many of remote sensing applications use features (objects) extraction from aerial images and link it with the related data to make analysis applications. Buildings are one of the important features to be extracted in remote sensing applications due to its importance in remote sensing data interpretation. Generally, building extraction from aerial images is the process of building detection and reconstruction, and achieving this process manually, consumes long time and large human effort.

The output of the proposed building extraction techniques differ in level of the resulted information, for example some low-level techniques detect only the locations of the building objects on the aerial image. The higher-level techniques provide the user with an estimated shape or rectangular box of buildings. Approaches that are more detailed extract the exact accurate shape of buildings, and give an accurate data of shape area and some approaches find the height of building.

However, automating building extraction process is very important for many applications it can be a challenging process for many reasons.

2.1.1 Building Extraction Challenges

Building extraction from aerial images conclude many challenges. One of these challenges is the quality of aerial image. Aerial image quality is judged by set of elements, one of these elements is the *resolution of image*, the availability and evolution of high-resolution quality images enables more detailed and higher level of remote sensing applications. Another important element of image quality is *noise*, which must be processed before extraction to improve the extraction results. In addition, the aerial image must be captured in good weather and lighting conditions to get better feature extraction results [1].

Usually in building extraction approaches image should be preprocessed to enhance image quality before extraction, but the operators of processing techniques must be appropriate for the image conditions. For example, removing noise with $n \times n$ filter window with window size n=5 may be more suitable for noise removing than n=3 filter, also in edge detection process, choosing a low threshold value ensures that we capture the weak but meaningful edges in the image, but it may also result an excessive number of unwanted edges such as edges caused by noise. In the other hand, too high threshold will lead to lines fragmentation, which may also represent significant building's contours. Figure 5 shows application of Canny edge map with threshold set to 0.15, then with a threshold set to 0.2 to show the difference.

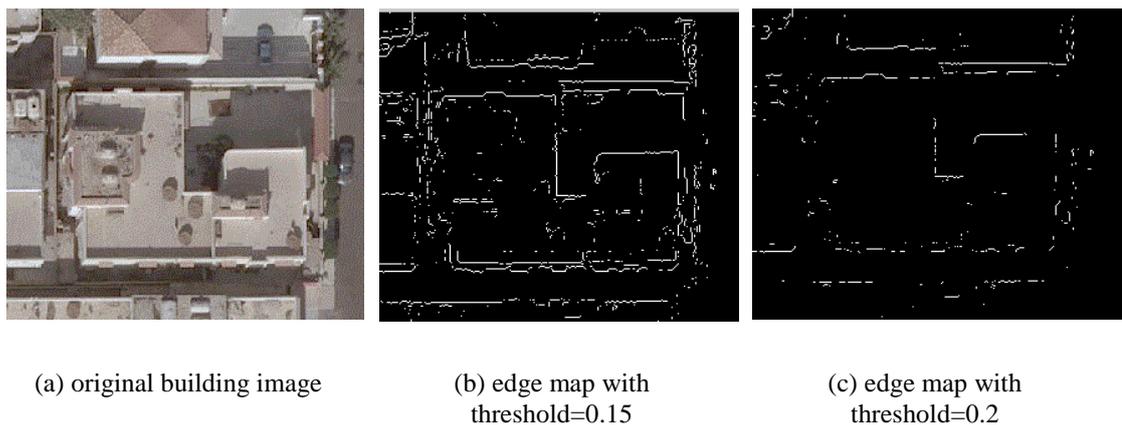| (a) original building image | (b) edge map with threshold=0.15 | (c) edge map with threshold=0.2 |

Figure 5: Effect of edge map threshold on edge map

Beside the mentioned challenges of aerial image quality, there are more challenges encounter building extraction, because buildings are usually close to each other in urban areas and many other objects are close proximity such as trees, vehicles, parking areas,

and power lines. Another affecting factor in buildings extraction is shadow, which affects contour extraction accuracy and leads to false objects detection, but Sirmacek and Unsalan [1] proposed to use colors and shadow information for building detection, where they used color invariant features to extract red rooftops and verify buildings existence by shadow information.

Another important factor in building extraction is the characteristics of environments, because buildings in some environments have similar characteristics such as rectangular buildings shapes and color of rooftops. Therefore, some approaches proposed to utilize this similarity of buildings characteristics to extract building objects, as Sirmacek and Unsalan proposed the red rooftop color.

Extracting buildings in different environments showed up different challenges, where in some environments, buildings have distinct structures and roof colors, and buildings have unstructured shapes, also in some environments buildings are located in complex unstructured urban areas. In general, buildings have different structures and roof colors, so these approaches cannot be generalized.

For the mentioned challenges, building extraction from aerial images became a research area and researchers proposed different approaches to overcome these challenges. We present different approaches of building extraction in section 2.1.2.

2.1.2 Building extraction approaches

The recent approaches of building extraction depended on different image processing techniques such as image segmentation, edge detection, and corner detection. The proposed techniques also differ in level of the extracted information, where some techniques intends to detect building's location only, a higher-level approaches get the exact building boundary (contour), and more detailed approaches intend to get detailed information from the extracted information such as area and height of buildings.

Most of proposed approaches in building extraction field are generally divided to automatic and semi-automatic approaches. In this section, we show some examples of each type.

Automatic building extraction

In recent works, researchers proposed many full automatic approaches that excludes user support in the extraction process.

Zwick and Saeedi [3] proposed to use orientation and magnitude to enhance edge detection then group detected lines/segments to form building hypothesis. After that, they proposed to apply multi levels for hypothesis verification and removing pools and green areas without any user intervention. Another different approach have been proposed by Park et al. to extract rectangular boundaries from aerial image using Centroid Neural Network (CNN) which is an unsupervised competitive learning algorithm based on the classical k-means clustering algorithm. The applicability of a CNN algorithm is to connect corrupted line segments into completed straight lines [4]. Baluyan et al. proposed a method based on a combination of k-means machine learning clustering technique and support vector machines (SVM). To achieve this, a two phases process is used, the first phase is a trained SVM to distinguish between rooftop and non-rooftop regions. The second phase is to use histogram algorithm to detect rooftops which were missed in the first phase [5]. Theng et al. proposed an improved ACM snake model with a cornered-radial cast model instead of centered-radial model depending on Haris Corner detector method to select an initial corner point to locate the initial snake position to automate the full extraction process.

In general, automatic building extraction approaches are complicated and take more processing time and suffer from a new challenge which is the number of unwanted extracted objects "false positives".

Semi-Automatic building extraction

The semi-automatic building extraction approaches require human support in one extraction step or more. Some approaches require user support in locating or detecting building objects and they continue contour extraction process. Other approaches require user support in extracted objects verification.

For example, Juliano and Porfírio [6] proposed to use ACM snake model in building extraction. The extraction process is started up by a human operator, which supplies an approximate polygon using a snake model that is optimized by dynamic programming. After that, the extraction process is performed and polygons describing the building roof contours are obtained as results. A similar approach have been proposed by Mayunga et

al. to use radial casting algorithm to initialize snakes contours. A human operator measures a single point at the approximate center of the building in the image space, then snake points along the contour are automatically generated and the fine measurements of building outlines is carried out using snakes model. As soon as the snake points are generated, the user has an option to accept or reject the snakes contour [7]. As we can see in their approach the human support is required in two different steps in initiating snakes and in results verification.

Although semi-automatic building extraction approaches requires more extraction time and human effort, it 's results are usually more accurate and get less "false positive" results. In our thesis, we aim to use a semi-automatic building extraction approach to test improvements on ACM Model.

In the next section, we present ACM model, which is one of the most used building extraction methods because it is a useful model to extract structured and unstructured objects from aerial images in informal settlements.

2.2 Active Contour Model (ACM)

Active Contour Model (ACM) is one of the most used techniques in building extraction. ACM also is called snake, was first presented by Kass et al. (1988) and defined as an energy minimizing deformable spline influenced by constraint and image forces that pull it towards object contours. ACM is deformed due to **external forces** that attract it towards salient features of the image and **internal forces** that try to preserve the condition of smoothness in the shape of the curve [2] .

2.2.1 ACM theoretical background

ACM is used to find object boundaries regardless of shape, it is applied in many approaches, such as detecting edge segments then link them together. Another approach is initiating a smooth curve near the target and refined until extract object boundaries. Figure 6 shows the idea of the initial snake movement.
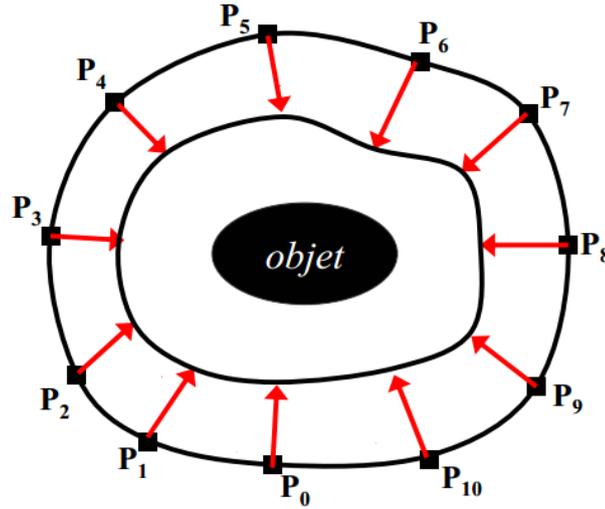
Figure 6 Active Contour Model (snakes)

The refinement process of snake movement depends on minimizing *Internal Energy* and *External Energy*, this occurs when the snake's control points reach the accurate object boundaries. The energy function of snake found with the equation:

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

<div align="right">Eq. 1</div>

Where E_{int} is the internal energy due to bending and elasticity, and serves to impose smoothness constraint. E_{ext} is the external constraint force that attracts the contour to feature of interest in the image, and v is the vector of snake contour.

The **External Energy** is minimized when the snake close to the object boundary position, it is computed by the gradient of the image [8], and found by the equation:

$$E_{external} = E_{image} + E_{con}$$

<div align="right">Eq. 2</div>

E_{image} is image forces pushing the snake toward image features (lines, edges, etc.), E_{con} is the external constraints are responsible for putting the snake near the desired local minimum, it may come from higher level interpretation, or user interaction.

The **Internal Energy** is minimized when the snake have a shape which supposed to be relevant to the shape of the sought object, it is related to curvature and continuity of the object's contour, and found by equation :

$$E_{internal} = E_{cont} + E_{curv} \qquad\qquad \text{Eq. 3}$$

Where E_{cont} denotes the continuity of the snake and E_{curv} denotes the curvature degree of snake, and defined as follows:

$$E_{cont} = \alpha(s)\,|\,v_s(s)|^{\,2}$$

$$E_{curv} = \beta(s)\,|\,v_{ss}(s)|^{\,2}$$

$$E_{internal} = (\alpha(s)\,|\,v_s(s)|^{\,2} + \beta(s)\,|\,v_{ss}(s)|^{\,2})/2 \qquad\qquad \text{Eq. 4}$$

Minimizing $\alpha(s)$ will make the energy function insensitive to the amount of stretch, and small values of $\beta(s)$ will make the energy function insensitive to curves in the snake [9].

An implementation of the traditional snake through a discrete formulation will be explained later in greedy snake subsection in the next section.

2.2.2 ACM Implementation

The overall object extraction process using ACM done through three steps: image preprocessing and enhancement, edge detection and contour extraction process.

Image Enhancement

To get better building extraction results, an image enhancement process must be done before, it's done by processing image so that the result is more suitable for many application such as sharpening or de-blurring an out of focus image, highlighting edges, improving image contrast, or brightening an image and removing noise [10].

The noise in aerial images obstructs the snakes contour extraction process, because snakes are highly sensitive to high gradient noise. So the image should be smoothed to reduce the noise by preprocessing or blurring steps.

Edge detection

Edge detection is the process of identifying and locating sharp discontinuities in an image, discontinuities are abrupt changes in pixel intensity, which characterize boundaries of objects [11]. Different edge detection techniques were approved

depending on changing operators determined by types of edge characteristic and direction in which it is most sensitive to edges.

Edge detection process is very sensitive to noise, because both the noise and edges contain high-frequency content changes. Attempts to reduce the noise result in blurred and distorted edges. Some edge detection operators are:

- Canny edge detection,
- Prewitt edge detection,
- Sobel edge detection, and
- Laplacian edge detection.

As we mentioned previously in building extraction challenges in section 2.1.1, when we use one of these edge detection approaches, operators must be used carefully with a measured threshold because it affects the extraction results.

One possible way to extract edges is 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 kernel G_σ, gives the following image energy term

$$E_{img} = -\nabla |G_\sigma(x, y) * I(x, y)|$$
Eq. 5

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

Object contour extraction

After image preprocessing and edge detection steps, the object contour finally is ready for extraction. The snake model start moving by the external force effect toward image features which is the detected edges or lines as shown in Figure 6.

In each snake iteration, each snake points calculate the total energy of snake points for $n \times n$ neighbors window size, internal and external energies are calculated in terms of eq. 2, and 3.

Then it calculate total energy for all n neighbors in terms of eq. 1 and move the snake point to the location of the minimum snake point and repeat this step for all snake points.

The snake keep moving until a certain condition, either a fixed number of iterations, or a percentage threshold of a moving snake points. The final snake vector expresses the extracted object's contour. The described model is a continuous description of snake model, and the Greedy snake model is the discrete formulation of this model.

Greedy Snake

Greedy snake algorithm works like an elastic band being stretched around an object and then being released, it makes locally optimal choices, hopes that the final solution will be globally optimum. Energy function for each point in the local neighborhood is calculated, then point is moved to the next point with lowest energy function. This process is repeated for every point and iteration is done until termination condition met [12].

In the discrete formulation of greedy snake model, the contour is represented as a set of snake points $v_i = (x_i, y_i)$ for $i = 1, 2, ..., n$ where x_i and y_i are the x and y coordinates of snake point respectively and n is the total number of snake points. The greedy snake model is based on the definition of an energy function where:

$$E_{snake} = \sum_{i=1}^{n}(E_{int}(v_i) + E_{ext}(v_i))$$

Eq. 6

$$E_{snake} = \sum_{i=1}^{n}(\alpha E_{cont}(v_i) + \beta E_{curv}(v_i) + \gamma E_{ext}(v_i))$$

Eq. 7

Same as ACM model, E_{int} is the internal energy force, E_{ext} is the external force that attracts the contour to features of interest, E_{cont}, E_{curv} are the continuity and curvature of the internal energy force.

Greedy snake called greedy because it works as greedy algorithm, it looks for optimal solution locally only and it computes the movement of each snake point by looking at the neighborhood of pixels around the snake point and then move this snake point to the position in the neighborhood which minimizes the energy term [13], Figure 7 show the greedy snake external energy effect on movement.
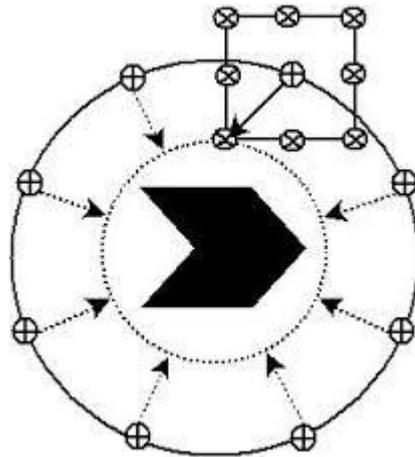
Figure 7: Greedy algorithm external energy minimization

Minimization

The algorithm minimizes the energy of each point individually. For a given point, it looks at a window of $n \times n$ given size centered on the starting point. For each of the points within that window, it computes the value of the energy functional at that location. It then looks for the point that minimizes the energy and updates the coordinate of the point within the spline to that location, Figure 8 shows how the greedy algorithm will move from pixel another to reach a new location of lower energy.
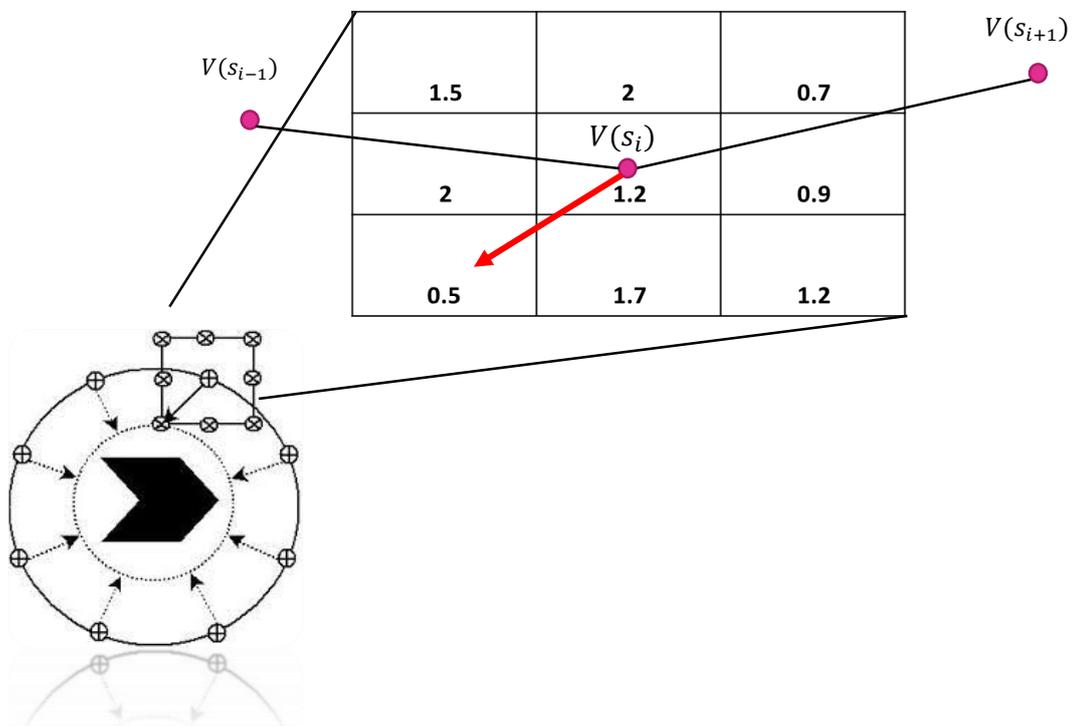
Figure 8: Greedy snake neighborhood energy minimization

Relaxation (Drawing Corners)

In this step, the algorithm loops through all the newly updated points, calculating the normalized curvature at each point, then looks for local maxima if the curvature and the absolute image energy at one of those points is higher than a given threshold the point is assumed to be a corner and is relaxed by setting its weight β to 0 to prevent the corner from being moved due to its high curvature energy.

Varying energies coefficients

Based on Eq.7, choosing different coefficients values dependent on feature to extract:

- Set α high if there is a deceptive Image Gradient.
- Set β high if smooth edged Feature, low if sharp edges.
- Set γ high if contrast between Background and Feature is low.

The following pseudo code summarize the greedy snake algorithm

---

Repeat

    For each snake point p_i

        For neighbors n of p_i

        - Calc. $E_{int}(n_k), E_{ext}(n_k)$ for n neighbors of $p_i$then find $E(n_k)$

        Next n

        - Find n_k where $E(n_k) = min_{k=1}^{8}(E(n_k))$

        - If $E(n_k) < E(p_i)$ then

            o move p_i to n_k

      - if $E_{curvature}(n_k) > threshold$

            o set β(n_k) =0

    Next p_i

Until (x% of snake points are not moving any more)

Figure 9: Greedy snake pseudo code

2.2.3 Shortcomings of Traditional ACM Model

- The result contour is strongly dependent on the initial position of the snake [14].
- Converging to noise [14], snake is so sensitive to noise because noise sometimes have high gradient values that stop snake converging before reaching real object contour.

- Weak edges leakage, when edges got low gradient changes the snake does not stop converging on these weak edges.
- Cannot converge inside deep narrow concavities [15], as shown in Figure 10-a, (we aim to solve this problem in our thesis).
- The snake is sensitive if it inside or outside the object.

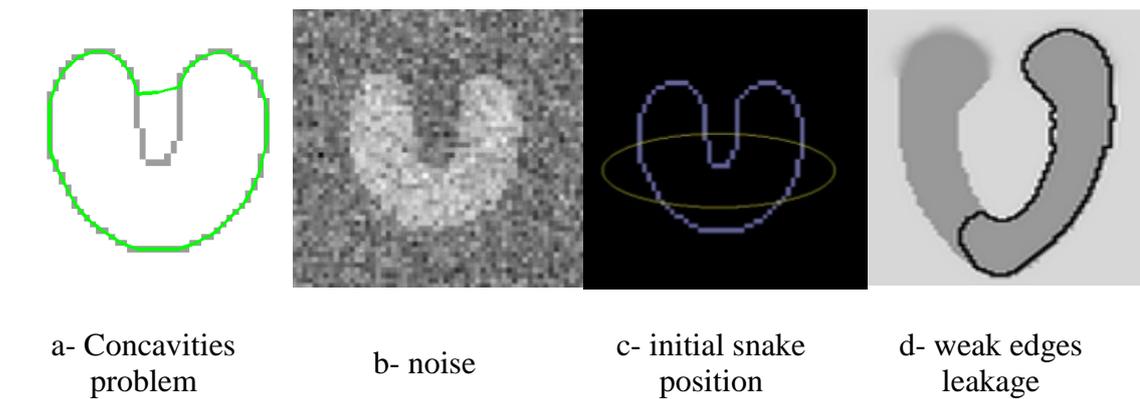Figure 10 shows the main shortcomings of the traditional Active Contour Model that defined by Kass in [2].

| a- Concavities problem | b- noise | c- initial snake position | d- weak edges leakage |

Figure 10: Traditional ACM model common shortcomings

According to the mentioned shortcomings of the traditional ACM Model many approaches where proposed to overcome these shortcomings, in the next section we review the most used improved methods of traditional ACM model.

2.2.4 Improved ACM Models

In this section we review two of the improved ACM models that have been mostly used in feature extraction applications.

Gradient Vector Flow (GVF) Snake

Gradient Vector Flow (GVF) is proposed by Chenyang Xu et al. in [16] to solve the problem of boundary concavities and weak edges, GVF is computed as a diffusion of the gradient vectors of a gray-level or binary edge map derived from the image and used as an external force of snake and called a *GVF snake*.

The advantages of using GVF in snake model are:

- The snake become insensitive to the initial position of snake, if it is inside or outside object.

- The ability to move into boundary concavities.
- The large capture range, which means that, barring interference from other objects, it can be initialized far away from the boundary.

The advantage of using the diffusion of the gradient vectors in snake converging into concavities as shown in Figure 11, which shows how snake can converge inside U-shape.

Although the mentioned advantages of GVF-snake, it is difficult to realize accurate segmentation when detecting complex shape object with deep concavities, so that some researchers introduces improvements on GVF to overcome these challenges. Luo et al. presented an additional balloon force to the GVF-snake to prevent the active curve from trapping into local minima [17], also Zhu proposed a nonlinear filtering on GVF force field for further enlarge capture range [18].
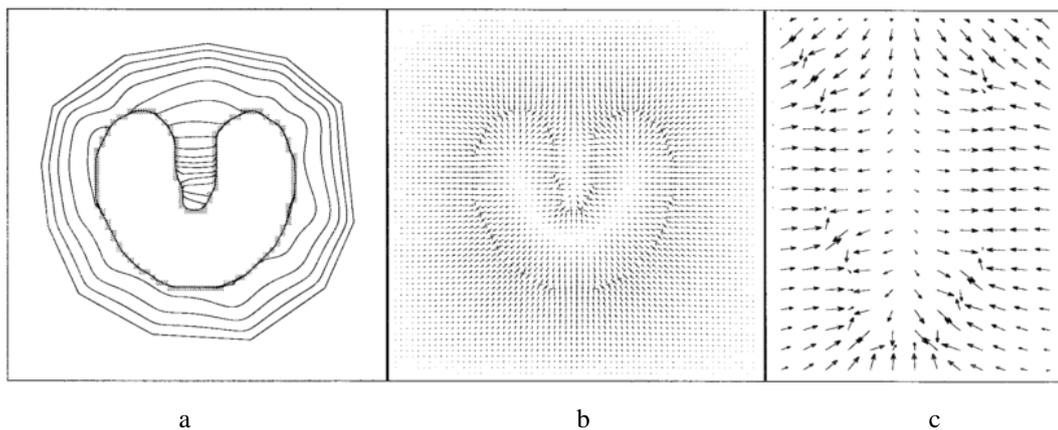
Figure 11: (a) Convergence of a snake using GVF (b) GVF external forces, (c) close-up within the boundary concavity [16]

ACM Balloon snake

Balloon snake is a new Active Contour Model proposed by Laurent Cohen in his paper [19]. The main idea of balloon snake works as the same principles of Kass snake model, but it make the curve behave like a balloon by adding an additional inflation force to enable snake converge when snake is not close enough to contours or when snake stopped by noise or weak edges. The new balloon force inflate the initial oriented curve with a pressure force that pushes the curve outside as if we introduced air inside balloon. Cohen suggested adding this F inflating force to the previous snake forces where:

$$F = k_1 \vec{n}(s) - k \frac{\nabla P}{||\nabla P||} \qquad \text{Eq. 8}$$

Where $\vec{n}(s)$ is the normal unity vector to the curve at point $v(s)$ and k_1 is the amplitude of this force and P is the image force (the potential term) that pushes the curve to the significant lines which correspond to desired attributes. If we change the sign k_1 or the orientation of the curve, it will leave an effect of deflation instead of inflation. k_1 and k are chosen where k_1 is smaller than pixel size and k is slightly larger than k_1, so an edge point can stop the inflation force, if the edge is too weak the curve can pass through weak edges by the effect of the pressure force.

ACM Balloon snake advantages:

- Balloon snake modify the external image forces to prevent pulling snake to high gradient regions far away from object's contour.
- Snake should not be close to boundary, because balloon increase the capture range of external force field and make the placing of initial curve easy.
- The snake become less sensitive to noise because snake converge to strong detected edges not to broken small segments of edges.
- The snake can be initiated inside the object and snake can converge outside to the object boundary.

3 Related Work

This chapter reviews some related works to building extraction using ACM model, in section 3.1 we review some Automatic Building extraction techniques from aerial images using different techniques other than ACM model, and then in sec. 3.2 we review some approaches that used the ACM model in building extraction from aerial images.

3.1 General Building Extraction Approaches

Zwick and Saeedi [3] used orientation and magnitude to enhance edge detection, using Burns Line edge detection approach to detect straight segment edges and group these lines/segments to form building hypothesis. After generating hypothesis, they proposed applying multi levels for hypothesis verification process to remove pools, green areas, and unwanted extracted objects. They opposed on using Canny edge detector for small and nonobvious buildings, but they used it in building hypothesis initialization step. The results of this approach extract a quadrated buildings contour and didn't detect the exact contour of the building shape as shown in Figure 12, so it is not a perfect solution for concavity contours extraction.

Figure 12: Zwick & Saeedi approach results [3]

Zuam and Muller in their research [20] proposed to use a seeded region growing algorithm by distributing center points every 15 pixels in aerial image, then adding neighborhood pixels to groups and similar groups (regions) are merged together. Then they extract some numeric features such as area, Hue angle, and circumference value,

and some structured features such as neighborhood (buildings usually stand in groups not alone) and shadow. All these features are extracted to ensure building hypothesis. They depended on red channel groups to form building hypothesis and remove non-red regions, this approach will not be suitable for other environments, also they proposed to remove non-straight shadow hypothesizes which may remove true hypothesis that affected by noise or trees on shadows.

In [1] Sırmacek and Unsalan rely on red rooftops for building detection, and they use color invariant to segment out red rooftops with Otsu's thresholding method. Then they use illumination direction and the center position of the shadow region for non-red rooftops to estimate the possible building location. Finally, they used Box-Fitting approach to determine buildings shapes, this approach start with detecting possible corners for line segments then choose a starting corner, which have the smallest value of $|\pi/2-\beta|$, where β is the angle between lines for each corner. The box start from the specified starting corner then expands to reach minimum value of E, where E is the sum of distances between edges and box contour.

The proposed approach in their research was specified for a small type of buildings that have red rooftops and rectangular shapes and reject not rectangular buildings, also concave regions inside buildings will prevent the expanding box from reaching the real contour, so it is not applicable for irregular building shapes.

Bhadauria et al. in [21] proposed to generate buildings hypothesis by using edge detection and line generation methods. Frist they used Canny edge detector to extract edge pixels, then used Hough Transform to generate line information from edge points that generated from previous step, in hypothesis generation they depend on threshold values for length of line and distances between lines. Finally, they aimed to support their hypothesis generation with producing color segmented image using K-Means algorithm. In this paper, only rectilinear shaped buildings are considered, so hypothesis generation stage search for rectangular structures in the image. Also, all the building detection rate is associated with adjusting threshold parameters in hypothesis generation step, such as by lowering the threshold value more buildings can be detected in the image but a lower threshold also catches more noise in the scene which will result in false positives.

Park et al have proposed another different approach in [4] to extract rectangular boundaries from aerial image using Centroid Neural Network (CNN), which is an

unsupervised competitive learning algorithm which based on the classical k-means clustering algorithm. The proposed method is to use a likelihood metric designed particularly for line segments in conjunction with the CNN algorithm to connect corrupted line segments into completed straight lines as shown in Figure 13. The used metric for line segments measures the likelihood that two given line segments belong to a single linear structure.
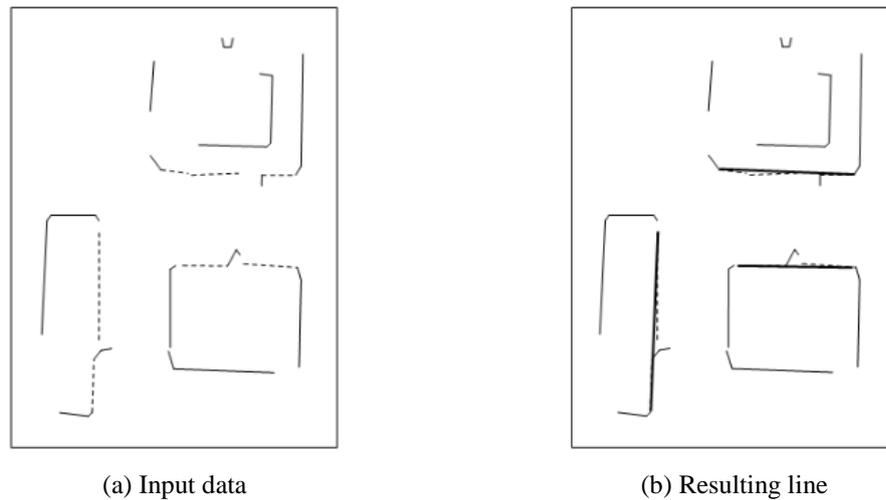

(a) Input data   (b) Resulting line

Figure 13: Examples of line segments grouped by CNN with metric measure of line segments [4]

Using this approach, noisy elements in the image can be reduced and the detection of rooftops can be achieved more accurately, but depending on the CNN may extract strong contours for other unwanted objects and it will omit weak edges of some buildings.

Baluyan et al. proposed a method based on a combination of k-means machine learning clustering technique and support vector machines (SVM). To achieve this, a two phases process is used, the first phase is a trained SVM to distinguish between rooftop and non-rooftop regions. Moreover, the second phase is the histogram method to detect rooftops which were missed in the first phase [5].

However, using k-means clustering depends on the choice of an appropriate value of k (the number of clusters), and a range of values must be tested to set k value, and it get poor performance when many different rooftop colors are encountered. In addition, the SVM cannot separate buildings that are proximity closed.

3.2 Building Extraction using ACM model

Salman Ahmady et al. in their paper that published in 2008 [22], proposed a full automatic building extraction method that deal with full aerial urban image that contains multi objects as an initial input. Their proposed method used an improved type of ACM model called Chan-Vese model that based on Mumford-Shah image segmentation method. In Chan-Vese method, image is segmented to regions, where each region have high similarity. As a testing method, they used the Mckeon's shape accuracy factor for evaluation, in this evaluation, the relation of the area of buildings in ground truth is compared with the area of buildings that is detected by model, where:

$$Shape\ accuracy = 1 - \frac{|A-B|}{A} * 100$$

Eq. 9

Where A is the area of a building in ground truth. And B is the area of its corresponding detected building.

Depending on Chan-Vese active contour model, give the ability to extract objects without obvious edges from images and decrease the sensitivity to noise.
However, there are two major disadvantages of the full automatic process of their approach, firstly the automatic generation of a series of regular circles as initial snakes consumes large processing time, secondly this cause detection of other feature objects that have similar spectral information buildings.

Kovacs and Sziranyi in [23] proposed to improve Harris Corner detector to detect more corner points in aerial images to support contour detection, then to extract directions of feature point's neighborhood, after that Shearlet method is applied to strengthen edges in the defined direction. Finally, they used Chan-Vese active contour algorithm to initialize the contour for buildings candidates. The researchers tried to improve edge detection by increasing the feature points set, this will increases the detected objects which also include the detection of not buildings objects.

Antonio and Aluir in their paper [6] proposed to use Dynamic Programming (DP) algorithm for it's advantages as an optimization algorithm such as reducing number of required operations to find set of optimal variables for the energy function, also they proposed imposing constraint variables where building contours are polygons so corners

will be applied to the energy function. DP is applied for optimization based upon two main steps. Firstly, new edge points are sampled by using information derived from the set of edge points optimized in a previous iteration. At the beginning of the extraction procedure, this set defines the approximate polygon provided by the human operator. Second step, apply DP algorithm to select the best set of edge points currently describing a building roof contour that results a minimum energy value. These steps are sequentially repeated until the goal function is minimized, that is, no more vertices are added to the current building roof contour.

The results showed that depending only on edge segments as an external force cues extraction of false objects or inaccurate roof contours, because of noise affect. Also they didn't solve the snakes convergence disability to weak edges and it was not tested to extract concave regions.

Lau Theng in his paper [24] proposed to improve the radial casting active contour model by a circular casting rather than radial casting. Firstly to detect a corner point of the building contour using Haris corner detector (Haris and Stephens 1988), then a circular cast is initiated from any of the first found corner points as a control point. Finally, it start checking pixels within the cast's boundary, and the iteration stops when active contour locks a building outline.

Theng's proposed model solved the problem of determining the number of the radial lines wanted to detect the complex shapes of building contours accurately, also solved the problem of cast initialization that causes deletion then rebuilding control points, but a predicted problem will be encountered when an internal strong edges is found which will stop the snake movement.

The idea of Kabolizade et al in [25] is optimize internal and external forces coefficients using Genetic Algorithm (GA). The first initialization of candidate buildings is done automatically using Digital Surface Model (DSM). The basic idea of using a DSM in a building extraction method is that the man-made objects with different heights over the terrain can be detected by applying a threshold to DSM. After that, GA is applied to optimize internal and external forces coefficients in three steps, first step is to define array of variable values in 30 Chromosome. Then in mutation step, the chromosome with lowest cost will have high probability of mating. Finally, parents are selected according mating and offsprings of next generation are made.

After convergence of GA algorithm, the weight coefficients of snake were extracted from the best chromosome. Then buildings are extracted using GVF snake model and calculated coefficients.

The peculiar in Kabolizade paper is initiating the snake automatically using the DSM model and decreasing the human intervention, and the external energy function consisting of building edges and surface consistency makes the snake model more stable to converge to the true building contours, but this approach did not mentioned how it operates with concavity contours.

Researchers in [26] used Balloon ACM model to identify approximated building regions without human intervention depending on edge lines as an external force. Their approach done in five steps, in the first step they applied edge extraction using Burns line edge detector to use detected lines later as an external force for snake movement. After that, close edges to balloons are selected, this increase building hypothesis exist. Then buildings hypothesizes are initiated with a proposed rules for some predicted building shapes such as: F and U shapes, and to prevent other shapes such as Z and H shapes. Finally, hypothesis verification step is done by finding hypothesis value, which is sum of pixels needed to connect line components of building hypothesis.

The results of this approach showed that this approach fails in some cases where external force (line segments) is not enough to pull balloons or stop growing of balloons to contours, also internal force was - in some cases - not enough to extract complicated shapes (such as concave regions). In addition, grouping detected line segments to generate building hypothesis is sometimes not enough for extraction accurate building contour.

Overall, the presented approaches used different techniques for building extraction, and the quality and level of result information is different from one approach to another. Approaches vary in quality of extracted shape, as we have seen some approaches extracted the location of building with an estimated rectangular shape of building and some others get the real contour of buildings with different accuracy and quality. However, in our proposed method we aim to improve the traditional ACM model to solve the contour concavity problem to get better accurate shape results.

4 Proposed ACM Model

In this chapter, we view our proposed improvements on the traditional ACM model, our improvements are divided to modifications on the ACM model and on the method of applying the ACM model on building contour extraction, we describe each of the improvements in section 4.1, and 4.2.

4.1 Proposed ACM Model Improvements

The idea of the proposed improvement on ACM model is to depend on the effect of the internal and external forces coefficients weights on the snake convergence in concavities.

Figure 14: The effect of α on snake stretch

As Figure 14 shows, Alpha coefficient of the internal energy effects the stretch of the snake, which enables more converging into concavities, but in traditional ACM model, if the internal and external forces coefficients are fixed for all snake control points, and if it is increased for all snake points, it will not converge regularly because of snake rigidity property.

Also we found that the internal curvature force is high and the external energy is low for the snake control point in the concavity regions, as shown in Figure 15.
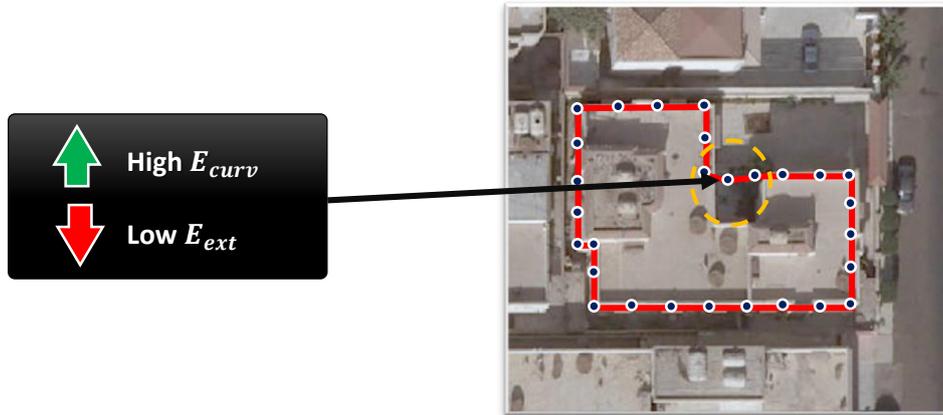
Figure 15: external and curvature energy in concavities

We proposed this study to solve the concavity contours extraction problem by adapting ACM model forces coefficients during snake iterations with adding a concavity index to indicate that snake points stop in a concave region or not, then adapt these coefficients to allow snake converge inside concave regions depending on the new concavity index value.

4.2 Proposed ACM Method

In this subsection we present the steps of the overall approach of building extracting with our adaptive ACM model, Figure 16 shows a flowchart of the overall approach of our improved ACM model.
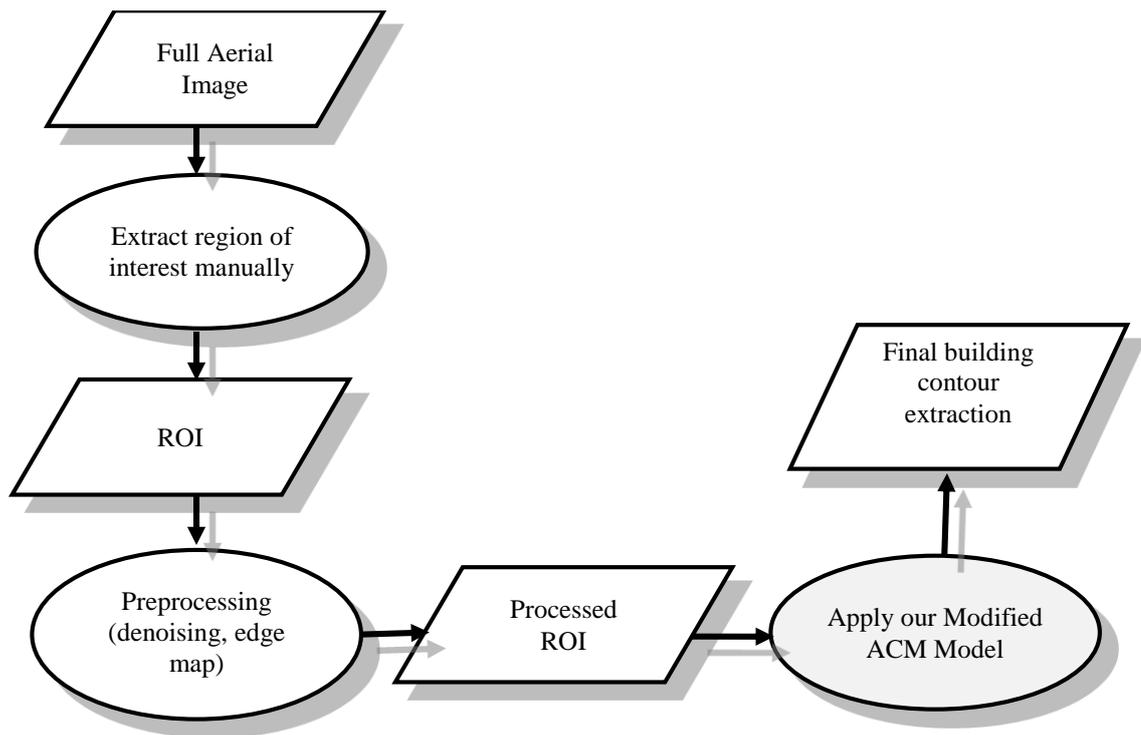
Figure 16: Overall approach for building contour extraction

The used method for building extraction using ACM is the same steps of using the ACM model in other applications, a common step should be done first; such as preprocessing and extracting ROI, the complete method steps are as the following:

Step 1: Extraction of Region of Interest (ROI)

After loading the full aerial image, the user must define the sub image of the building manually, which is the Region Of Interest (ROI), in this thesis; our improved model is applied on ROI, which contains one building that contains concavity contour.

Step 2: image preprocessing

Before applying ACM model the ROI image must be preprocessed, and remove noise and other interrelated objects, then run edge detection process.

Edge detection

Edge detection is the process of finding sharp contrasts in intensities in an image, it reduces the amount of data in the image, while preserving the most important structural features of that image [21].

We propose in our thesis to use Canny edge detector because it's considered to be optimal edge detection operator for building detection as Bhadauria et al. mentioned in [21]. Canny defined optimal edge finding as a set of criteria that maximize the probability of detecting true edges while minimizing the probability of false edges.

The Process of Canny edge detection algorithm done as the following steps:

1. Apply Gaussian filter to smooth the image in order to remove the noise, since edge detection results are easily affected by image noise.
2. Find the intensity gradients of the image, where Canny algorithm uses four filters to detect horizontal, vertical and diagonal edges in the blurred image.
3. Apply non-maximum suppression to thin extracted blurred edges.
4. Apply double threshold to determine potential edges and remove false edge pixels that caused by noise, by comparing edge pixels with high and low threshold value.
5. Track edge by hysteresis: Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

Step 3: improved ACM model

According to the proposed improvements on ACM model in section 4.1 and the effect of internal coefficients, we proposed to solve the concavity regions problem by adapting these coefficients iteratively without adding new external forces. We attend in improving our model on the greedy snake model.

Greedy snake model description

The greedy snake algorithm is the discrete implementation of the ACM model, we have described the greedy model in details on section 2.2, here we summarize the steps to show the modifications later.

Greedy algorithm generally executed throw two main steps:

1. **Minimization process:** for each snake point the algorithm looks in given **n*n** window size, and compute the energy values of the neighborhood pixel points, then move the coordinate of the point to the point that minimize energy.
2. **Relaxation**: The algorithm then loops through all the newly updated points, calculating the normalized curvature at each point, then looks for local maxima.

If the curvature and the absolute image energy at one of those points is higher than a given threshold (the chosen thresholds are: curvature=60 degree and image energy=0.16.) the point is assumed to be a corner and is relaxed by setting its weight β to 0 to prevent the corner from being moved due to its high curvature energy.

The following steps summarize the proposed modified ACM method and Figure 17 shows a flowchart of these steps.

1. **The user must extract region of interest (ROI) from the aerial image manually.**

2. **Apply a preprocessing approach by removing noise with Gaussian filter, then find edge map by Canny edge detector.**

3. **Apply modified ACM model, as follows:**

   - **Do the following, while ($\%X_i$) of snake points still moving ($\%X_i$ is a parameter which the user can change to control snake stopping criteria):**

     - **Calculate internal and external forces of all snake's control point (p_i) neighbors (n_i).**

     - **Calculate total energy E (n_i) for (p_i) and (p_i) neighbors.**

     - **In our modified ACM model, we attend to adapt coefficient of internal force (Alpha) during contour extraction process, depending on a new value called concavity index (ConcIndex), which is a new parameter found in terms of external and internal energies values. Where:**
     $$ConcIndex\ (p_i) = mean(\ E_{ext}\ (p_i))/\ mean(E_{curv}(p_i))$$
     We used mean to prevent division by zero because external force and curvature force matrix values can be zeros

   - **If less than Y_i percentage of snake points still moving (Y_i threshold is set to 40%), then**

     - **check if (p_i) is on contour or not, if not**

       - **recalculate Alpha for Pi where $\alpha(p_i) = \alpha\ (p_i) * ConcIndex\ (p_i)$**

       - **recalculate total energy for (p_i)**

   - **Then, if less than Z_i percentage of snake points still moving (Z_i is set to 10%), do the following:**

- *Do the Insertion/Deletion refining contour shape process of snake points. This process done by adding new snake points and removing unnecessary points by computing the distances between these points until a threshold percentage of snake points still moving, the step of insertion/deletion snake points improves the final extracted contour shape as shown in Figure 18*

- *Finally, do the relaxation step, where if curvature and image energy of (p_i) is more than the given threshold value, then set β to 0.*
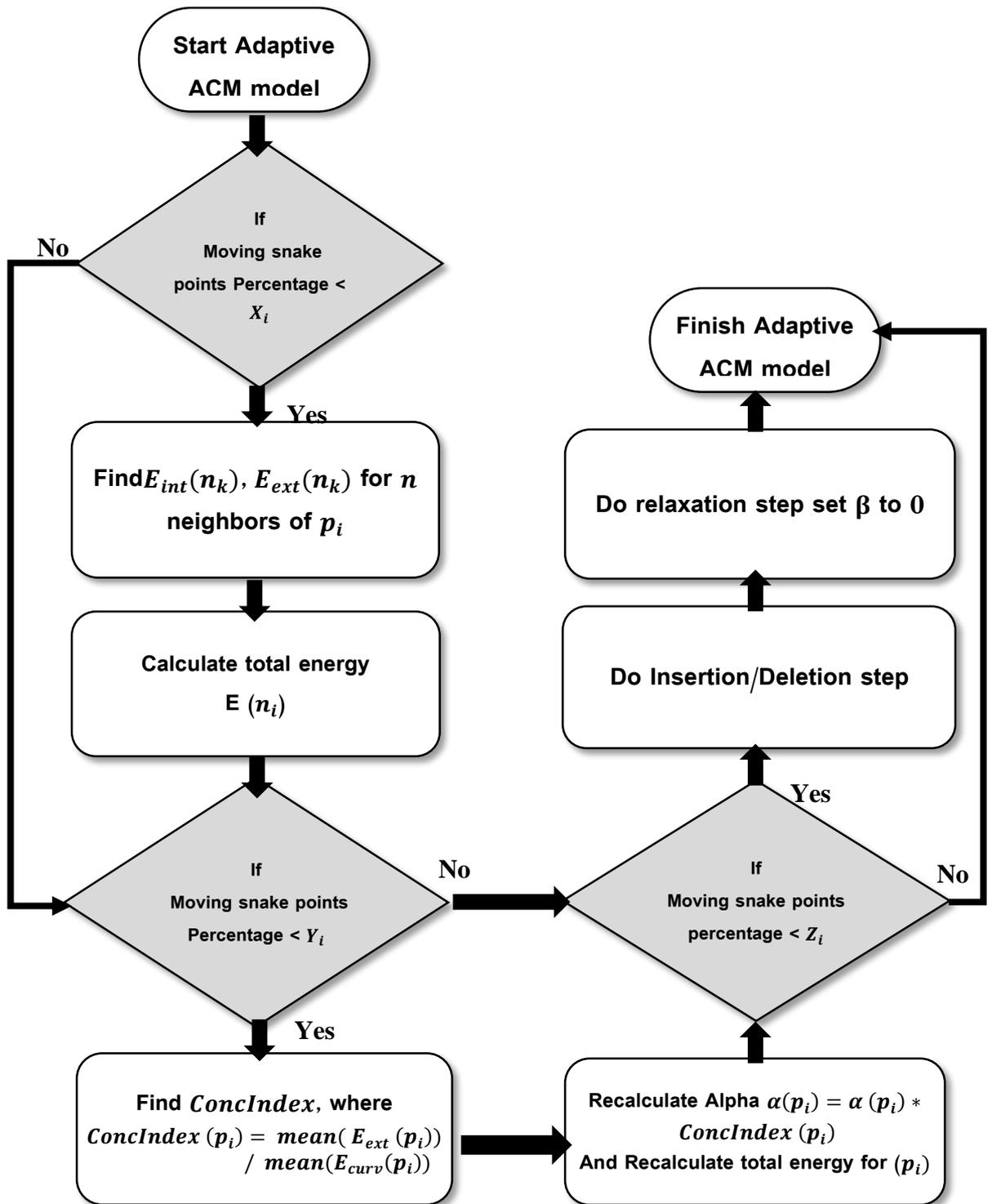
Figure 17: Flowchart of our adaptive ACM model approach

(a) Original building image   (b) Extracted contour without insert/delete points   (c) Extracted contour with insert/delete points
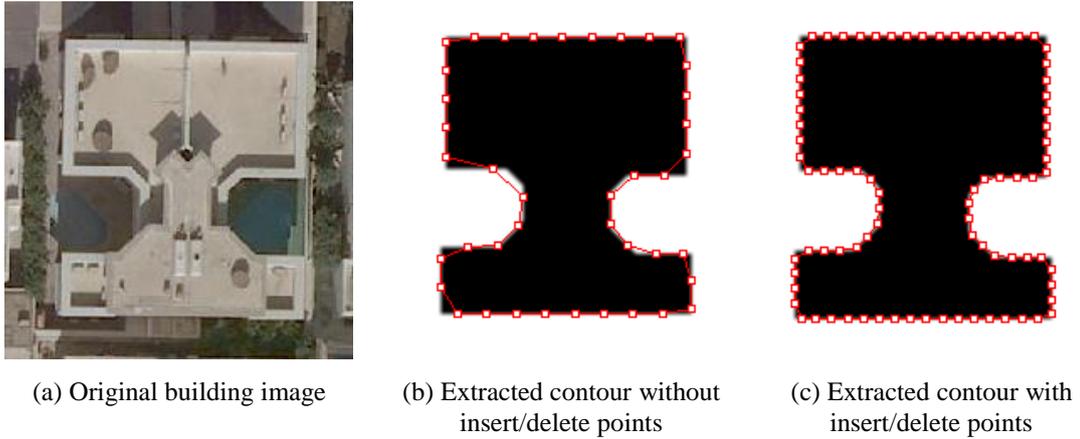
Figure 18: Effect of insertion/deletion snake points process

The following pseudo code describes the overall approach of our improved ACM method.

Repeat

For each snake point p_i

    For neighbors n of p_i

-     **Calculate $E_{int}(n_k), E_{ext}(n_k)$ for n neighbors of p_i then find $E(n_k)$**
-     **Find n_k where $E(n_k) = min_{k=1}^{8}(E(n_k))$**
-     **If $E(n_k) < E(p_i)$ then**
  -         **move p_i to n_k**
-     **Find $ConcIndex\ (p_i)\ =\ mean(E_{curv}\ (p_i))\ /\ mean(E_{ext}\ (p_i))$**
-     **if moving snake points < X% of snake points**
  -         **check if p_i is not on contour**
    -             **Set $\alpha_i = \alpha_i * ConIndex\ (p_i)$**
    -             **recalculate $E_{int}(n_k), E_{ext}(n_k)$ for n neighbors of p_i then find $E(n_k)$**
-     **if moving snake points < Y% of snake points**
  -         **do insertion/deletion snake points to refine contour shape**
-     **if curvature of $E_{curvature}(p_i) > threshold$, then**
  -         **set β = 0**

Next p_i

Until a given threshold percentage of snake points still moving

Figure 19: Pseudo code of the improved ACM method

5 Experimental Results and Evaluation

In this chapter, we present our evaluation and results in three subsections. In section 5.1, we describe the system we have developed and how it is used. Next, we describe the evaluation methods in 05.2.2 and 5.2.3 and present their results in 5.3.

5.1 Implementation

We have implemented a testing application to test our improved ACM snake method using MATLAB version no. 7.11.0. We have designed a graphical user interface to make the testing process easier and faster, this GUI enables the user to initiate a radial snake around the object, and we give the user the ability to merge this initial snake with a spline vector to control the number of the initiated snake points.
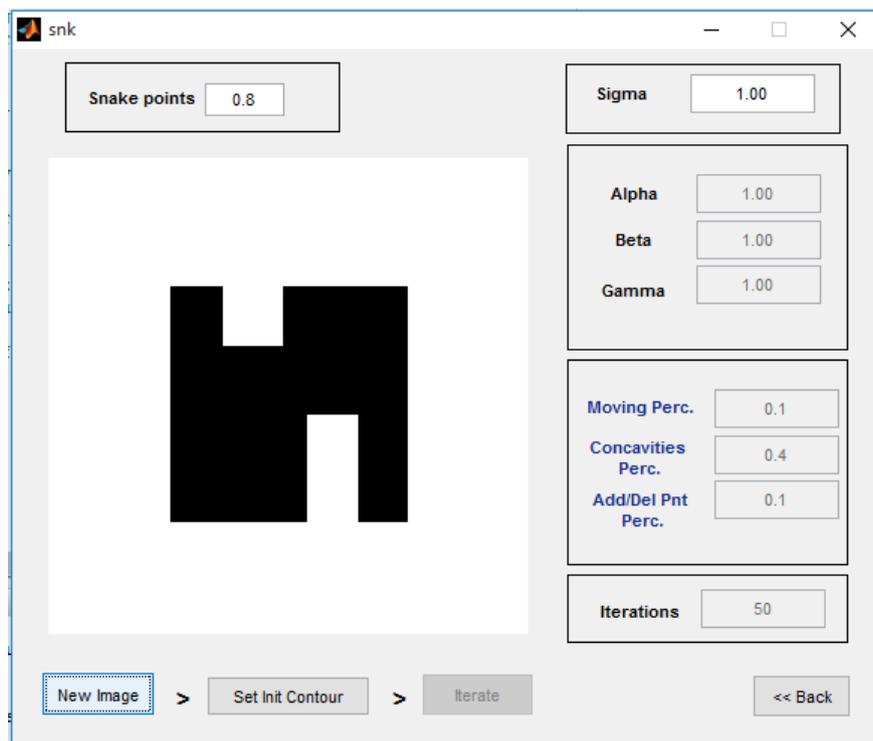
Figure 20: The proposed implemented application GUI

The implemented application is built to apply our improved ACM model and test it in four image sets, the user can choose the input image that must be an 8-bit image, then the user applies the edge detection process. Firstly, the application applies a Gaussian filter to convolve with the image. This step will slightly smooth the image to reduce the

effects of obvious noise on the edge detector, and the user can change the parameter of sigma (σ) to change the effect of the Gaussian filter.

We also give the user the opportunity to control the snake movement criteria, by a percentage value of moving snake points, and it is initially set to 0.1 which means that the snake still moving until less than 10% of snake points are still moving.

The user also can change the converging percentage which is the percentage of the moving snake points criteria before recalculating snake energies after changing the coefficients of these energies in term if concavity index value. Additionally the user can set a percentage of moving snake points before applying insertion and deletion of snake points to refine contour shape. Figure 20 shows our implemented GUI.

After initiating the snake, the user can start contour extraction process. The snake start moving until the final object's contour is detected as Figure 21 shows. The system also find the results of the two testing methods that we explain in section 5.2.
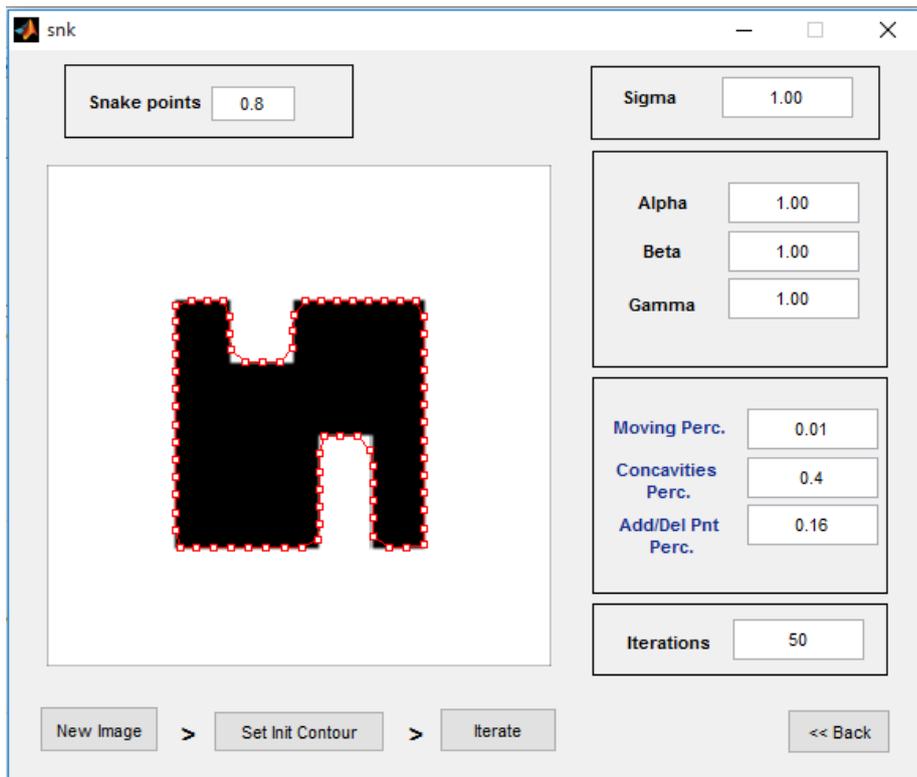
Figure 21: Final contour extraction output example using our application

The following tables, show some results of applying our implemented application on different set of images, these sets will be described in 5.2.1. Table 1 shows some results

of using the implemented GUI to test our proposed method on some images of Set-1, Table 2 shows some results of Set-2 images, Table 3 shows some results of applying our improved ACM model of Set-3 images, and Table 4 shows results of testing Set-4 images.

Table 1: Sample results of applying improved ACM on Set-1
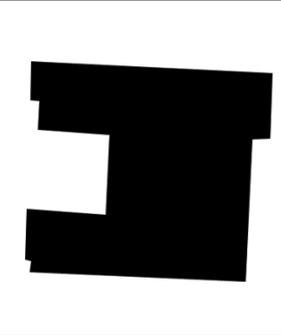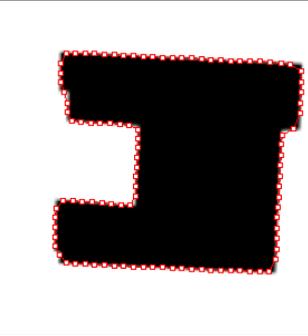
Image	Original ROI image	Processed image	Final image with extracted contour
Img-A-2			
Img-A-3			
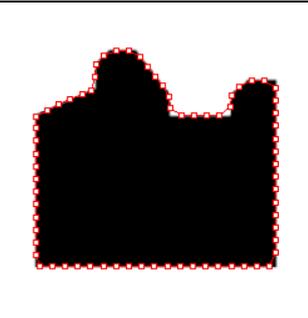
Table 2: Sample results of applying our improved ACM on Set-2
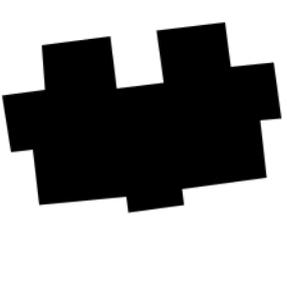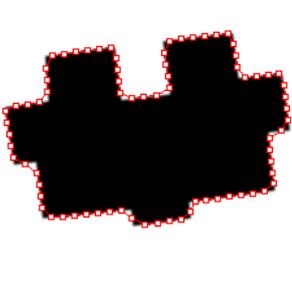
Image	Original ROI image	Processed Image	Final image with extracted contour
Img-B-1			
Img-B-2			
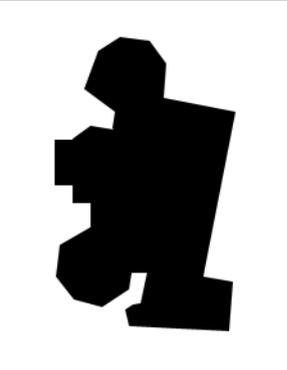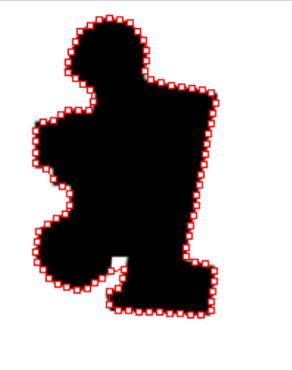
Table 3: Sample results of applying our improved ACM on Set-3
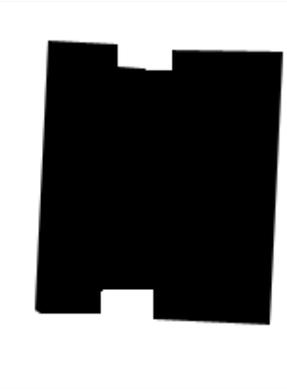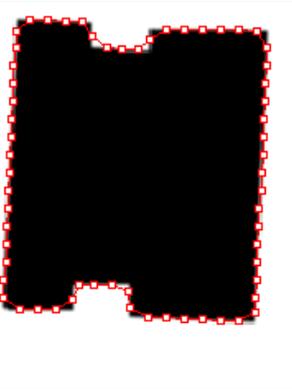
Image	Original ROI image	Processed Image	Final image with extracted contour
img-C-1			
img-C-3			
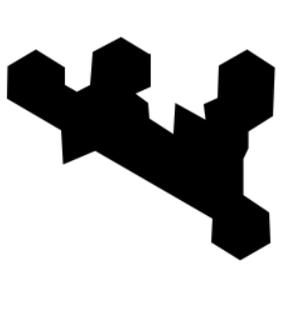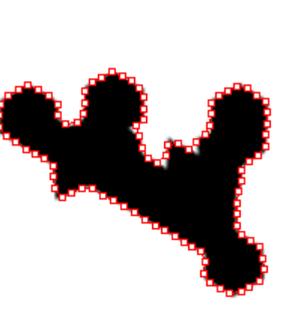
Table 4: Sample results of applying our improved ACM on Set-4

Image	Original ROI image	Processed Image	Final image with extracted contour
img-D-2			
img-D-5			

5.2 Experiments Setup

To test our improved ACM model we have used the implemented application to extract object's contour then use two testing methods to assure the usefulness of our improvements. Method 1 uses Confusion Matrix analyzing method to get accuracy and precision of the results. Method 2 uses an error measure of the total error distances between the resulting contour and the actual object's boundary.

We have also used the testing methods to test GVF snake and compare the results with our improved ACM model. We used an open source code to test the GVF model [27].

As we described previously in section 5.1, the implemented system is designed to find results of two testing methods after contour extraction process. We have executed the experiments by initial energies coefficient values for all image sets to ones, where (Alpha) α =1, (Beta) β =1, and (Gamma) γ=1.

In addition, it is important to mention that the testing laptop machine processor is CoreI3 2.40 GHz speed and 2 GB RAM.

5.2.1 Evaluation Image Sets

The tested set of images is categorized to four different sets of images of buildings cropped from aerial image taken of a complex urban scene extracted from Jaddah Geographic Explorer website[1]. We have tested 30 different images from Jaddah city in Kingdom of Saudi Arabia (KSA). The resolution of the aerial image is 5 cm per pixel, which considered a high-resolution aerial image compared to other aerial/satellite images such as Google Maps aerial images or other. Below we describe each set of images.

Set-1: Images with wide concavities

This set includes images of buildings that contain one wide concavity region in the building to test the performance of our improved model in this case and compare it with the GVF model performance.

Set-2: Images with narrow concavities

The images of this set share the property of containing narrow concavity regions to test the performance of our adaptive model in narrow regions.

Set-3: Images with multiple narrow concavities

The third set includes images that contain multi narrow concavity regions in the same building to test the performance of our improved ACM model and compare it with GVF model performance in this case.

Set-4: Images with custom shapes and concavities

The images of set are custom shapes contain concavity regions to test the performance of our improved model on different custom shapes.

5.2.2 Evaluation Method 1

This testing method uses the Confusion Matrix analyzing method to find the precession and accuracy of the final contour extraction results. **Precision** is the number of *true*

---

[1] http://maps.jeddah.gov.sa/

positives (TP) divided by sum of true positives and *false positives (FP)*, where true positives is the number of points correctly labeled as belonging to the extracted contour and edge map, false positives is the number of incorrectly points that is labeled as a contour but they are not edge map. Also **Recall** is the number of true positives points divided by sum of true positives (TP) and *false negatives (FN).* Where false negatives is the number of edge map points that are not labeled as an contour. The **Accuracy** is the sum of true positives and *true negatives (TN)* divided by the sum true positives, true negatives, false positives and false negatives. Where true negatives is the number of snake points that are not edge map points and are not labeled as contour points. Figure 22 shows a description of evaluation Method1.

- $Precision = TP/(TP + FP)$
- $Accuracy = (TP + TN)/(TP + TN + FP + FN)$
- $Recall = TP/(TP + FN)$
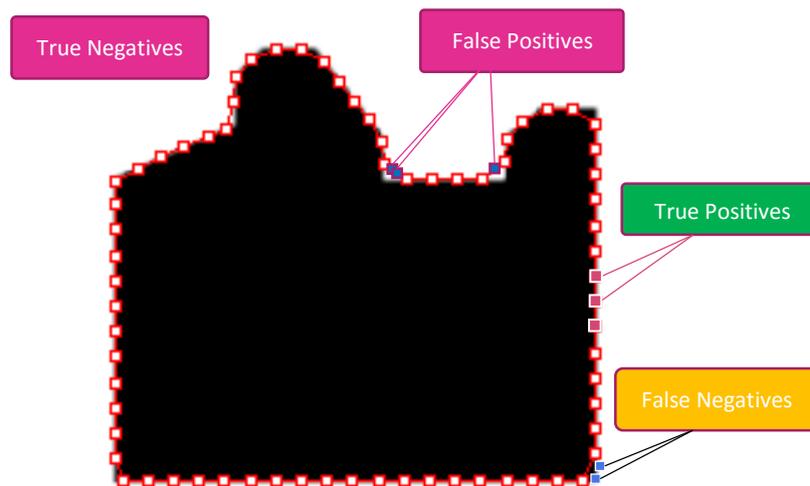
Figure 22: Evaluation Method1 description

5.2.3 Evaluation Method 2

To test our new modifications on ACM model we have used a test method to find the exact accuracy of the final contour shape by finding the ratio of distances between each snake point and the true edge pixels, and the following steps do this:

- Get all the coordinate points between the final snake points.

- For the new generated snake points, we use a window with $N \times N$ of neighbor pixels and using the edge map of the tested image. Then all distances found between this snake point and the edge map pixel with one's values, and get the minimum distance for this point and the neighbor one's pixel.

- Then we get the sum of all the result distances between the snake points and edge map, and find the Error Distance Ratio (ER_d) which is the average ratio of distance between each snake point and the true edge map point (by pixels), using the equation:

$$ER_d = \frac{\sum_{i=1}^{n} d_i}{n} \qquad \text{Eq. 10}$$

Where n is the total number of snake points, and $\boldsymbol{d_i}$ is the distance between snake point $\boldsymbol{i}$ and the nearest object outline pixel.

Testing Algorithm

The following algorithm in Figure 23 describe briefly the testing algorithm to get the ratio of distances for snake points.

- firstly, get all coordinate points for snake
- for each snake point $\boldsymbol{n}$:
    - allocate $N \times N$ neighbors window for this point
    - for all point $\boldsymbol{n}$ neighbors
        - if the edge map value of pixel's neighbors =1
            - Find the distance between point $\boldsymbol{n}$ and the current pixel.
            - Get the minimum distance of these distances
- Find the sum of minimum distances for all snake points.
- Find Error Distance Ratio (ER_d) = sum Of Distances/ points No. , where :

$$ER_d = \frac{\sum_{i=1}^{n} d_i}{n}$$

Figure 23: Improved ACM testing algorithm

5.3 Experiments Results

In the following subsections, we present the process of testing our improved ACM model and GVF snake using the two methods described before, then we present the results of the experiments, finally we compare the results of testing the GVF snake with our improved ACM model.

5.3.1 Results of Experiments

In this section, we present results of contour extraction using our improved ACM model and GVF model using the two testing methods described in 5.2.2 and 05.2.3. GVF is considered as one of the most known techniques that solved the problem of extracting concavity contours [16]. For more details about GVF snake model, you can return to page17.

Results of Method 1:

In the following subsections, we present the results of testing our improved ACM model and GVF model using Test Method 1 that described in 5.2.2.

Our Improved ACM model testing results

The following table shows the results of applying results of testing our improved ACM model using evaluation Method 1 by Convolution Matrix to find precision, accuracy, and recall for 35 different aerial images of buildings and custom shapes.

Table 5: The proposed ACM model evaluation results using method-1

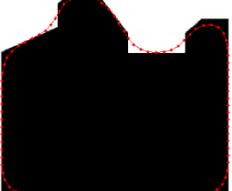| Image	Number of iterations	Testing time in seconds	final snake Points	Precision	Recall	Accuracy %
Image Set1						
Img-A-1	59	15.312	55	0.7511	0.1722	91.15
Img-A-2	119	29.880	100	0.7342	0.2276	98.08
Img-A-3	78	21.4579	70	0.6081	0.1738	98.06
Img-A-4	111	28.006	95	0.6322	0.1756	98.16
Img-A-5	107	24.067	72	0.6838	0.2298	98.89
Img-A-6	143	34.979	90	0.4689	0.1267	98.21
Img-A-7	86	21.065	75	0.4713	0.1160	98.30
Img-A-8	82	19.683	87	0.3305	0.1051	97.88
Img-A-9	72	17.060	66	0.6472	0.1592	98.14
Img-A-10	135	30.807	81	0.6459	0.2469	98.66
Img-A-11	109	19.693	80	0.4614	0.1439	97.35
Img-A-12	110	21.531	67	0.3445	0.1078	96.89

Image	Number of iterations	Testing time in seconds	final snake Points	Precision	Recall	Accuracy %
Image Set2						
Img-B-1	77	19.119	76	0.6163	0.1698	97.73
Img-B-2	121	33.359	94	0.6586	0.1737	98.10
Img-B-3	66	15.728	57	0.6197	0.1833	99.02
Img-B-4	184	59.059	114	0.6441	0.1866	98.55
Img-B-5	164	54.081	113	0.7041	0.1958	98.32
Img-B-6	103	27.190	62	0.6975	0.1724	98.23
Img-B-7	41	9.017	60	0.5816	0.1426	97.72
Img-B-8	62	16.494	68	0.7236	0.2592	98.41
Img-B-9	87	26.427	88	0.6416	0.1617	98.33
Img-B-10	108	27.169	84	0.7279	0.1890	98.08
Img-B-11	139	34.539	81	0.6794	0.2352	98.56
Image Set3						
Img-C-1	42	8.678	62	0.7414	0.1998	97.27
Img-C-2	90	28.719	101	0.3866	0.1069	98.49
Img-C-3	105	22.063	92	0.5964	0.2219	98.58
Img-C-4	48	10.751	55	0.6332	0.1449	98.08
Img-C-5	100	26.681	81	0.6478	0.1730	97.68
Img-C-6	126	38.646	100	0.7630	0.1929	98.51
Img-C-7	143	48.973	93	0.6013	0.1717	98.64
Image Set4						
Img-D-1	115	23.71	69	0.4236	0.1307	97.04
Img-D-2	72	14.097	68	0.7016	0.1875	97.44
Img-D-3	55	13.73	67	0.531	0.3362	98.81
Img-D-4	123	26.027	76	0.3079	0.1975	98.19
Img-D-5	169	36.462	93	0.7599	0.2111	97.88
Average	**101.5**	**25.8**	**79**	**0.605**	**0.18**	**98%**

From Table 5, we observe that we did not get a high precision and recall values, because the image processing classification is not purely true or false classification as data mining, we can explain this result that the snake point can be too close to the edge map pixel but not exactly on the same coordinates, so it will be counted a false negative result. To assure and explain this, we used the second evaluation method which finds the ratio of distances between each extracted contour point and the truly edge map point.

As we have mentioned before we compare our testing results with GVF model, because it is one of the most known approaches that solved the concavity contours problem.

Gradient Vector Flow (GVF) model testing results

Table 6 shows the results of testing GVF model using evaluation Method 1 by Convolution Matrix to find precision, accuracy, and recall for 35 different aerial images of buildings and custom shapes. An open source implemented application [27] is used to test GVF using the two evaluation methods.

Table 6: GVF evaluation results using method-1

Image No.	Number of iterations	Testing time in seconds	precision	Recall	Accuracy %
Image Set1					
Img-A-1	1000	127.069	0.4	0.2331	99.27
Img-A-2	1000	76.283	0.3333	0.2052	98.86
Img-A-3	660	59.92	0.154	0.1168	98.92
Img-A-4	1000	82.959	0.2658	0.1691	99.08
Img-A-5	450	53.821	0.1073	0.0872	99.21
Img-A-6	500	59.739	0.1961	0.1378	99.16
Img-A-7	450	54.423	0.1906	0.1279	99.22
Img-A-8	600	54.799	0.1722	0.1318	98.88
Img-A-9	500	46.311	0.1583	0.1216	99.1
Img-A-10	1000	82.642	0.2601	0.1659	99.05
Img-A-11	900	73.506	0.2209	0.1491	98.6
Img-A-12	600	36.792	0.1793	0.1184	98.41
Image Set2					
Img-B-1	800	75.185	0.1124	0.0736	98.76
Img-B-2	600	54.659	0.1305	0.0857	99
Img-B-3	400	46.376	0.2016	0.1484	99.45
Img-B-4	850	111.072	0.2148	0.1415	99.2
Img-B-5	500	58.343	0.3495	0.2102	99.14
Img-B-6	500	58.434	0.3907	0.2123	99.24
Img-B-7	600	56.888	0.1495	0.107	98.9
Img-B-8	650	53.543	0.2579	0.1714	98.79
Img-B-9	750	78.364	0.1636	0.1087	99.18
Img-B-10	850	69.142	0.2403	0.1587	99.01
Img-B-11	900	70.995	0.2745	0.1718	99.01

Image No.	Number of iterations	Testing time in seconds	precision	Recall	Accuracy %
Image Set3					
Img-C-1	850	61.55	0.2306	0.159	98.47
Img-C-2	500	64.969	0.2771	0.1845	99.32
Img-C-3	500	49.733	0.1235	0.0845	98.95
Img-C-4	350	36.487	0.205	0.1461	99.15
Img-C-5	600	53.394	0.172	0.1145	98.76
Img-C-6	750	80.412	0.0839	0.0646	99.19
Img-C-7	900	122.252	0.2249	0.1332	99.31
Image Set4					
Img-D-1	950	71.62	0.302	0.1821	98.58
Img-D-2	700	69.4	0.0265	0.0218	98.52
Img-D-3	550	53.038	0.0484	0.0364	98.53
Img-D-4	800	61.183	0.1579	0.0954	98.18
Img-D-5	850	67.59	0.1986	0.142	98.83
Average	**696**	**66.65**	**0.205**	**0.135**	**99%**

Results of Method 2:

As described in section 5.2.3 about evaluation using Method 2, to find **Error Distance Ratio (ER_d)** we must first get minimum distances of $N \times N$ window size of snake point neighbors, so we did this experiments by choosing a window size **N=11** to find minimum distance between each snake point and it's neighbor edge map pixel. We have chosen this window size after testing different window sizes such as N=5, 7, 11, and 21, and we noticed that the sum of distances does not change in more than N=11 and the distance ratio is the same.

Table 7 shows results of testing our improved ACM model and GVF model using Method 2 for 35 different aerial images of buildings and custom shapes, to get Error Distance Ratio (ER_d) as described in Method 2 in section 5.2.3.

Table 7: Results of testing our improved ACM model and GVF model using method-2

Image	Our improved model Error Distance Ratio (ER_d) (pixels)	GVF model Error Distance Ratio (ER_d) (pixels)
Image Set1		
Img-A-1	0.3211	1.4446
Img-A-2	0.3875	1.3846
Img-A-3	0.4921	1.6438
Img-A-4	0.4173	1.6088
Img-A-5	0.3273	2.1125
Img-A-6	0.5777	1.3752
Img-A-7	0.6816	1.7873
Img-A-8	0.7440	2.0654
Img-A-9	0.4508	2.9364
Img-A-10	0.3854	1.4528
Img-A-11	0.6165	1.9858
Img-A-12	0.7850	1.8839
Image Set2		
Img-B-1	0.4843	2.2924
Img-B-2	0.3848	2.1955
Img-B-3	0.4617	1.5643
Img-B-4	0.4657	1.3554
Img-B-5	0.3970	1.9098
Img-B-6	0.4211	1.4708
Img-B-7	0.4537	2.6542
Img-B-8	0.3813	1.3717
Img-B-9	0.4201	1.9058
Img-B-10	0.3125	1.6941
Img-B-11	0.3564	1.2862
Image Set3		
Img-C-1	0.3276	1.6542
Img-C-2	0.7133	1.6455
Img-C-3	0.4252	2.4228
Img-C-4	0.3973	1.6908
Img-C-5	0.3705	1.7994
Img-C-6	0.2633	2.0419
Img-C-7	0.5919	1.6414
Image Set4		
Img-D-1	0.5978	1.474
Img-D-2	0.2819	1.6776
Img-D-3	0.4203	2.4059
Img-D-4	0.7557	1.613
Img-D-5	0.2109	1.3531
Average	**0.46**	**1.79**

After applying our improved ACM model and GVF snake model on the same set of images, and test these two methods using the two described evaluation methods, the

results showed that the final extracted contour by our improved ACM is much better than using GVF snake.

Geometrically, by viewing the extracted contour the extracted shape using our improved model is fits the goal contour much better than the GVF model result contour. Table 8 shows some results of extracting contours using our improved ACM model compared to GVF model for images belongs to different sets.

Table 8: Contour extraction samples using improved ACM and GVF snake

Image	Original ROI image	Improved extracted ACM contour	GVF extracted contour
Img-B-1			
Img-A-3			
Img-B-2			
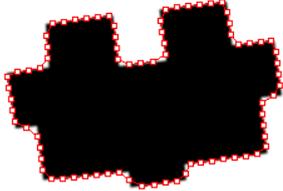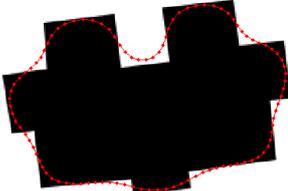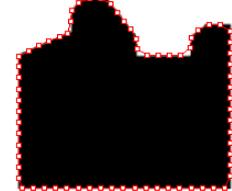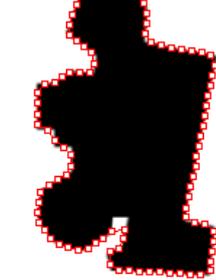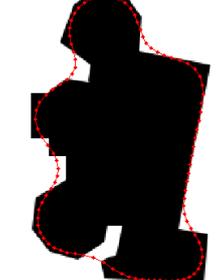
In addition, by comparing the testing results of evaluation Method1 and Method2 it shows approximate results in accuracy, but our improved model get better execution time and better Error Distance Ratio (ER_d) than GVF model. Table 9 summarize the different average values for each of GVF snake and the improved ACM model using test Method1 and Method2.

Table 9: General Comparison results of our Improved ACM and GVF

	Average number of iterations	Average ext. time (seconds)	Average Precision	Average Recall	Average Accuracy%	Distance Error Ratio (ER_d)
Improved ACM	101	25.8	0.6	0.18	98	0.46
GVF snake	696	66.65	0.20	0.13	99	1.79

The results of applying each of our Improved ACM Model and GVF model using the proposed evaluation methods showed a better results of Improved ACM than GVF in Precision and a comparable Accuracy results as shown in Figure 24. In addition, the results show a better execution time and lower error distance rate, where:

- The average value of the precision of our improved ACM is **0.6** while the GVF average precision value is **0.20,** which means that the ability of our improved model to detect contour points is higher than the ability of GVF model.

- The Error Distance Ratio (ER_d) of our improved ACM is **0.46**, and for the GVF ER_d average is **1.82** pixels, which means that the resulted contour snake points of our improved model is closer than the result snake points of the GVF model to the real contour. Figure 25 presents ER_d for all testing images.

- The execution time of our improved ACM is much better than the GVF, where the improved ACM execution average time is about **25.8** seconds, and the GVF average time is **67.0** seconds. Figure 26 presents a comparative diagram for executing time for GVF and our improved ACM model.
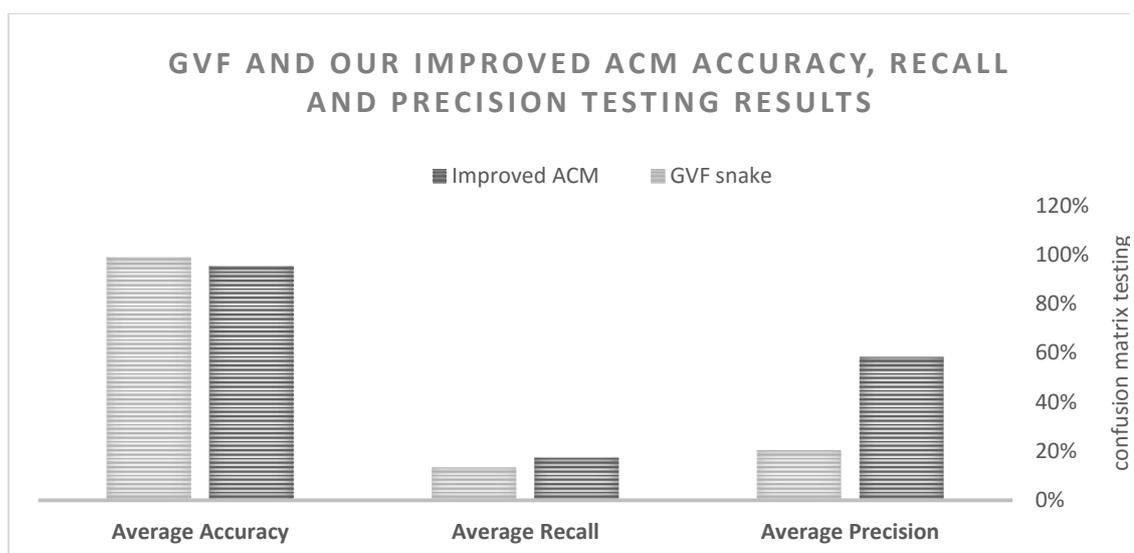
Figure 24: GVF and our improved ACM Accuracy, Recall and Precision testing results

Figure 25 presents the results of our improved model and GVF model results using testing Method2 for all the tested images, and get the Error Distance Ratio in pixels.
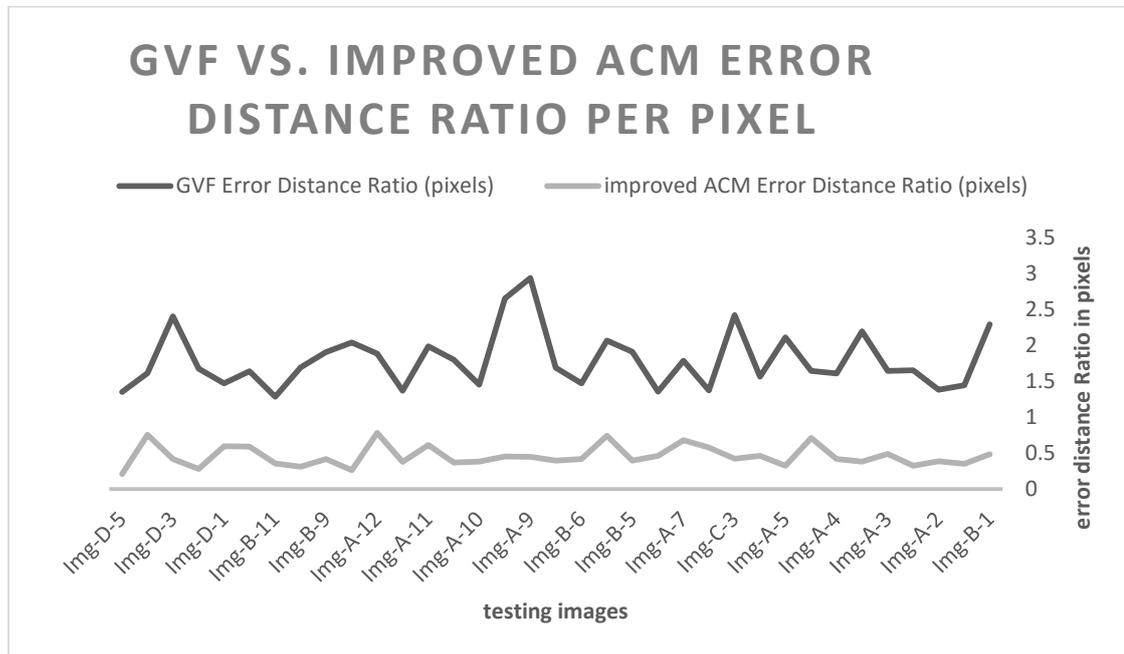
Figure 25: GVF vs. improved ACM distances ratio

Figure 26 shows a comparative results of time needed for contour extraction using our improved ACM and GVF, the results showed that execution time using GVF is more than the time of improved ACM.
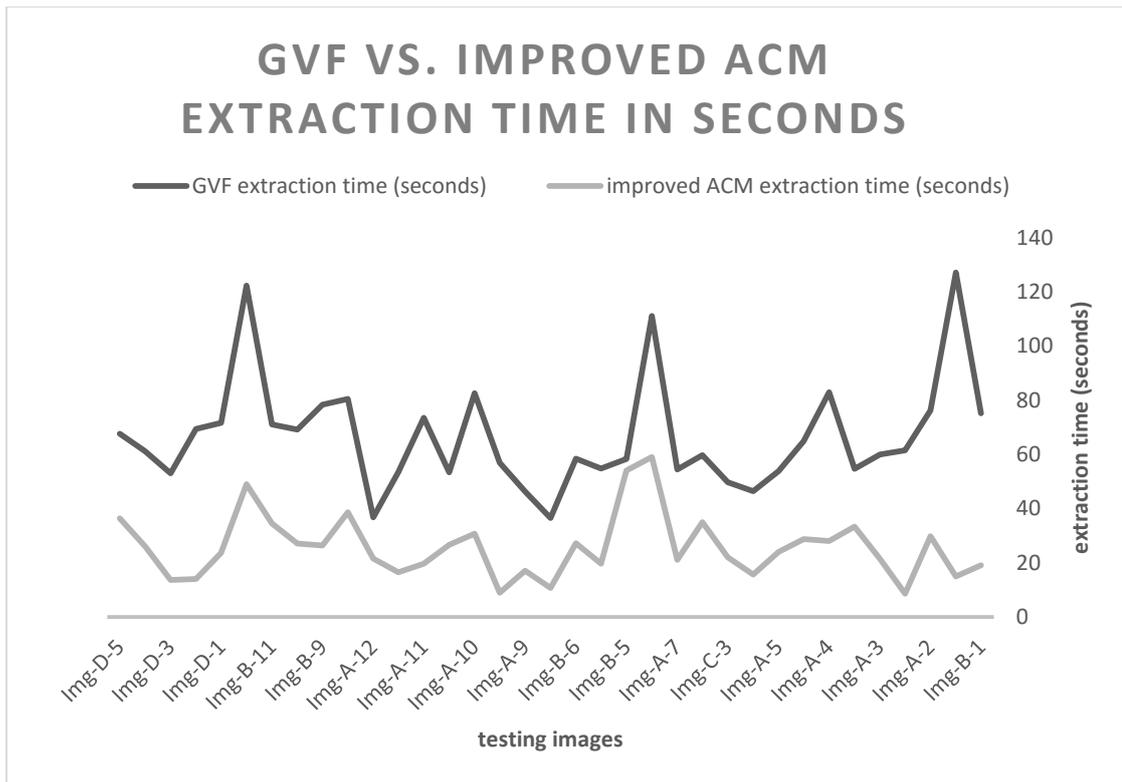
Figure 26: Results of GVF and our improved ACM executing time

6 Conclusion and Future Work

In this chapter we discuss the results of testing and evaluating the proposed ACM Model and the comparative results with the GVF Model, then we discuss the future works.

6.1 Conclusion

In this thesis, we have improved the Active Contour Model to solve the problem of extracting contours of concavity regions. The proposed improvements depend on the effect of the internal and external energies coefficients effect on snake movement. A concavity index (***concIndex***) parameter have been proposed to adapt internal force coefficient and make the snake converge inside narrow concavities.

The proposed method is tested using two evaluation methods, the first method finds the Accuracy and Precision and the second method finds the Error Distance Ratio, which is the ratio of the distances between each contour snake point and the true contour point by pixels. Also we have compared our improved ACM model with GVF model, which is one of the most used methods that solved the extraction of concavity contours problem.

The evaluation results of our improved ACM Model Accuracy is comparable with the GVF Model, where the average Accuracy of our improved ACM Model is 98% and GVF Model average accuracy is 99%. However, the Precision of our improved model is 60% while the GVF average precision value is 20%. In addition, the average execution time of our improved ACM model is much better than GVF Model, where the average execution time of our improved ACM Model is about **25.8** seconds, and GVF average execution time is **67.0** seconds.

To assure our results we tested the two models using the second evaluation method, which showed that the error ratio of distances between each snake point and the true edge pixels for the GVF model is **1.82** pixels and for our improved ACM is **0.46**. This means that the resulted contour shape of our improved model is closer than the GVF model to the real contour. In general, satisfying results is shown after testing our improved ACM model in execution time and accuracy compared to the GVF Model.

6.2 Future work

We have concentrated in our thesis to improve the ACM model to solve the problem of extracting buildings contour concavities in aerial images, so we are looking forward to improve the full process of automatic building extraction technique including our improved ACM method, which include other situations such as:

- Preprocessing image to remove noise.
- Automate the snake initialization process.
- Full automate building detection and removing other feature objects.
- Test and modify our improved ACM method to extract close buildings that separated by narrow region.

References

[1] B. Sırmacek and C. Unsalan, "Building Detection from Aerial Images using Invariant Color Features and Shadow Information," in *ISCIS 23rd International Symposium on Computer and Information Sciences*, Istanbul, 2008.

[2] M. Kass, W. Andrew and T. Demetri, "Snakes: Active Contour Models," *International Journal Of Computer Vision,* pp. 321-331, 1988.

[3] P. Saeedi and H. Zwick, "Automatic Building Detection in Aerial and Satellite Images," in *Intl. Conf. on Control, Automation, Robotics and Vision*, Hanoi, Vietnam, 2008.

[4] D.-C. Park, V. T. L. Huong, D.-M. Woo and Y. Lee, "Extraction of Rectangular Boundaries from Aerial Image Data," in *International Conference on Computer Engineering and Technology*, 2009.

[5] H. Baluyan, B. Joshi, A. AlHinai and W. L. Woon, "Novel Approach for Rooftop Detection Using Support Vector Machine," *Hindawi Publishing Corporation,* vol. Volume 2013, no. 1, pp. 1-12, 2013.

[6] A. J. Fazan and A. Porfírio, "Building Roof Contours Extraction from Aerial Imagery Based On Snakes and Dynamic Programming," in *FIG Congress 2010 - Facing the Challenges – Building the Capacity*, Sydney, 2010.

[7] S. Mayunga, Y. Zhanga and J. Coleman, "Semi-Automatic Building Extraction Utilizing Quickbird Imagery," in *CMRT05: Object Extraction for 3D City Models, Road Databases, and Traffic Monitoring - Concepts, Algorithms, and Evaluation*, Vienna, 2005.

[8] H. Fu, W. Rongqiu and W. Weimin, "Active Contour Model based on Dynamic Extern Force and Gradient Vector Flow," *2008 International Conference on BioMedical Engineering and Informatics,* pp. 863-867, 2008.

[9] Wikipedia, "Active contour model," Wikipedia, the free encyclopedia, 2015. [Online]. Available: http://en.wikipedia.org/wiki/Active_contour_model. [Accessed 10 10 2015].

[10] N. Efford, Digital Image Processing - A Practical Introduction Using Java, Pearson Education Limited, 2000.

[11] R. Maini and H. Aggarwal, "Study and Comparison of Various Image Edge Detection Techniques," *International Journal of Image Processing (IJIP),* pp. 1-11, 2009.

[12] A. Chopra and B. R. Dandu, "Image Segmentation Using Active Contour Model," *International Journal Of Computational Engineering Research,* pp. 819-822, June 2012.

[13] N. Saqer, A Two-Phase Snake Method for Object Segmentation against Background Clutter and Boundary Concavities, 2015.

[14] Z. Hui, W. Yuening and Y. Huaip, "A Snake Model Based on Improved Internal Energy," in *2010 International Conference on Computational Aspects of Social Networks*, 2010.

[15] S.-h. Kim, A. Alattar and J. W. Jang, "A Snake-Based Segmentation Algorithm For Objects With Boundary Concavities," *ICME 2006,* pp. 265-268, 2006.

[16] . X. Chenyang and P. Jerry L., "Snakes, Shapes, and Gradient Vector Flow," *IEEE Transaction on image processing,* vol. NO. 7, pp. 359-369, March 1998.

[17] S. L. a. R. Li, "A New Deformable Model Using Dynamic Gradient Vector Flow and Adaptive Balloon Forces," in *APRS Workshop on Digital Image Computing*, Brisbane, 2003.

[18] Z. Mengmeng, Q. Li, L. Li and P. Bai, "An Improved Algorithm Based on the GVF-Snake for Effective Concavity Edge Detection," *Journal of Software Engineering and Applications,* pp. 174-178, April 2013.

[19] L. Cohen, "On Active Contour Models and Balloons," *Computer Vision, Graphics, and Image Processinf: Image Understanding,* vol. 2, pp. 211-218, 1991.

[20] S. Muller and D. W. Zaum, "Robust Building Detection In Aerial Images," in *CMRT05: Object Extraction for 3D City Models, Road Databases, and Traffic Monitoring - Concepts, Algorithms, and Evaluation*, Vienna, 2005.

[21] A. Bhadauria, H. Bhadauria and A. Kumar, "Building Extraction from Satellite Images," *IOSR Journal of Computer Engineering (IOSR-JCE),* pp. 76-81, Jun. 2013.

[22] S. Ahmady, H. Ebadi, M. Valadan Zouj and H. A. Moghaddam, "Automatic Building Extraction From High Resolution Aerial Images Using Active Contour Model," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences.,* vol. XXXVII, pp. 453-456, 2008.

[23] A. Kovacs and T. Sziranyi, "Orientation Based Building Outline Extraction In Aerial Images," in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Melbourne, 2012.

[24] L. B. Theng, "Automatic Building Extraction from Satellite Imagery," *Engineering Letters,* 4 November 2006.

[25] M. Kabolizade, H. Ebadi and S. Ahmadi, "An Improved Snake Model for Automatic Extraction of Buildings from Urban Aerial Images and LiDAR Data Using Genetic Algorithm," *In: Paparoditis N., Pierrot-Deseilligny M., Mallet C., Tournaire O. (Eds), IAPRS,* pp. 45-49, 1-3 September 2010.

[26] T. Yoon, T. Kim, W. Park and T. G. Kim, "Building Segmentation using an Active Contour Model," 20 Jan. 2000. [Online]. [Accessed 2015].

[27] D.-J. Kroon, "File Exchange - MATLAB Central," MATLAB, November 2011. [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/28149-snake---active-contour/content/Snake2D.m. [Accessed 1 October 2015].

[28] Wikipedia, "Image segmentation," Wikipedia, the free encyclopedia, [Online]. Available: http://en.wikipedia.org/wiki/Image_segmentation. [Accessed 23 2 2015].

[29] K. Waseem, "Image Segmentation Techniques : A Survey," *Journal of Image and Graphics,* pp. 166-170, 2013.

[30] R. Maurya, D. P. Gupta and A. S. Shukla, "Road Extraction Using K-Means Clustering and Morphological Operations," in *International Conference on Image Information Processing*, 2011.

[31] C. Xu and J. L. Prince, "Gradient Vector Flow: A New External Force for Snakes," in *IEEE Proc. Conf. on Comp. Vis. Patt. Recog*, 1997.

[32] W.-P. Choi, K.-M. Lam and W.-C. Siu, "An adaptive active contour model for highlyirregular boundaries," *The journal of the pattern recognition society,* pp. 323-331, 2001.

[33] J. Cheng and X. Sun, "Medical Image Segmentation with Improved Gradient Vector Flow," *Research Journal of Applied Sciences, Engineering and Technology,* pp. 3951-3957, April 2012.

[34] Wikipedia, the free encyclopedia, "Geographic information system," 9 Dec. 2015. [Online]. Available: https://en.wikipedia.org/wiki/Geographic_information_system. [Accessed 17 Dec. 2015].

[35] B. Sırmaçek and . Ü. Cem, "Urban Area Detection Using Local Feature Points and Spatial Voting," *IEEE Geoscience And Remote Sensing Letters,* pp. 146-150, January 2010.

[36] J. Youn and J. S. Bethel, "Adaptive Snakes For Urban Road Extraction," Purdue University, West Lafayette, 2002.

[37] L. B. Theng and R. Chiong, "An improved Snake for Automatic Building Extraction," in *6th WSEAS Int. Conference on Computational Intelligence, Man-Machine Systems and Cybernetics*, Tenerife, 2007.

[38] R. Valdés, V. Medina and O. Yáñez, "Coupling of Radial-Basis Network and Active Contour Model for Multispectral Brain MRI Segmentation," *IEEE Transactions On Biomedical Engineering,* March 2004.

Appendix

Test set-1: Images with wide concavities
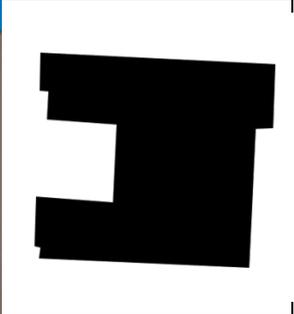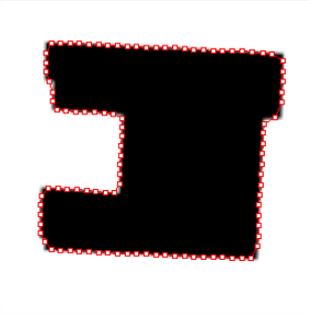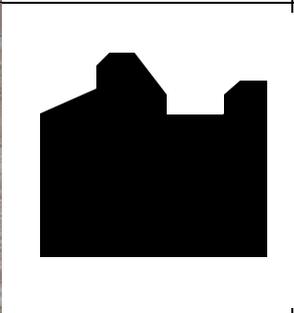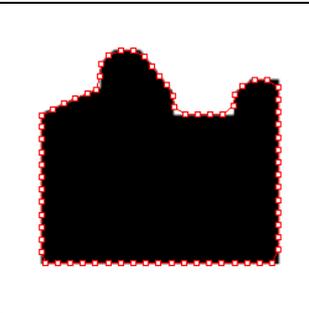
Image.	Original ROI image	Processed image	Final image with extracted contour
Img-A-2			
Img-A-3			
Img-A-4			
Img-A-5			
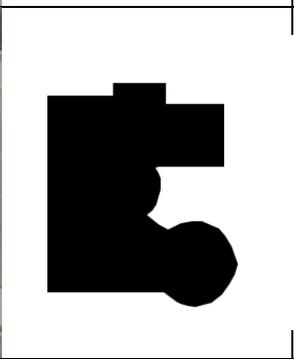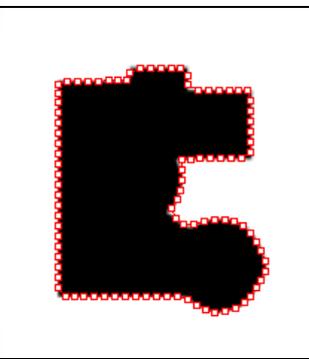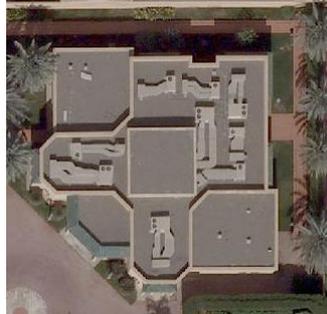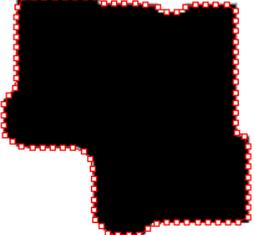
|			
Img-A-6			
Img-A-7			
Img-A-8			
Img-A-9			
Img-A-10			
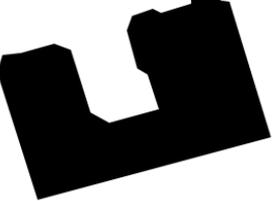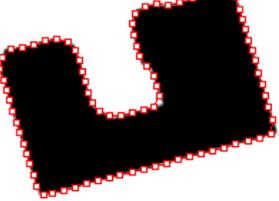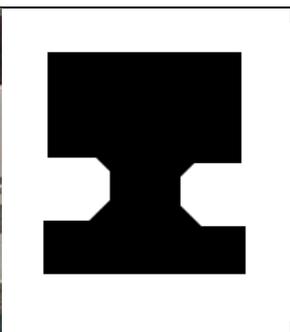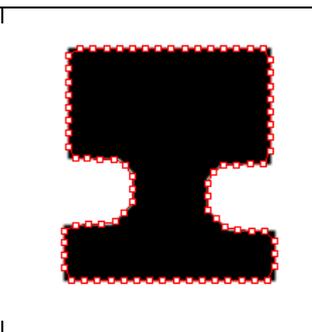
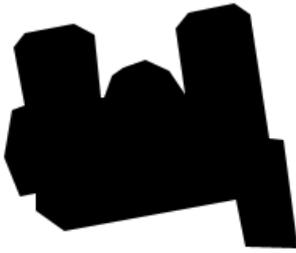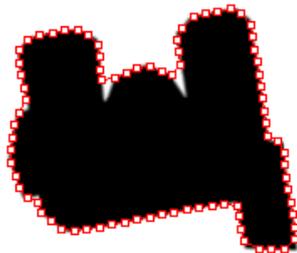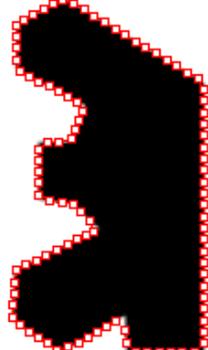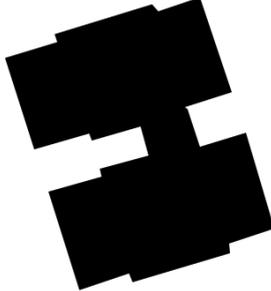| Img-A-11			
Img-A-12			

Test set-2: Images with narrow concavities
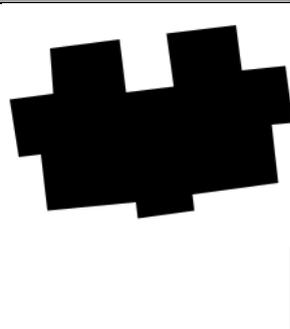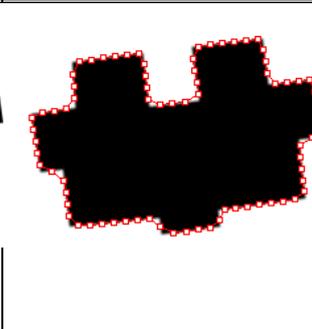
Image.	Original ROI image	Processed image	Final image with extracted contour
Img-B-1			
Img-B-2			
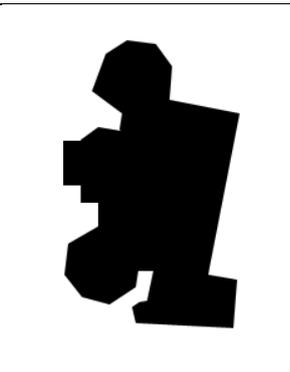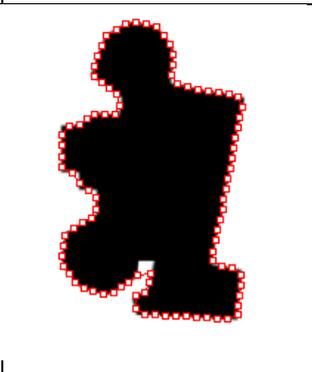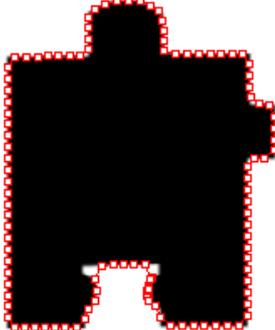
Img-B-3			
Img-B-4			
Img-B-5			
Img-B-6			
Img-B-7			
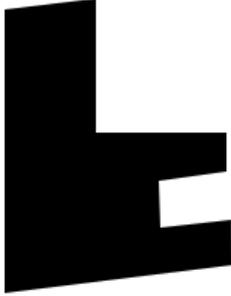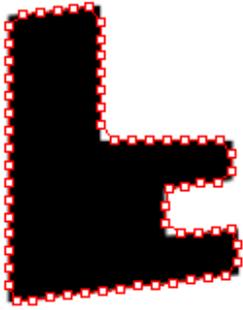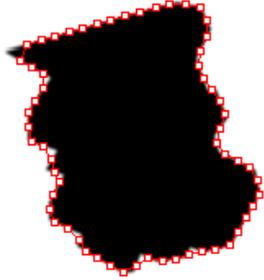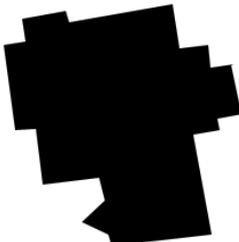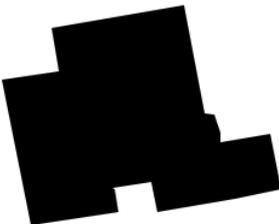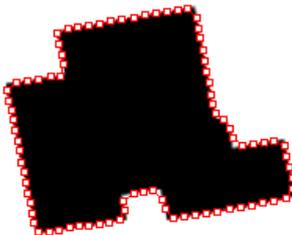
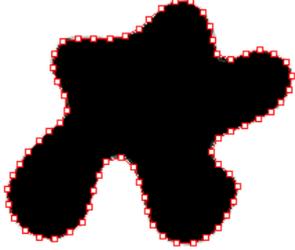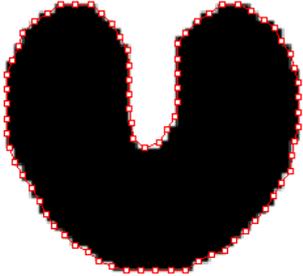|			
Img-B-8			
Img-B-9			
Img-B-10			
Img-B-11			
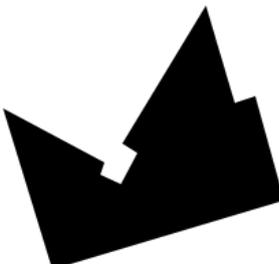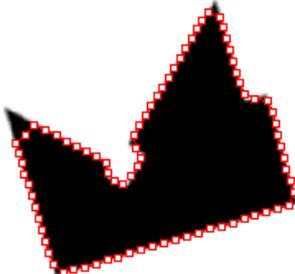
Test set-3: Images with multiple narrow concavities
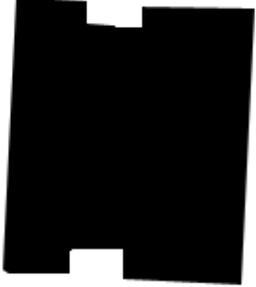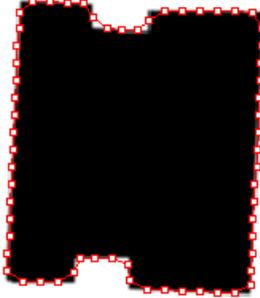
Image	Original ROI image	Processed image	Final image with extracted contour
Img-C-1			
Img-C-2			
Img-C-3			
Img-C-4			
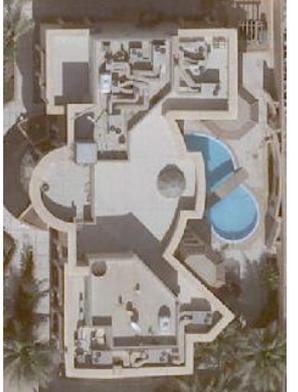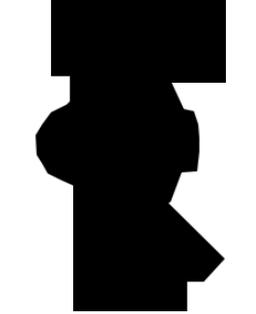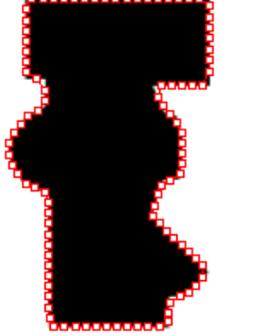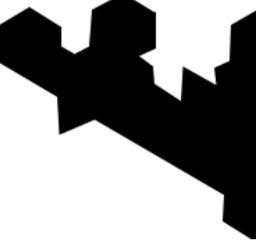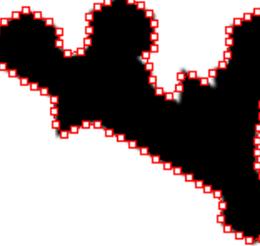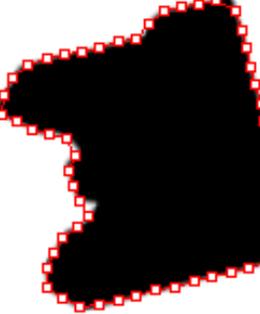
Img-C-5			
Img-C-6			
Img-C-7			

Test level-4: Images with custom shapes and concavities

Image	Original preprocessed image	Final image with extracted contour
Img-D-1		
Img-D-2		

Img-D-3		
Img-D-4		
Img-D-5		