SELF-ORGANIZING APPROACH TO LEARN A LEVEL-SET FUNCTION FOR OBJECT SEGMENTATION IN COMPLEX BACKGROUND ENVIRONMENTS

Dissertation

Submitted to

The School of Engineering of the

UNIVERSITY OF DAYTON

In Partial Fulfillment of the Requirements for

The Degree of

Doctor of Philosophy in Electrical Engineering

By

Fatema A. Albalooshi

UNIVERSITY OF DAYTON

Dayton, Ohio

May, 2015
SELF-ORGANIZING APPROACH TO LEARN A LEVEL-SET FUNCTION FOR
OBJECT SEGMENTATION IN COMPLEX BACKGROUND ENVIRONMENTS

Name: Albalooshi, Fatema A.

APPROVED BY:

Vijayan K. Asari, Ph.D.
Advisory Committee Chairman
Professor, Department of Electrical and
Computer Engineering

Raúl E. Ordóñez, Ph.D.
Committee Member
Professor, Department of Electrical and
Computer Engineering

Eric J. Balster, Ph.D.
Committee Member
Associate Professor, Department of
Electrical and Computer Engineering

Muhammad Usman, Ph.D.
Committee Member
Associate Professor, Department of
Mathematics

John G. Weber, Ph.D.
Associate Dean
School of Engineering

Eddy M. Rojas, Ph.D., M.A., P.E.
Dean, School of Engineering
ABSTRACT

SELF-ORGANIZING APPROACH TO LEARN A LEVEL-SET FUNCTION FOR OBJECT SEGMENTATION IN COMPLEX BACKGROUND ENVIRONMENTS

Name: Albaloshi, Fatema A.
University of Dayton

Advisor: Dr. Vijayan K. Asari

Boundary extraction for object region segmentation is one of the most challenging tasks in image processing and computer vision areas. The complexity of large variations in the appearance of the object and the background in a typical image causes the performance degradation of existing segmentation algorithms. One of the goals of computer vision studies is to produce algorithms to segment object regions to produce accurate object boundaries that can be utilized in feature extraction and classification.

This dissertation research considers the incorporation of prior knowledge of intensity/color of objects of interest within segmentation framework to enhance the performance of object region and boundary extraction of targets in unconstrained environments. The information about intensity/color of object of interest is taken from small patches as seeds that are fed to learn a neural network. The main challenge is accounting for the projection transformation between the limited amount of prior information and the appearance of the real object of interest in the testing data. We address this problem by the use of a Self-organizing Map (SOM) which is an unsupervised learning neural network. The segmentation process is achieved by the construction of a local fitted image level-set
cost function, in which, the dynamic variable is a Best Matching Unit (BMU) coming from the SOM map.

The proposed method is demonstrated on the PASCAL 2011 challenging dataset, in which, images contain objects with variations of illuminations, shadows, occlusions and clutter. In addition, our method is tested on different types of imagery including thermal, hyperspectral, and medical imagery. Metrics illustrate the effectiveness and accuracy of the proposed algorithm in improving the efficiency of boundary extraction and object region detection.

In order to reduce computational time, a lattice Boltzmann Method (LBM) convergence criteria is used along with the proposed self-organized active contour model for producing faster and effective segmentation. The lattice Boltzmann method is utilized to evolve the level-set function rapidly and terminate the evolution of the curve at the most optimum region. Experiments performed on our test datasets show promising results in terms of time and quality of the segmentation when compared to other state-of-the-art learning-based active contour model approaches. Our method is more than 53% faster than other state-of-the-art methods. Research is in progress to employ Time Adaptive Self-Organizing Map (TASOM) for improved segmentation and utilize the parallelization property of the LBM to achieve real-time segmentation.
TABLE OF CONTENTS

ABSTRACT ................................................................. iii
LIST OF FIGURES ........................................................ vii
LIST OF TABLES .......................................................... x
LIST OF NOTATIONS AND ABBREVIATIONS .......................... xi

I. INTRODUCTION ......................................................... 1
   1.1 Focus and Contributions ........................................ 4
       1.1.1 Self-Organized Learning Based Active Contours (SOLAC) . 4
       1.1.2 Self-Organized Lattice Boltzmann Active Contours (SOLBAC) . 6
   1.2 Specific Objectives ............................................. 7
   1.3 Dissertation Outline ........................................... 7

II. BACKGROUND AND RELATED WORK ................................. 9
   2.1 Active Contour Models ........................................... 9
       2.1.1 Parametric Active Contours (PACs) .......................... 10
       2.1.2 Geometric Active Contours (GACs) ........................ 12
       2.1.3 The Level-Set Function Method ............................. 13
       2.1.4 Level-Set Function-Based Segmentation ................... 21
       2.1.5 ACM Driven by Local Image Fitting Energy ............... 23
   2.2 Active Contour Segmentation with Prior Information .......... 24
       2.2.1 Active Contour Segmentation with Neural Networks .......... 26
   2.3 The Self-Organizing Map (SOM) ................................ 28
       2.3.1 Self-Organizing Map-Based Segmentation .................. 32
   2.4 Lattice Boltzmann Method (LBM) ................................ 33
       2.4.1 Lattice Boltzmann Active Contour Segmentation ............ 35
   2.5 Summary ......................................................... 36

v
## III. SELF-ORGANIZED LEARNING BASED ACTIVE CONTOURS (SOLAC) SEGMENTATION

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 SOM Training Phase</td>
<td>42</td>
</tr>
<tr>
<td>3.2 SOM Mapping Process</td>
<td>44</td>
</tr>
<tr>
<td>3.2.1 Integrating Retrieved Prior Information in Level-Set Cost Function</td>
<td>44</td>
</tr>
<tr>
<td>3.3 Results and Evaluation</td>
<td>45</td>
</tr>
<tr>
<td>3.3.1 Evaluation Strategies</td>
<td>46</td>
</tr>
<tr>
<td>3.3.2 Evaluation of SOLAC on Visible Imagery</td>
<td>47</td>
</tr>
<tr>
<td>3.3.3 Application of SOLAC to Different Types of Imagery</td>
<td>54</td>
</tr>
<tr>
<td>3.4 Summary</td>
<td>66</td>
</tr>
</tbody>
</table>

## IV. A SELF-ORGANIZED LATTICE BOLTZMANN ACTIVE CONTOUR (SOLBAC) SEGMENTATION

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 The Proposed SOLBAC Approach</td>
<td>67</td>
</tr>
<tr>
<td>4.2 Results and Evaluation</td>
<td>72</td>
</tr>
<tr>
<td>4.2.1 Evaluation of SOLBAC on Visible Imagery</td>
<td>72</td>
</tr>
<tr>
<td>4.2.2 Application of SOLBAC to Different Types of Imagery</td>
<td>78</td>
</tr>
<tr>
<td>4.2.3 Computation Speed</td>
<td>78</td>
</tr>
<tr>
<td>4.3 Summary</td>
<td>82</td>
</tr>
</tbody>
</table>

## V. CONCLUSION AND FUTURE WORK

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUBLICATIONS</td>
<td>86</td>
</tr>
<tr>
<td>BIBLIOGRAPHY</td>
<td>89</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

1.1 Segmentation results of different objects ................................. 2
1.2 Over segmentation illustration ............................................. 6
2.1 Heaviside step function ...................................................... 17
2.2 Dirac delta function .......................................................... 18
2.3 Level-set function illustration ............................................. 19
2.4 A local fitted image after several iterations .............................. 24
2.5 A target human detected and tracked using active contour model .... 27
2.6 Self-organizing map architecture ......................................... 29
2.7 BMU’s neighbourhood ....................................................... 30
2.8 Shrinking neighbourhood radius ......................................... 31
2.9 Lattice structure .............................................................. 34
3.1 SOLAC system diagram ....................................................... 40
3.2 The functional framework of the proposed SOLAC segmentation system. . 41
3.3 Comparison of segmentation results in an image of an airplane ......... 48
3.4 SOLAC result in a new testing image .................................... 49
3.5 Comparison of segmentation results in an image of a goose ............. 50
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.5</td>
<td>Results of people segmentation obtained using different learning-based methods</td>
<td>75</td>
</tr>
<tr>
<td>4.6</td>
<td>Results of a dog segmentation obtained using different learning-based methods</td>
<td>76</td>
</tr>
<tr>
<td>4.7</td>
<td>Results of sheep segmentation obtained using different learning-based methods</td>
<td>77</td>
</tr>
<tr>
<td>4.8</td>
<td>Average statistics for a set of 100 natural images</td>
<td>78</td>
</tr>
<tr>
<td>4.9</td>
<td>Segmentation results of a hyperspectral image using SOLBAC</td>
<td>79</td>
</tr>
<tr>
<td>4.10</td>
<td>Segmentation results of a medical image using SOLBAC</td>
<td>80</td>
</tr>
<tr>
<td>4.11</td>
<td>Segmentation results of a thermal image using SOLBAC</td>
<td>80</td>
</tr>
<tr>
<td>4.12</td>
<td>Average time consumption comparison for a set of 100 images using different learning-based segmentation techniques</td>
<td>81</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>3.1</td>
<td>Average segmentation statistical results for a set of 100 natural images.</td>
<td>54</td>
</tr>
<tr>
<td>3.2</td>
<td>Average segmentation statistical results for five hyperspectral images.</td>
<td>57</td>
</tr>
<tr>
<td>3.3</td>
<td>True positives and true negatives rates.</td>
<td>63</td>
</tr>
<tr>
<td>3.4</td>
<td>Segmentation accuracy metrics.</td>
<td>63</td>
</tr>
<tr>
<td>3.5</td>
<td>Average segmentation statistical results for 66 thermal images.</td>
<td>65</td>
</tr>
<tr>
<td>4.1</td>
<td>Execution time for various segmentation methods</td>
<td>75</td>
</tr>
<tr>
<td>4.2</td>
<td>Average statistics for a set of 100 natural images</td>
<td>77</td>
</tr>
<tr>
<td>4.3</td>
<td>Average time consumption comparison using different learning based</td>
<td>81</td>
</tr>
<tr>
<td></td>
<td>segmentation techniques</td>
<td></td>
</tr>
</tbody>
</table>
LIST OF NOTATIONS AND ABBREVIATIONS

\( \delta(\phi) \) \hspace{1cm} \text{Dirac delta function}

\( \delta_{\varepsilon}(\phi) \) \hspace{1cm} \text{regularized Dirac delta function}

\( \hat{n} \) \hspace{1cm} \text{normal vector}

\( \lambda \) \hspace{1cm} \text{time constant}

\( \lambda_1 \) \hspace{1cm} \text{fixed positive parameter}

\( \lambda_2 \) \hspace{1cm} \text{fixed positive parameter}

\( \mu \) \hspace{1cm} \text{fixed positive parameter}

\( \nabla \) \hspace{1cm} \text{gradient operator}

\( \Omega \) \hspace{1cm} \text{image domain}

\( \phi \) \hspace{1cm} \text{level-set function}

\( \sigma \) \hspace{1cm} \text{radius of neighborhood function in a SOM}

\( \sigma_c \) \hspace{1cm} \text{the convection coefficient}

\( \tau \) \hspace{1cm} \text{relaxation time}

\( \mathbf{F} \) \hspace{1cm} \text{force acting upon a contour front}
\( \vec{r} \)  position of a cell in an LBM lattice

\( C \)  contour or curve

\( c_1 \)  average intensity value inside a contour

\( c_2 \)  average intensity value outside a contour

\( E(\phi) \)  level-set energy function

\( E^{LFI} \)  local image fitting energy

\( E^{SOM}(\phi) \)  SOM-based level-set energy function

\( e_i \)  velocity vector of a link in an LBM lattice

\( F \)  speed function

\( f^{eq}_i \)  equilibrium particle distribution

\( f_i \)  particle distribution of a link in an LBM lattice

\( G \)  Gaussian filter

\( g_e \)  function that captures prominent edges

\( H(\phi) \)  Heaviside function

\( H_{e}(\phi) \)  regularized Heaviside function

\( I \)  Input image

\( I^{LFI} \)  locally fitted image

\( I^{SOM} \)  self-organizing map fitted image

\( k \)  curvature of the contour
$L$ neural network learning rate

$m_1$ average of image intensities in a Gaussian window inside the contour

$m_2$ average of image intensities in a Gaussian window outside the contour

$S$ sign function

$s$ smoothed sign function

$s_1, s_2$ level-set foreground and background parameters, respectively

$t$ time

$U$ set of training patterns

$V$ fixed positive parameter

$V_0$ a constant

$W$ neural network weight matrix

ACM active contour models

ANN artificial neural network

CSOM-CV concurrent self-organizing map Chan-Vese based segmentation

CT computed tomography

EB-SOM-ACM edge-based self-organizing map active contour model

FN false negative

FNR false positive rate

FP false positive
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPR</td>
<td>false negative rate</td>
</tr>
<tr>
<td>GAC</td>
<td>geometric active contour</td>
</tr>
<tr>
<td>LBM</td>
<td>lattice Boltzmann method</td>
</tr>
<tr>
<td>LFI</td>
<td>locally fitted image</td>
</tr>
<tr>
<td>LITFL</td>
<td>life in the fast lane</td>
</tr>
<tr>
<td>LSE</td>
<td>level-set equation</td>
</tr>
<tr>
<td>MRI</td>
<td>magnetic resonance imaging</td>
</tr>
<tr>
<td>P</td>
<td>precision value</td>
</tr>
<tr>
<td>PAC</td>
<td>parametric active contour</td>
</tr>
<tr>
<td>PCA</td>
<td>principal component analysis</td>
</tr>
<tr>
<td>PDE</td>
<td>partial differential equation</td>
</tr>
<tr>
<td>RE</td>
<td>recall value</td>
</tr>
<tr>
<td>SOLAC</td>
<td>self-organizing learning based active contours</td>
</tr>
<tr>
<td>SOLBAC</td>
<td>self-organizing lattice Boltzmann active contours</td>
</tr>
<tr>
<td>SOM</td>
<td>self-organizing map</td>
</tr>
<tr>
<td>SP</td>
<td>specificity value</td>
</tr>
<tr>
<td>SRG</td>
<td>seeded region growing</td>
</tr>
<tr>
<td>TASOM-ACM</td>
<td>time adaptive self-organizing map active contour model</td>
</tr>
<tr>
<td>TCIA</td>
<td>the cancer image archives</td>
</tr>
<tr>
<td>TN</td>
<td>true negative</td>
</tr>
<tr>
<td>-----</td>
<td>--------------</td>
</tr>
<tr>
<td>TP</td>
<td>true positive</td>
</tr>
</tbody>
</table>
CHAPTER I

INTRODUCTION

Human visual experience in distinguishing different objects has been always under the focus of computer vision researchers. Studies and explorations of human visual perception have been the main source of inspiration for computer pattern recognition applications. Understanding how the human brain represents basic attributes of objects helps in developing computer vision algorithms that perform automatic object interpretation and scene understanding. Because human visual perception is based on the neural coding of fundamental features, such as object boundaries, color, orientation, shape, etc.; finding the contours and boundaries of visual objects provides the first step for object recognition and interpretation algorithms.

Image scenes usually contain different objects with varying attributes like shape, size, color, and texture as shown in figure 1.1. In order to build algorithms for object detection and identification in computer vision, image segmentation is generally an essential preprocessing stage. The goal of segmentation is to simplify and change the representation of an image into something that is easier to analyze. The role of segmentation is crucial in most tasks requiring image analysis, scene understanding and object interpretation because the accuracy of segmentation can tremendously affect the results of those applications.
Figure 1.1: Segmentation results of different objects; (a) Input boat image; (b) Segmented boat; (c) Input basket image in cluttered environment (d) Segmented basket; (e) Input table image (f) Segmented table. Imagery obtained from [1].
However, many practical applications utilize natural daily scenes that may consist of relatively random objects in a cluttered environment, making segmentation process very challenging. Standard segmentation is identifying object/objects of interest and separating them entirely from the background. If such an ideal segmentation were possible, then other subsequent processes like object recognition, identification, classification and interpretation would be more efficient since only segmented regions need to be considered.

The increasing attention to the accuracy of image segmentation is motivated by several post-processing steps such as feature extraction and classification. As a preprocessing step of a wide range of computer vision applications, image segmentation needs to be accurate, reliable, and also provide high performance results. Classical segmentation techniques usually perform the segmentation process to cluster and classify pixels in the whole image. In applications that require particular object segmentation, prior information of the object of interest has to be incorporated in the segmentation algorithm. Our primary objective is to obtain an accurate region segmentation and boundary extraction of objects in complex background environments.

The Active Contour Model (ACM) method [2] has achieved a great success in image segmentation because it is capable of extracting boundaries of objects with complex shapes and different sizes, it assures continuous closed boundaries in the resulting segmentation, and it has been proven to be powerful in motion tracking [3]. Region based active contour models use regional statistics of the input image having the advantage of being more robust to noise and less sensitive to the placement of the initial contour. The use of local energy minimization techniques to extract local image information is useful to handle the problem of accurate boundary extraction and segmentation in images of inhomogeneous intensity levels [4]. Therefore, in order to improve the quality of active contours method, in this dissertation research, a local image information is extracted in order to help better evolution of the initial contour.
The Level-Set Function (LSF) technique introduced in [5] provides a methodology for tracking curves and thus produces accurate boundaries of objects. This method has been used broadly for surface evolution because it can implicitly represent the evolution of contours by embedding them as the zero level of a level-set function. Region based active contour method can be implicitly implemented by the level-set function technique.

1.1 Focus and Contributions

The focus of this research is to develop a novel technique to accurately segment objects of different attributes in complex background environments using a learning based active contour model.

1.1.1 Self-Organized Learning Based Active Contours (SOLAC)

Despite the increasing popularity of the ACM, this novel approach still has several limitations. One limitation is the problem of sticking at local minima, causing over-segmentation and resulting in poor boundary output. This problem can be clearly observed in images that contain varying shades of colors as shown in Figure 1.2. In this research, we address this challenge by utilizing prior knowledge of object of interest in the segmentation process. Segmentation with prior knowledge has been used to enhance the performance of segmentation and to guide the cost functions towards objects of interests. The nature of prior information differs according to segmentation method being used, and it varies to be either color distribution, intensity level, texture information, gradient, motion, or shape. It can also be a combination of several features to characterize regions of objects of interest. We introduce a segmentation technique that integrates intensity/color prior information of object of interest in the segmentation process that is achieved by utilizing a level-set framework. Motivated by the specific ability of Self-Organizing Maps (SOMs) to learn information about the object of interest and its background, self-organizing ACMs have been proposed with the aim of modeling and controlling the evolution of the active contour in an effective manner [6–11]. One
A reason to prefer SOMs over other neural networks is their specific ability to learn the intensity/color information via their topology preservation property. In addition, they have the property of reduction of dimensionality and thus reduce computational time. The accurate segmentation of object boundaries and region extraction is established in this dissertation research by building a Local Image Fitting (LIF) level-set cost function where the dynamic variable is the prior information of the object of interest and its background. This prior information is retrieved using SOM neural networks. Prior knowledge of the intensity/color of the object of interest can extremely help in guiding the segmentation process, specially when the object boundaries are not well defined and when the image contains inhomogeneous intensity variations. In this research we embed neural networks within the level-set ACM evolution function and this results in promising results for object boundary and region extraction. Thus, we are combining a spatially blind clustering technique (SOM) with spatially guided approach (level-set ACM) to attain the desired segmentation output.

The main difficulty is to incorporate the representation of prior intensity/color information into the level-set evolution equation of the active contour model during the segmentation process. This problem is addressed by constructing a novel segmentation framework that employs a dual SOM to learn the object of interest and the background independently. The trained SOM models are then used to extract the characteristics of the objects of interest in the testing image. After that, the clustered testing image is introduced to the level-set ACM framework in order to guide the active contour to extract the target region and lead to an accurate boundary.

One reason to prefer SOMs to other neural network models consists in the specific ability of SOMs to learn the intensity and/or color information via their topology preservation property. In addition, it has the property of reduction of dimensionality.
Figure 1.2: Over segmentation illustration; (a) original image; (b) over-segmented image due to the presence of local minima.

1.1.2 Self-Organized Lattice Boltzmann Active Contours (SOLBAC)

In classical level-set methods, the cost function is solved using the gradient descent method to characterize curve evolution. This leads to solving a Partial Differential Equation (PDE). Although this scheme showed a great promise, the need to simplify and speed up the computations has led us to the Lattice Boltzmann Method (LBM) [12] as an alternative approach for solving the Level-Set Function (LSF) [13, 14]. One of the merits of using the LBM method is that it can implicitly manipulate the curvature term in the LSF and therefore speeds up the process of curve evolution convergence.

In this dissertation research, we introduce a self-organized learning based active contour model with a lattice Boltzmann convergence criteria for fast and effective segmentation preserving the precise details of the object’s region of interest. The lattice Boltzmann method is utilized to evolve the LSF faster and terminate the evolution of the curve at the most optimum region, which segments objects in complex background environments.
1.2 Specific Objectives

The specific objectives for this dissertation research are the following:

1. Build an algorithm for accurate object region and boundary extraction.

2. Utilize segmentation with prior knowledge in order to enhance the performance of segmentation and to guide the cost function towards the objects of interests.

3. Build a segmentation technique that integrates intensity/color prior information of object of interest in the segmentation process that is achieved by using level-set framework.

4. Utilize self-organizing active contours to model and control the evolution of the active contour effectively.

5. Increase efficiency of segmentation performance in terms of time and accuracy by employing LBM based approach in the segmentation process.

1.3 Dissertation Outline

The dissertation is organized as follows. First, the background and related work are discussed in chapter II, which includes a summary of active contour models in section 2.1, active contour segmentation with prior information in section 2.2, the self-organizing map in section 2.3, lattice Boltzmann method in section 2.4, and chapter summary is provided in section 2.5. Next, self-organizing learning based active contours (SOLAC) segmentation is presented in chapter III. The proposed system is explained in a top-down approach, where the framework as a whole is described in section first, followed by an explanation of SOM training phase in section 3.1. SOM mapping phase integrating retrieved prior information in the level set cost function and are explained in section 3.2. Results and Evaluation are discussed in section 3.3. Section 3.4 summarizes the chapter.
In chapter IV a self-organizing lattice Boltzmann active contour approach for fast and robust object region segmentation is introduced. Section 4.1 provides a description of the proposed SOL-BAC approach. We discuss our findings and metrics used to evaluate this segmentation technique in Section 4.2. Summary is provided in section 4.3. Finally, chapter V concludes the dissertation and provides a description of future work for this research.
CHAPTER II

BACKGROUND AND RELATED WORK

This chapter presents a brief background about state-of-the-art active contour model-based segmentation techniques.

2.1 Active Contour Models

Active contours introduced by Kass et al. [2] provided an excellent approach of using elastic contours that stretch or expand to fit object regions in a scene. A contour refers to a $1 - D$ curve in a $2 - D$ space, $2 - D$ surface in a $3 - D$ space, generally a $(D - 1)$ dimensional hyper-surface in a D-dimensional space. Active contour models (ACMs) are well know spatially guided approach; in which, the segmentation process is accomplished by spatial relationships of pixels. Active contours approach make an effective use of specific prior information about object regions and boundaries, this makes them sufficient for segmentation. This segmentation method assures continuous closed boundaries in the resulting segmentation because the nature of active contours to deform a closed contour.

Active contour models involve the evolution of curves toward the boundary of an object through the solution of an energy function minimization problem. The energy function in active contour
models depend on the image characteristics and the shape of the contour. Therefore, they are considered a high-level image segmentation scheme as opposed to the traditional low-level schemes such as edge detectors or thresholding methods.

There are many techniques proposed in order to enhance the performance of curve evolution in ACMs. Such techniques include integrating disparate characteristics within the energy function of the ACM such as: local image region information, global image region information, object shape information, object feature prior information, and a-posteriori information learned from examples. In general, the existing active contour models can be classified as either parametric active contours (PACs) or geometric active contours (GACs) [4].

2.1.1 Parametric Active Contours (PACs)

PACs are typically rendered in a Lagrangian formulation in which the evolving curves are associated with a specific energy function expressed as a sum of external and internal energy terms. This approach was first introduced by Kass et al. [2], and within this model the edge-based ACMs utilize image intensity gradient as an additional constraint to stop the contours on the boundaries of the desired objects. Usually, a stopping function is used to attract the contours to the desired boundaries. In order to enlarge the capture range of the force, a balloon force term is often incorporated into the evolution function, which controls the contour to shrink or expand. However, it is challenging to choose a proper balloon force because either a too large or a too small balloon force will result in undesirable effects [4].

A family of smooth contours $C$ can be described as:

$$C(p, t) = \begin{bmatrix} x(p, t) \\ y(p, t) \end{bmatrix}$$

(2.1)

where $p \in [0, 1]$ parametrizes the set of points on each curve, and $t \in [0, \infty)$ parametrizes the family of curves at different time evolutions. With this scheme, a closed contour has the property
that \( C(0, t) = C(1, t) \ \forall t \). Let \( F(C) \) denotes the Euler-Lagrange equation such that the first variation of an energy function \( J(C) \) with respect to the contour \( C \) is zero. Then, the necessary condition for \( C \) to be the minimizer of \( J(C) \) is that \( F(C) = 0 \) under general assumptions. The solution to this necessary condition can be solved as the steady state solution to the following Partial Differential Equation (PDE)

\[
\frac{\partial C}{\partial t} = F(C) \tag{2.2}
\]

The above equation is denoted as the flow or curve evolution for the curve \( C \), where \( F \) denotes the force acting upon the contour front. Generally, the force \( F \) has two components: the component that points in the normal direction with respect to contour front \( F_n \), and the component that is tangent to the contour \( C \), \( F_t \).

In curve evolution, \( F_n \) is the force that moves the contour forward or (inward), which means that it is the force responsible to change the geometry of the contour. Therefore, the curve evolution equation is often reduced to the following equation:

\[
\frac{\partial C}{\partial t} = V_0 F \hat{n} \tag{2.3}
\]

where \( F \) is the speed function, \( \hat{n} \) is the normal force and \( V_0 \) is a constant. If \( V_0 \) is positive then the contour expands, and if \( V_0 \) is negative then the contour will shrink. This is because it corresponds to the minimization of the area within the closed contour.

Furthermore, equation (2.3) has been modified to move the contour \( C \) in the direction of its normal vector \( \hat{n} \) as follows [3]:

\[
\frac{\partial C}{\partial t} = -g_e \cdot (k + V_0) \hat{n} \tag{2.4}
\]

where \( k \) is the curvature of the contour and \( g_e \) is a function that captures prominent edges, it is described as follows:

\[
g_e = \frac{1}{1 + |\nabla G * I(x, y)|} \tag{2.5}
\]
where $\nabla$ is the gradient operator and is denoted by $\nabla I(x, y) = \left( \frac{\partial I(x, y)}{\partial x}, \frac{\partial I(x, y)}{\partial y} \right)$, $G$ is a smoothing Gaussian filter, $(\ast)$ is the convolution operator and $I$ is the input image. The contour front will slow down where the value of the edge map $|\nabla G * I(x, y)|$ is high because $g$ approaches zero; however, it will keep moving at a constant speed where the edge map value is zero.

A major drawback to this method is that the initial contour must be placed close enough to the true object boundary in order to achieve convergence to a pleasant outcome. If this was not the case, the evolving contour might stop at undesirable false edges or the contour might not move at all if the potential force on the contour front is not present. As a result, the initial contour is often obtained manually in this method. In addition, this scheme works well for objects that have well-defined edge map. However, when the object boundary is difficult to distinguish, the evolving contour may leak out [3].

### 2.1.2 Geometric Active Contours (GACs)

On the other hand, GACs are represented implicitly as an Eulerian formulation in which evolving curves are rendered as the level-sets of a distance function of two dimensions. Region based ACMs are examples of GACs that depend on regional statistics (e.g., intensity, texture, color distribution, etc.) and this gives them the advantage of being more robust to noise and less sensitive to the placement of the initial contour [4]. They utilize the image statistical information to construct constraints and have more advantages over edge-based models. They do not use the image gradient, and can successfully segment objects with weak boundaries or even without boundaries. Moreover, the initial contour can start anywhere in the image, and the interior contours can be automatically detected.
This method is known to be more robust to noise than the edge-based method because it relies on the regional statistics for segmentation [3]. More details about level-set ACMs is provided in the next section.

### 2.1.3 The Level-Set Function Method

The evolution of active contours [3] is often described by Partial Differential Equations (PDE), which can be tracked by different schemes, one of which is the level-set method.

Level-set methods are a collection of numerical algorithms for solving a particular class of PDEs. The advantage of level-set methods is that they can describe many types of surface motion and it can handle the merging and separation of surfaces without any significant increase in theoretical or implementation complexity.

For a contour \( C \) which evolves with normal \( \hat{n} \) and speed \( F \), one can derive a corresponding partial differential equation for the embedding function \( \phi \), with the fact that \( \phi(c(t), t) = 0 \) at all times, then the total time derivative of \( \phi \) at locations of the contour must vanish as:

\[
\frac{d}{dt}\phi(C(t), t) = \nabla \phi \frac{\partial C}{\partial t} + \frac{\partial \phi}{\partial t} = \nabla \phi F \cdot \hat{n} + \frac{\partial \phi}{\partial t} = 0
\]  

(2.6)

Since normal \( n = \frac{\nabla \phi}{|\nabla \phi|} \), we get:

\[
\frac{\partial \phi}{\partial t} = -|\nabla \phi| \cdot F
\]  

(2.7)

This specifies the evolution of \( \phi \). Thus, one can derive the Euler-Lagrange equation which minimizes \( E(\phi) \) as:

\[
\frac{\partial \phi}{\partial t} = - \frac{\partial E(\phi)}{\partial \phi}
\]

(2.8)

Osher and Sethian [5] developed the level-set technique for tracking curves in the Eulerian framework, written in terms of a fixed coordinate system. The level-set method implicitly represents
the evolving contour \(C(t)\) by embedding it as the zero level of a level-set function \(\phi\); that is,

\[
C(t) = \{(x, y) \in \Omega : \phi(x, y, t) = 0\}
\] (2.9)

Depending on the chosen embedding, one can obtain slightly different evolution equation for \(\phi(x, y, t)\). Chan and Vese [15] proposed an active contours scheme that does not use edges but combines an energy minimization approach with a level-set based solution. If \(C\) is a contour that partitions the domain of an image \(I(x, y) := (x, y) \in \omega\), where \(\omega\) is the image domain, into two regions, and if the region inside the contour is \(\Omega_1\) and the region outside the contour is \(\Omega_2\), then their approach minimizes the energy function given as:

\[
E_{CV}(C) = \mu \cdot \text{length}(C) + V \cdot \text{area}(\Omega_1) + \lambda_1 \int_{\Omega_1} |I(x, y) - c_1|^2 dxdy + \lambda_2 \int_{\Omega_2} |I(x, y) - c_2|^2 dxdy \tag{2.10}
\]

where \(\mu \geq 0\), \(\lambda_1, \lambda_2 > 0\), and \(V \geq 0\) are fixed parameters. \(c_1\) and \(c_2\) are the average intensity values of the image inside and outside the contour, respectively. Minimizing the above energy function by using the steepest decent method and representing the contour with the zero level-set, i.e. \(C(t) = \{(x, y) \in \Omega : \phi(x, y, t) = 0\}\), the following variational equation is obtained:

\[
\frac{\partial \phi}{\partial t} = \left[\mu \cdot \text{div} \left(\frac{\nabla \phi}{|\nabla \phi|}\right) - V - \lambda_1 (I - c_1)^2 + \lambda_2 (I - c_2)^2\right] \delta(\phi) \tag{2.11}
\]

where \(\delta(\phi)\) is the Dirac delta function. The data term \((-\lambda_1 (I - c_1)^2 + \lambda_2 (I - c_2)^2)\) plays a key role in curve evolution, and \(\lambda_1\) and \(\lambda_2\) govern the trade-off between the first term and the second term. Assuming that this level-set graph has positive values inside the contour \(C\) and negative values outside \(C\), that is,

\[
\text{inside}(C) = \Omega_1 = \{(x, y) \in \Omega : \phi(x, y, t) > 0\}, \tag{2.12}
\]
The level-set function $\phi$ can be implemented as the signed Euclidean distance to the contour $C$, using the standard Heaviside function as in:

$$H(\phi) = \begin{cases} 1, & \text{if } \phi \geq 0, \\ 0, & \text{if } \phi < 0. \end{cases}$$  \hspace{1cm} (2.14)$$

Therefore, the function $H(\phi)$ represents the binary template of the image pixels that are inside or on the contour. The function $(1 - H(\phi))$ represents the binary template of the image pixels that are strictly outside the contour. Thus, to select only the pixels that are on the contour $C$, we can use $(H(\phi) - [1 - H(-\phi)])$.

In order to facilitate numerical implementations, the regularized Heaviside function and its derivative, the regularized delta function, are often used instead. Steps to implement the level-set segmentation are the following:

1. Determine the Hamilton-Jacobi equation,

2. Pick the types of parameters,

3. Decide on the desired order of accuracy of the time derivative approximation,

4. Pick the boundary conditions,

5. Create the grid,

6. Create the initial condition $\phi(x, y, 0)$, and

7. Integrate forward in time.
Regularized Dirac delta functions

The concept of regularized Dirac delta function is widely used in level-set method to measure object surface area or to distribute a singular force [16]. Here, the formulation of level-set function includes a regularized Dirac delta measure support to track the evolution of curves and surfaces. The regularized versions of Heaviside step function is shown in equation (2.15)

\[
H_\varepsilon(\phi) = \frac{1}{2} \left( 1 + \frac{2}{\pi} \arctan\left( \frac{\phi}{\varepsilon} \right) \right)
\]

then the regularized delta function is

\[
\delta_\varepsilon(\phi) = \frac{d}{d\phi} H_\varepsilon(\phi),
\]

\[
= \frac{1}{\pi} \cdot \frac{\varepsilon}{\varepsilon^2 + \phi^2}, \quad \phi \in \mathbb{R}
\]

where \( \phi \) is the level-set function, \( \varepsilon \) is a positive real number that controls the width of \( \delta_\varepsilon(\phi) \). Figures 2.1 and 2.2 show normal and regularized Heaviside and Dirac delta functions, respectively, for different values of \( \varepsilon \).

Figure 2.3 shows an example of a level-set function and its corresponding contour.

Reinitialization of the level-set function

The implicit surface function known as the level-set function \( \phi \), is usually initialized to be a signed distance function before the evolution, and then it needs to be converted again into a signed distance function periodically during the evolution. Thus, reinitialization means modifying \( \phi \) to satisfy \( |\nabla \phi| \approx 1 \) without moving the level-set’s zero isosurface. Several numerical approximations have been proposed to achieve the reinitialization step properly in which to adjust the gradient magnitude of the level-set function to be close to unity[17–20].

16
Figure 2.1: Heaviside step function; (a) normal Heaviside step function; (b) regularized Heaviside function of equation (2.15) for different values of $\varepsilon$ and $|\phi| \leq 20$. 
Figure 2.2: Dirac delta function; (a) normal Dirac delta function; (b) regularized Dirac delta function of equation (2.17) for different values of $\varepsilon$ and $|\phi| \leq 20$. 
Figure 2.3: Level-set function illustration; (a) a level-set function $\phi$; (b) the same level-set function $\phi$ from a different angle; (c) zero level curve of the corresponding level-set function $\phi$. 
For example, if the curve evolution function is represented as:

\[
\frac{\partial \phi}{\partial t} + S(\phi_0)\left(\sqrt{\frac{\partial \phi^2}{\partial x} + \frac{\partial \phi^2}{\partial y}} - 1\right) = 0, \quad \phi(x, y, 0) = \phi_0(x, y)
\] (2.18)

Where \(\phi\) is the level-set function and \(S(\phi_0)\) is the sign function and \(\phi_0\) is the initial level-set function at time \(t_0\). Then this equation can be re-written as:

\[
\frac{\partial \phi}{\partial t} + \left(\frac{S(\phi_0)\frac{\partial \phi}{\partial x}}{\sqrt{\frac{\partial \phi^2}{\partial x} + \frac{\partial \phi^2}{\partial y}}} \right) \frac{\partial \phi}{\partial x} + \left(\frac{S(\phi_0)\frac{\partial \phi}{\partial y}}{\sqrt{\frac{\partial \phi^2}{\partial x} + \frac{\partial \phi^2}{\partial y}}} \right) \frac{\partial \phi}{\partial y} = S(\phi_0)
\] (2.19)

Using Godunov’s reinitialization method, if \(S(\phi_0)\frac{\partial \phi}{\partial x} \geq 0\) and \(S(\phi_0)\frac{\partial \phi}{\partial y} \geq 0\), then \(\frac{\partial \phi}{\partial x}\) is used. On the other hand, if \(S(\phi_0)\frac{\partial \phi}{\partial x} \leq 0\) and \(S(\phi_0)\frac{\partial \phi}{\partial y} \leq 0\) then \(\frac{\partial \phi}{\partial x}\) is used. Moreover, if \(S(\phi_0)\frac{\partial \phi}{\partial x} > 0\) and \(S(\phi_0)\frac{\partial \phi}{\partial x} < 0\) then \(\frac{\partial \phi}{\partial x} = 0\) is used. If \(S(\phi_0)\frac{\partial \phi}{\partial x} < 0\) and \(S(\phi_0)\frac{\partial \phi}{\partial x} > 0\) then the smoothed sign function is given by:

\[
s = \frac{S(\phi_0)(|\frac{\partial \phi}{\partial x}| - |\frac{\partial \phi}{\partial x}|)}{\frac{\partial \phi}{\partial x} - \frac{\partial \phi}{\partial x}}
\] (2.20)

The same procedure is repeated for \(S(\phi_0)\frac{\partial \phi}{\partial y}\) and the appropriate values for \(\frac{\partial \phi}{\partial x}\) and \(\frac{\partial \phi}{\partial y}\) are plugged into equation(2.20).

It is important to do the reinitialization step after each curve update due to the fact the the level-set function losses accurate track of boundary after each iteration of curve evolution and this results in sharp and/or flat shapes during the evolution, and causes inaccurate computational outcomes.

From practical point of view, the re-initialization process can be quite complicated and computationally expensive. In addition, most of the level-set methods suffer from the problem of how and when to re-initialize the level-set function [21]. Conversely, it was found that any function \(\phi\) satisfying \(|\nabla \phi| \approx 1\) is the signed distance function plus a constant. This dissertation research is
based on regularizing the level-set function using Gaussian kernel filtering after each iteration to eliminate the requirement of re-initialization as in [4].

2.1.4 Level-Set Function-Based Segmentation

Level-Set function-based segmentation is considered to be one of the significant spatially guided approaches, in which the contour is guided by spatial relationships of pixels for segmentation. There have been several choices for the representation of the different phases and their boundaries by level-sets.

Mumford and Shah[22] introduced the concept of variational problems for applications in computer vision; in which, they proposed, by proposing theoretical results of existence and regularity of minimization problems for segmentation using level-set method.

Osher and Sethian[5] proposed algorithms that approximate the equations of motion of propagating fronts, that approximate Hamilton-Jacobi equations with viscosity terms. These approximations were used to compute the solutions to a variety of surface motion problems, due to its flexibility to provide automatic change of topology, such as merging and splitting.

Chan and Vese[15] present active contours without edges method that automatically detects interior contours in which the stopping term does not depend on the gradient of the image, as in the classical active contour models, instead it is related to a particular segmentation of the image. The level-set formulation employs ”mean-curvature flow” like evolving the active contour which will stop on the desired boundary. One of the main advantages of the this method is that the initial curve can be placed anywhere in the image and it does not necessarily surround the objects of interest. Zhao et al. [23] introduced a variational level-set approach to multi-phase motion by combining the level-set method of Osher and Sethian [5] with a theoretical variational formulation of the motion [24]. In this method, the energy function is evaluated entirely in terms of level-set functions.
Samson et al. [25] illustrate a supervised classification model based on a variational approach. In this method, each set of regions and boundaries that is associated to a class is defined by a unique level-set function. The energy functions of these level-sets are defined through the minimization of a unique function; in which, a dynamical scheme is used to consider minimization of the coupled PDEs related to that unique function.

Paragios et al. [26] present a variational method for image segmentation that unifies boundary and region-based information sources under the geodesic active region framework. The boundary information is determined using a probabilistic edge detector, while the region information is estimated using components of Gaussian mixture model. The defined objective function is minimized using a gradient descent method where a level-set approach is used to implement the resulting PDE system.

Brox and Weickert [27] proposed a level-set based minimization scheme for variational segmentation model. The method utilizes divide-and-conquer principle to attain initialization to make the method less sensitive to local minima. The segmentation framework uses one level-set function for each region allowing extraction of an arbitrary number of regions.

Vese et al. [28] designed a multiphase level-set framework for image segmentation using the Mumford and Shah model [22]. This study shows that triple junctions and complex topologies can be presented with a reduced number of level-set functions.

Moreover, Karoui et al. [29] present variational region-level criterion for supervised and unsupervised texture-based image segmentation. Their approach minimizes a cost function comprised of a similarity measure between region features and texture descriptors.

It could be concluded that the variational calculus and Euler-Lagrange equation approaches are fundamental in the energy minimizing scheme in image processing. Gradient descent flow method depends on solving Euler-Lagrange equation by using PDE method. Therefore, the evolution of the
PDE of a level-set function can be directly derived from the problem of minimizing a certain energy function defined by the level-set [3]. In fact, many other authors have studied the minimization of the Mumford-Shah function and related problems for segmentation, both in theory and in practice.

2.1.5 ACM Driven by Local Image Fitting Energy

This method is proposed by Zhang et al. [4] in which, a region-based active contour employing image local information is introduced. The method utilizes the local image information to construct a local image fitting (LIF) energy function in order to complete the segmentation process. The constructed energy function is viewed as a constraint of the difference between the fitting image and the original image. In addition, this method embeds Gaussian filtering for variational level-set to achieve regularization of the level-set function after each iteration and thus, eliminate the need of re-initialization of the level-set function.

In image segmentation, active contours are dynamic curves that move towards the object boundaries. To achieve this goal, an energy function is defined that can move the zero level curve towards the object boundaries. Local image fitting energy (LIF) function [4] is used by minimizing the difference between fitted image and original input image as shown in the following equation:

\[
E^{LIF}(\phi) = \frac{1}{2} \int_{\Omega} |I(x, y) - I^{LFI}(x, y)|^2 dxdy, \quad x, y \in \Omega
\] (2.21)

where \( \Omega \) is the image domain and \( I^{LFI} \) is the local fitted image and is defined as follows:

\[
I^{LFI} = m_1 H_\varepsilon(\phi) + m_2(1 - H_\varepsilon(\phi))
\] (2.22)

where \( H_\varepsilon(\phi) \) is the regularized Heaviside function, and \( m_1 \) and \( m_2 \) are averages of image intensities in a Gaussian window inside and outside the contour, respectively. \( m_1 \) and \( m_2 \) are defined as follows:

\[
m_1 = mean(I(x, y) \in \{(x, y) \in \Omega | \phi(x, y, t) < 0\} \cap W_k(x, y))
\] (2.23)
\[ m_2 = \text{mean}(I(x, y) \in \{(x, y) \in \Omega | \phi(x, y, t) > 0 \} \cap W_k(x, y)) \]  

(2.24)

where \( W_k(x, y) \) is a rectangular window function. An example of a local fitted image (LFI) after different iterations is shown in Figure 2.4. Using the steepest gradient descent method, the following gradient descent flow is obtained:

\[
\frac{\partial \phi}{\partial t} = (I(x, y) - m_1 H_\varepsilon(\phi) - m_2 (1 - H_\varepsilon(\phi)))(m_1 - m_2)\delta_\varepsilon(\phi) = (I(x, y) - I^{LFI}(x, y))(m_1 - m_2)\delta_\varepsilon(\phi)
\]  

(2.25)

where \( \delta_\varepsilon(\phi) \) is the regularized Dirac delta function.

2.2 Active Contour Segmentation with Prior Information

Leventon et al. [30, 31] first introduced the concept of incorporating prior shape information into the level-set evolution function. They provided a portrayal for the deformable shapes and defined a probability distribution over the variances of a set of training shapes. In their method, the prior shape information and the image information are utilized in such a way to estimate the maximum a posteriori (MAP) position that shapes the object in the image at every iteration of the
curve evolution. They used both global and local surface evolutions towards the MAP estimate and image gradients respectively.

Chen et al. [32] provide a variational level-set based segmentation formulation that uses both shape and intensity prior information that are learned from a training set. They utilized an energy function that consists of shape and image energy parts. Image information is specified using regional intensity distributions to get rid of the heuristic weighting factor that balances image energy and shape energy terms. The learned intensity information is integrated into image model using a non-parametric density estimation method in order to yield segmentation for inhomogeneous objects.

Cremers et al. [33] present a segmentation method that combines the non-linear shape statistics with a Mumford-Shah based segmentation process. In their method, training silhouettes are utilized to drive the non-linear shape statistics by a method of density estimation. They implemented a probabilistic framework that is based on kernel Principal Component Analysis (PCA).

Sun et al. [34] developed a level-set method with shape prior to implement a shape-driven image segmentation. They utilize image moments to strip the shape priors of position, scale and angle information, consequently and to obtain the aligned shape priors. Furthermore, they employed the locality preserving projections (LPP) to map shape priors into a low dimensional subspace in which the probability distribution is predicted by using kernel density estimation. The segmentation process is handled using an energy function with shape priors that combines the negative log-probability of shape priors with other data-driven energy items.

Magee and Leibe[35] proposed a framework for the combination of statistical prior information with the level-set for object tracking. Level-set evolution is based on the maximization of a set of likelihoods on mesh values at features, which are located using a stochastic sampling process. Curve evolution is based on the interpolation of likelihood gradients using kernels centered at the features that are based on moments of color histogram.
Rousson and Paragios [36] provide an energetic form to integrate shape constraints to level-set representations. The shape prior construction is done using a variational approach that is based on a shape-to-area principle. They developed a shape-driven propagation energy function that incorporates level-set function with the prior shape information. The minimization of the energy function is done using calculus of variations and motion parameters.

Oktay et al. [37] present a level-set based segmentation method with shape priors that guide the level-set contortions so that the contour extraction process is excited by both the local image properties and the expert knowledge in the form of manual contours. Their system uses manual expert contours to produce new level-set surfaces which are corrupted into the surface from the level-set process. The prior information is incorporated into the level-sets by re-initializing these corrupted surfaces as new level-set surfaces.

2.2.1 Active Contour Segmentation with Neural Networks

Constructing boundaries of objects using neural networks has been introduced in the literature. Tabb et al. [7] introduced a system that integrates active contours with neural networks to track walking humans in the visual field. They utilized a feed-forward error-back propagation neural network as a classifier for the tracked boundary which comes from a snake ACM. This means that neural network is being used in the second phase of the segmentation process to classify the detected boundary. They basically employ active contour models to produce and track the shape of a moving object which is a human body in their case. They start a preprocessing stage with Sobel edge detection followed by motion detection and blob removal procedures to produce the edges of moving objects in the image; which as a consequence, simplifies the active contour model’s task. The tracked human body boundaries are shown in figure 2.5 in which different shapes are produced from different frames. The boundaries produced from the first stage are processed using feed-forward error-back propagation neural network to categorize them as either human or non-human.
objects. They used neural network with various input units and two output units, where one output unit was trained to identify human shapes, and the other non-human shapes.

Shah-Hosseini and Safabakhsh [8] developed a modified form of time adaptive self-organizing map (TASOM) network for active contour modelling. The adaptive Self-organizing map has an individual learning rate and neighborhood function for each neuron in the network. Their method is based on initializing a contour by the user through the selection of a number of control points inside or outside the object boundary, which are then used as the initial weight vectors of a TASOM network. These weight vectors are then modified by the TASOM algorithm in the second stage of the method. The input data, which are used for weight modification, are included in the feature point set \( \{x_1; x_2; \cdots ; x_K\} \) of the object of interest and are represented by the edge points of the object boundary. These points occupy most of the inside and boundary of the object of interest and are detected by an edge detection algorithm. The initial boundary of the active contour is manually selected by the user. For a complex boundary containing long concavities, the convergence will take longer in comparison to a simple object boundary. The number of epochs on the feature points that is needed for convergence of the contour depends on various factors. The speed of convergence depends on values of the lowest and highest allowed distances between any two adjacent neurons. In addition, it depends on speed parameters that control the movement speed of the control points toward the object boundary. Their method is limited in that it assumes that there is only one object
per image and would not succeed if the object is placed in inhomogeneous background since the method depends on detecting the edges of object boundary. In addition, TASOM is more complex than the self-organizing map (SOM) network which means that it is more time consuming and each iteration of the TASOM algorithm takes more time than that of the basic SOM.

Venkatesh et al. [9] proposed self-organizing neural networks based on spatial isomorphism for active contour modelling. They utilized the basis of spatial isomorphism and self-organization in order to generate flexible boundaries that define shapes in images. Their implemented model is semi-automatic, in the sense that a user-interface is needed for initializing the process. In their method, a neural network isomorphic to an initial contour is constructed, and subjected to deformation in order to map onto the nearest salient contour in the image. The correspondence between the salient contour and the network is established by mapping the latter onto the former by using the self-organization scheme. Their method starts by computing the edge map of the test image; then, the initial contour points are set using an initialization scheme. For static images, the initialization is done using generalized Hough transform, while optical flow analysis or image differencing techniques are used for the initialization of video sequences. After that, the region of interest is chosen according to the location of the initial contour. Then, a SOM network is constructed isomorphic to the initial contour. Due to the reason that this contour model starts with edge detection then it yields a continuous set of points that makes it hard to distinguish an object if it is placed in a complex background.

2.3 The Self-Organizing Map (SOM)

Self-organizing map (SOM) or self-organizing feature map (SOFM) is an Artificial Neural Network (ANN) that is alternatively known as Kohonen map. This ANN is introduced by Kohonen [38] and it has the special property of effectively creating spatially organized internal representations of
various features of input data. A SOM consists of neurons organized on a grid in which, each neuron is a $d$-dimensional weight vector where $d$ is equal to the dimension of the input vectors. The grid forms what is known as the output space, while the input space contains the data patterns (Figure 2.6). SOM mapping creates topological relations between the input patterns and output nodes through relevant weighting scheme that designates the strength of various connections.

![Figure 2.6: Self-organizing map architecture.](image)

After the training phase, mapping phase would allow to classify a random new input sample according to the constructed feature map. The basic $SOM$ training method can be described as follows:

1. Let vector $U$ of $n$ dimensions: $u_1, u_2, u_3, \cdots, u_n$ be the set of training patterns.

2. Let grid $W$ of nodes $w_i$ be a set of grid nodes’ weights.

3. Let $L$ be the learning rate, where $0 < L < 1$, initialized to a given initial learning rate.
4. Let $\sigma_0$ be the radius of the neighborhood function at time $t_0$.

5. Each node’s weights are initialized.

6. A vector is chosen at random from the set of training patterns and presented to the grid.

7. Every node is examined to calculate which one’s weights are most like the input vector. The winning node is commonly known as the Best Matching Unit (BMU).

8. The radius of the neighboring of the BMU is updated. This is a value that starts large, typically set to the radius of the grid, but diminishes each time-step. Any nodes found within this radius are deemed to be inside the BMU’s neighborhood as shown in Figure 2.7.

![Figure 2.7: BMU’s neighbourhood.](image)

9. Each neighbouring node’s (the nodes found in step 8) weights are adjusted to make them more like the input vector. The closer a node is to the BMU, the more its weights get altered.

10. Repeat steps 6-9 for N iterations.

The BMU is determined by calculating the Euclidean distance between each node’s weight vector and the current input vector. The node with a weight vector closest to the input vector is
tagged as the BMU as:

\[ BMU = \min_i \{||u - w_i||^2\} \]  

(2.26)

where \( u \) is the current input pattern and \( w_i : i = 1, \ldots, l \) is the node’s weight vector. Usually Euclidean metric is used, although other choices are possible as well. Every node within the BMU’s neighborhood has its weight vector adjusted according to Equation (2.27).

\[ w_i(t + 1) = w_i(t) + \Theta(t)L(t)[u(t) - w_i(t)] \]  

(2.27)

where \( t \) is the time step, \( L(t) \) is the learning rate, and \( \Theta(t) \) is the amount of influence that a node’s distance from the BMU has on its learning. \( L(t) \) is defined as:

\[ L(t) = L_0 \cdot \exp\left(\frac{-t}{\lambda}\right); \quad t = 1, 2, 3, \ldots \]  

(2.28)

where \( L_0 \) is the learning rate at time \( t_0 \). \( \Theta(t) \) is defined as:

\[ \Theta(t) = \exp\left(\frac{-||u - w_i||^2}{2\sigma(t)^2}\right); \quad t = 1, 2, 3, \ldots \]  

(2.29)

where \( \sigma(t) \) is the radius of the neighborhood function at time \( t \). Over time the neighborhood will shrink to the size of just one node, which is the BMU.

![Figure 2.8: Shrinking neighbourhood radius.](image-url)
2.3.1 Self-Organizing Map-Based Segmentation

SOM gained a huge popularity for classification applications as it is known to be most closely model of how the human brain actually works. It has been used in several applications such as speech recognition, pattern recognition, process control, robotics, processing semantic information, and many more. Abdelsamea et al. [6] illustrated a Concurrent Self Organizing Map Chan-Vese based model (CSOM-CV) which concurrently combined the pixel information extracted by a concurrent SOM into the level-set framework of the Chan-Vese (C-V) model [39] to build an ACM segmentation approach. Yao et al. [40] presented an unsupervised two-step high resolution sonar images segmentation in which SOM segmentation is compared with Markovian segmentation. The study presented simplicity and robustness of SOM scheme compared to Markovian methods. Brown et al. [41] developed a SOM based approach to skin detection in real time systems. They used a combination of SOM approach with fast versatile hardware. Hang et al. [42] constructed a color image segmentation methodology based on SOM. They incorporated color similarity and spatial relationship into the segmentation process which uses two phases of SOMs. In the first phase, color information based on human vision perception is extracted, while in the second step, spatial information based on computing the spatial distance between any two color planes obtained from first stage was taken to classify the color planes into segmented clusters. Li and Chi [43] developed an MR brain image segmentation based on SOM map network. They aimed to identify the principal tissue structures by combining Markov Random Field (MRF) that provides spatial constraints within the SOM network for image segmentation. Li et al. [44] designed an adaptive segmentation of color images. Colour feature vector is determined with the aid of a SOM network and then fuzzy clustering is used to provide segmentation for different types of color images. Demirhan et al. [45] demonstrated a study that combines stationary wavelet transform and SOMs for brain MR image segmentation. Multi-resolution information is obtained using the stationary wavelet transform.
(SWT) and it is used for distinguishing different tissues into multidimensional feature vector. The feature vector is then used as input to a SOM to segment images in an unsupervised approach. Ilea and Whelan [46] constructed an unsupervised image segmentation framework that is based on the adaptive inclusion of color and texture in the process of data partitioning. The method is referred to as CTex. The dominant colors and estimation of the optimal number of clusters in the $RGB$ and $YIQ$ color spaces of the input image were computed using unsupervised SOM classifier. In the second phase of the algorithm, texture features are extracted using a multichannel decomposition scheme based on Gabor filtering. The association of the color and texture features provided a description of the image regions with homogeneous characteristics. Ortiz et al. [47] instituted a segmentation technique using 3D statistical features extracted from a volume image. Their method is based on unsupervised vector quantization and fuzzy clustering techniques. The statistical feature vectors are modelled using SOM to reduced the feature space to a number of prototypes that are grouped together using a fuzzy c-means (FCM) algorithm. Dong et al. [48] introduced a segmentation system that constitutes both unsupervised and supervised segmentation that measures the color difference in a modified $L^*u^*v^*$ color space. The method achieves color reduction in which image colors are projected into a small set of prototypes using a SOM network.

2.4 Lattice Boltzmann Method (LBM)

The lattice Boltzmann method (LBM) is a mathematical technique that is initially developed to simulate the flow of fluid systems based on the Boltzmann equation [12]. The enforcement of LBM in boundary conditions started from the fact that fluid dynamics extremely rely on the surrounding environment, which is mathematically defined through the description of boundary conditions. The LBM approach has been widely used as an alternative to conventional fluid solvers because it can model the collision behavior of fluids in an approximated discrete form of physical space and time. Moreover, this method is capable of solving complex non-linear Partial Differential
Equations (PDE)s by discretizing space and time. The discretization is made by discretizing space in grids (lattices) and discretizing time in time steps, and allowing the particles to move from one lattice node to another. Thus, the physical space of the LBM consists of set of uniformly spaced nodes in a lattice that characterize the particles, and a corresponding set of discrete microscopic velocities of particles along with the particle distributions. The LBM models Boltzmann particle dynamics using different lattice structures in 1D, 2D, or 3D as shown in Figure 2.9. A $D1Q3$ lattice means one dimensional lattice and three lattice speeds, a $D2Q9$ lattice means two dimensional lattice with nine lattice speeds, and a $D3Q19$ lattice means three dimensional lattice with nineteen lattice speeds. As described earlier, in the LBM lattice structure, each link has its velocity vector $e_i$

$$f_i(\vec{r}, t + 1) = f_i(\vec{r}, t) + \tau \left[ f_i^{eq}(\vec{r}, t) - f_i(\vec{r}, t) \right]$$

(2.30)

Figure 2.9: Lattice structure; (a) 1D structure (D1Q3); (b) 2D structure (D2Q9); (c) 3D structure (D3Q19).
where $\tau$ is the relaxation time that plays an essential role in the stability and accuracy of the LBM and it determines the fluid’s kinematic viscosity $\vartheta$, given by:

$$
\vartheta = \frac{1}{3}(\tau - \frac{1}{2}).
$$

(2.31)

and $f_{eq}^i$ is a simplified local equilibrium particle distribution that models collisions as a statistical redistribution of momentum, and locally drives the system toward equilibrium while conserving mass and momentum. It is given by the Bhatnager, Gross, Krook (BGK) model [50] as:

$$
f_{eq}^i(\rho, u) = \rho(A_i + B_i(e_i \cdot u) + C_i(e_i \cdot u)^2 + D_i u^2).
$$

(2.32)

where $A_i, B_i, C_i, D_i$ are constant scalar coefficients specific to the chosen lattice geometry, $\rho$ is the mass, and $\rho u$ is the momentum. In the traditional LBM, Equation (2.32) is simplified by eliminating the momentum dependency [49] as the following:

$$
f_{eq}^i(\rho, u) = A_i \rho.
$$

(2.33)

where $\rho = \sum_i f_i$.

In the case of a $D2Q9$ model, $A_i = 4/9$ for the zero link, $A_i = 1/9$ for the axial links, and $A_i = 1/36$ for the diagonal links. In Equation (2.30), the equilibrium and the forcing terms are the key to recover a specific PDE.

### 2.4.1 Lattice Boltzmann Active Contour Segmentation

The level-set active contour method utilizes an LSF to characterize the evolution of the zero level-set; in which a PDE is solved. Conventional approaches apply explicit computation of the curvature term in the LSF as in [5, 19–21, 23]. Since the birth of the lattice Boltzmann method, it became an alternative approach for solving the level-set equation because it can implicitly solve the curvature term that characterizes the contour in the LSF and thus increases the computation speed.
Balla-Arabé et al. [13] presented a level-set region based image segmentation using lattice Boltzmann method. They presented a design of a Signed Pressure stopping Function (SPF) that is based on Chan-Vese ACM model [39] and LBM is utilized to solve the SPF to achieve convergence of the active contour. Their method provides good segmentation results with lower computational time, however, the problem of sticking into local minima is still there resulting in undesired segmentation results when there is a variety of intensity variations in the image. Moreover, Balla-Arabé et al. [14] introduced an energy function based on a fuzzy c-means objective function and LBM for image segmentation. LBM parallelization property is used for solving the level-set equation to achieve segmentation of objects. Again, because this method is not a learning-based method, the problem of over segmentation is still there resulting in undesired segmentation results.

2.5 Summary

This chapter provided an overview of the active contour models, learning-based active contour models, and Lattice Boltzmann optimization method. The majority of learning-based ACM techniques are based on the existing state-of-the-art ACMs (edge based and Chan-Vese based ACMS). Although ACMs often provide a sufficient concept to extract smooth and well-defined contours, it still suffers from the problem of sticking into local minima of the energy function, causing over-segmentation and resulting in poor segmentation output specially when the image contains varying shades of colors. Thus, one of our objectives is to overcome this problem. Moreover, the level-set ACM method is just a dummy method that would not know the object of interest especially if it is surrounded with other objects and complex backgrounds. Therefore, in order to direct this minimization technique towards the desired object of interest, the step of giving it a prior information about that object would help a lot to capture that specific object. From the survey of existing literature, it was observed that the quality of the segmentation output can be increased by embedding the local image characteristics into the active contour cost function. In this research, the problem
of considering local image information in the learning-based active contours energy function is in focus.
CHAPTER III

SELF-ORGANIZED LEARNING BASED ACTIVE CONTOURS (SOLAC) SEGMENTATION

In general, segmentation is the first stage in any attempt to analyze or interpret an image automatically. It is considered to be an essential task for post-processing stages like object recognition and classification. Previous acquaintance on the characteristics of the object of interest can significantly facilitate the segmentation procedure, especially for objects of inhomogeneous color components. However the combination of object characteristics information with the segmentation process is non-trivial. The main difficulty is the need to account for possible transformations between the prior object features and the testing image containing the object to segment.

We suggest a modified level-set prior-based segmentation approach that integrates self-organized map based neural networks with the level-set active contour models for boundary extraction of objects in cluttered environments. This proposed method is named self-organized learning based active contour (SOLAC). One merit of SOLAC is that small seed patches representing the object of interest and other small seed patches representing the background region from one single reference image is enough to complete the training process. The outcomes of the algorithm include the detection of the object/objects of interest and correct extraction of the boundaries.

By employing self-organizing map neural networks within the level-set formulation; we gain a significant advantage over regular level-set active contour model segmentation. Our suggested
method is capable to direct the level-set function towards the boundary extraction of the desired object. In addition, our method is independent of the shape of the object of interest; therefore, it works well with different complicated shapes of objects. The outcome of the mapping process of the neural network is perfectly transformed on the energy function of the level-set to achieve accurate segmentation. This results in an elegant and powerful mathematical formulation to align the prior intensity information and the evolving object boundary. The elegant combination of the image data with that of the prior information is the essence of the proposed contribution. The suggested algorithm is also tested on a variety of challenging images, including natural imagery, hyperspectral imagery, medical imagery, and thermal imagery. The effective segmentation results and the reliable boundary extraction acknowledge this method as a capable tool for various segmentation applications.

Our method aims at boundary and region extraction of objects in both gray and color imagery. The first step in our method includes clustering the intensity and/or color information of both the object of interest and its background through unsupervised SOM technique to produce two SOM maps; one SOM network to represent the object of interest, and another SOM network representing the background region. The trained networks are employed into the second stage of our segmentation process to map the intensity levels of an input testing image. The mapped testing neurons are then utilized into the evolving curve energy function of a level-set protocol in the third stage of our proposed approach. Figure 3.1 gives more explanation of our method. A functional framework of the proposed segmentation method is shown in Figure 3.2.

Self-organizing map (SOM) is used to train input samples of both foreground and background regions. The input samples are small seed patches from the objects of interest and other seed patches from the inhomogeneous background regions. Thus, SOM is set to transform an incoming signal
Figure 3.1: SOLAC system diagram.
Figure 3.2: The functional framework of the proposed SOLAC segmentation system.
pattern of an arbitrary dimension into a lower dimensional space lattice - usually one or two dimensions - by creating a set of prototype vectors representing the features of that pattern.

The most optimum boundary will be the boundary that holds the complete object region of interest even if it has various intensity and/or color variations. Thus, that boundary would be the most proper one for subsequent processes such as scene identification and interpretation. The segmentation process is facilitated by the construction of a cost function; in which, the curve evolution is represented by the zero level of a level-set function that consumes local image information in order to get the optimum boundary and region representation of the object of interest. The problem of minimizing the cost function is handled using calculus of variations [22]. For segmenting color images, we use a three-dimensional SOMs as colors are represented with three components. Our method is based on the locally image fitting level-set energy function [4]. Since the level-set equation contains parts describing foreground object and background regions, we start our method by specifying seed patches describing foreground object of interest and other seed patches representing the background region. By introducing the SOM image fitting energy to highlight the object of interest, our model is able to segment objects in images with intensity inhomogeneities. Moreover, a Gaussian filtering for variational level-set is used to regularize the level-set function, to ensure the smoothness of the level-set function, and to eliminate the requirement of re-initialization, which is computationally expensive.

3.1 SOM Training Phase

We follow the steps below to complete the training process of the SOM.

1. Let vectors $P_o$ of $n$ dimensions: $\{p_o1, p_o2, p_o3, \cdots, p_on\}$ be the set of object training patterns and $P_b$ of $n$ dimensions: $\{p_b1, p_b2, p_b3, \cdots, p_bn\}$ be the set of background training patterns.
2. Let $W_o$ be the object grid with nodes $w_{oi,j}$ where $i$ and $j$ are object nodes’ coordinates, and $W_b$ be the background grid with nodes $w_{bkl}$ where $k$ and $l$ are background nodes’ coordinates.

3. Let $L_o$ be the object’s network learning rate, and $L_b$ be the background’s network learning rate, such that $0 < L_o, L_b < 1$.

4. Let $\sigma_o$ and $\sigma_b$ be the radii of the neighborhood functions for both object and background networks, respectively.

5. Each node’s weights are initialized to small normalized random values.

6. A vector is chosen at random from the set of training patterns and presented to its corresponding grid.

7. Every node is examined to calculate which one’s weights are most similar to the input vector. The winning node is commonly known as the Best Matching Unit (BMU).

8. The radius of the neighboring node of the BMU is now updated. This is a value that starts large, typically set to the radius of the grid, but diminishes in each time-step. All nodes found within this radius are deemed to be inside the BMU’s neighborhood.

9. Each neighboring node’s (the nodes found in step 8) weights are adjusted to make them more similar to the input vector. The closer a node is to the BMU, the more its weights get altered.

10. Repeat steps 6 - 9 for $N$ number of iterations.

At the end of the training process we will have two sets of BMUs; one representing the object and the other representing the background region.
3.2 SOM Mapping Process

At this stage, we utilize our trained SOMs to retrieve the object of interest and its background from the testing image. This is considered to be an opposite process from the training stage. However, here we input a whole image containing different objects and background. We employ both SOMs from the training phase as inputs to our modified level-set energy function to attain desired object region and boundary.

3.2.1 Integrating Retrieved Prior Information in Level-Set Cost Function

Our modified level-set function is based on the local image fitting energy function described in section 2.1.5. We embed the clustered data coming from SOMs into the level-set function. We use the energy equation:

\[ E_{SOM}(\phi) = \frac{1}{2} \int_{\Omega} |I(x,y) - I_{SOM}(x,y)|^2 dxdy, \quad x,y \in \Omega \]  \hspace{1cm} (3.1)

where \( I_{SOM} \) is the our new self-organizing map fitted image and is defined as follows:

\[ I_{SOM} = s_1 H_\epsilon(\phi) + s_2(1 - H_\epsilon(\phi)) \]  \hspace{1cm} (3.2)

where \( H_\epsilon(\phi) \) is the regularized Heaviside function given in Equation (2.15), and \( s_1 \) and \( s_2 \) are defined as follows:

\[ s_1 = W_o - \text{mean}(I \in \{(x,y) \in \Omega|\phi(x,y,t) < 0\} \cap G_k(x,y))) \]  \hspace{1cm} (3.3)

\[ s_2 = W_b - \text{mean}(I \in \{(x,y) \in \Omega|\phi(x,y,t) > 0\} \cap G_k(x,y))) \]  \hspace{1cm} (3.4)

where \( W_o \) and \( W_b \) are the best matching neurons of the foreground object region and background region respectively, \( G_k(x,y) \) is a rectangular Gaussian window with width \( k \), and \( s_1 \) and \( s_2 \) are the level-set foreground and background parameters, respectively. Equation (3.1) is minimized with
respect to \( \phi \) to get the corresponding gradient descent flow using the calculus of variation and the steepest method as shown below [4]. The variation \( \eta \) is added to the level-set function \( \phi \) such that \( \phi = \phi + \varepsilon \eta \). Keeping \( s_1 \) and \( s_2 \) fixed, differentiating with respect to \( \phi \), and letting \( \varepsilon \to 0 \), we have:

\[
\frac{\delta E^{SOM}(\phi)}{\delta \phi} = \lim_{\varepsilon \to 0} \frac{d}{d\varepsilon} \left( \frac{1}{2} \int_{\Omega} |I(x,y) - s_1 H_\varepsilon(\phi) - s_2 (1 - H_\varepsilon(\phi))|^2 dxdy \right)
\]

\[
= \lim_{\varepsilon \to 0} \left( - \int_{\Omega} [I(x,y) - s_1 H_\varepsilon(\phi) - s_2 (1 - H_\varepsilon(\phi))] (s_1 - s_2) \delta_\varepsilon(\phi) \eta dxdy \right)
\]

\[
= - \int_{\Omega} [I(x,y) - s_1 H_\varepsilon(\phi) - s_2 (1 - H_\varepsilon(\phi))] (s_1 - s_2) \delta_\varepsilon(\phi) \eta dxdy
\]  

(3.5)

Therefore, the Euler-Lagrange equation is obtained as follows:

\[
-[I(x,y) - s_1 H_\varepsilon(\phi) - s_2 (1 - H_\varepsilon(\phi))] (s_1 - s_2) \delta_\varepsilon(\phi) = 0
\]  

(3.6)

by the steepest gradient descent method, the following gradient descent flow is obtained:

\[
\frac{\partial \phi}{\partial t} = (I(x,y) - s_1 H_\varepsilon(\phi) - s_2 (1 - H_\varepsilon(\phi))) (s_1 - s_2) \delta_\varepsilon(\phi)
\]

\[
= (I(x,y) - I^{SOM}(x,y)) (s_1 - s_2) \delta_\varepsilon(\phi)
\]  

(3.7)

The main steps to apply level-set segmentation are summarized as follows:

1. Initialize the level-set function \( \phi \) as a binary function.

2. Evolve the level-set function as shown in Equation (3.7).

3. Smooth the level-set function \( \phi \) using a Gaussian kernel \( G_\zeta \), with standard deviation \( \zeta \), i.e. \( \phi = G_\zeta * \phi \). It is important to note that in order to enhance the smoothing capacity, \( \zeta \) should be larger than the square root of the time-step \( \Delta t \) [4].

4. Check whether the evolution is immobile, otherwise repeat steps 2-3.

### 3.3 Results and Evaluation

Proper evaluation of the proposed SOLAC method is a critical step to assess the segmentation performance. In addition, the choice of the datasets can have a tremendous impact on the overall
results because our segmentation technique depends on the local image properties and the prior knowledge of the object/objects of interests and the background. In order to confirm the performance of our SOLAC method, we conducted evaluation on challenging visible imagery dataset (PASCAL 2011 [1]) in which, images contain objects with variations of illuminations, shadows, and clutter. We also tested the effectiveness of the algorithm on hyperspectral imagery, medical imagery and thermal imagery. The hyperspectral imagery are natural images taken using Resonon Pika II hyperspectral camera, which provides 240 spectral channels that range from 400-900nm with 2.1nm spectral resolution. The medical imagery are taken from datasets composed of X-ray imagery of bones, lungs, livers, and brains. These datasets were selected from the publicly available online servers hosted by The Cancer Image Archives (TCIA) [51] and by Life In The Fast Lane (LITFL) [52] Radiology Image Database. The thermal imagery are real-life images that are taken by a FLIR T650sc thermal camera, which provides high sensitivity of temperature measurements.

3.3.1 Evaluation Strategies

When comparing a segmentation method outputs to the ground truth images, there are five possible outcomes that need to be identified. The segmentation method can either (a) correctly segment a region, (b) over-segment a region, (c) under-segment a region, (d) miss a region, or (e) incorrectly segment a noise region. Therefore, it is an essential task to calculate true positives, true negatives, false positives, and false negatives - also known as confusion matrix- to evaluate segmentation performance. These are defined as follows:

**True positive (TP):** a pixel that belongs to the expert segmented region and was detected as ”object-of-interest” by the algorithm;

**True negative (TN):** a pixel that does not belong to the expert segmented region and was detected as ”non-object-of-interest” by the algorithm;

**False positive (FP):** a pixel that does not belong to the expert segmented region and was detected
as "object-of-interest" by the algorithm;

**False negative (FN):** a pixel that belongs to the expert segmented region and was detected as "non-object-of-interest" by the algorithm.

In addition, for evaluating segmentation methods, the following factors are considered:

**Recall (RE),** measures the accuracy of the system to recognize positive cases;

\[
RE = \frac{TP}{(TP + FN)} \quad (3.8)
\]

**Specificity (SP)** measures the accuracy of the system to recognize negative cases;

\[
SP = \frac{TN}{(TN + FP)} \quad (3.9)
\]

**Precision (P),** is the proportion of the predicted positive cases that were correct;

\[
P = \frac{TP}{(TP + FP)} \quad (3.10)
\]

**False positive rate (FPR),** defined as:

\[
FPR = \frac{FN}{(FN + TP)} \quad (3.11)
\]

In addition; the **False negative rate (FNR),** defined as:

\[
FNR = \frac{FP}{(FP + TN)} \quad (3.12)
\]

An accurate segmentation system usually have high values of recall (RE) specificity (SP) and precision (P), while the values of the false positive rate (FPR) and false negative rate (FNR) should be small.

### 3.3.2 Evaluation of SOLAC on Visible Imagery

Given that the proposed segmentation seeks to extract object region and boundaries in complex background environments, the best database for testing would be a database which contains a
large variety of natural images of objects with different shapes, colors, shades and illuminations. PASCAL-2011 [1] challenging database is chosen for our tests in this category. In Figure 3.3 we depict a series of segmentation results of an airplane image using different segmentation methods. Assuming that the airplane is the object of interest, this image is considered challenging because both the object of interest and its background have a variety of colors. Figure 3.3(b) shows Canny edge detection result indicated in the blue boundaries. The result obtained from level-set ACM is shown in Figure 3.3(c) illustrated with the red boundary. Figure 3.4(a) shows segmentation result of our proposed SOLAC method.

Figure 3.3: Comparison of segmentation results of different algorithms; (a) input image (b) Canny edge detection output; (c) level set segmentation output; (d) proposed SOLAC segmentation result.
By comparing all three segmentation outputs, we can indicate that Canny edge detection produces a large amount of false positives that destroy the boundary of the object of interest. In addition, the resultant boundaries are not fully connected. The level-set ACM and our SOLAC method outputs are closer to the actual object of interest. Even though the output boundary did not include the whole torso of the airplane, the outer boundary shape still indicates an airplane. Thus, the latter two outputs can still be useful for object identification, classification and other post-processing stages. Once we train our learning-based SOLAC model to segment object/objects of interest, the same model is capable to be applied in other images having similar object/objects of interest. Figure 3.4 shows segmentation result using the same trained SOLAC model applied in another image that contains the same object of interest.

![Figure 3.4: Segmentation result of SOLAC in a new testing image; (a) input image (b) segmentation output.](image)

Figure 3.5 shows segmentation results of a swan being as the object of interest. Figure 3.5(b) shows Canny edge detection output. The result obtained from level-set ACM is shown in figure 3.5(c). Figure 3.5(d) shows segmentation result of our proposed SOLAC method.
Figure 3.5: Comparison of segmentation results of different algorithms; (a) input image; (b) Canny edge detection output; (c) level set segmentation output; (e) result using our proposed SOLAC method.
It is obvious that the Canny edge detection output produces large amount of false positives around the object of interest. The level-set segmentation result and our proposed SOLAC output both produce good results fitting the object of interest. Figure 3.6 shows segmentation result using the same trained SOLAC model applied in another image having similar category of the object of interest.

![Segmentation result](image)

Figure 3.6: Segmentation result of SOLAC in a different testing image than the training image; (a) input image (b) segmentation output.

In Figure 3.7 we depict a series of segmentation results of sheep image appearing in a complex background. The objects of interest in this image are the sheep. It is significant to notice that our objects of interest have non-homogeneous texture and disparate color shades. In addition, the grass around the sheep introduce complexity to the background region. Figure 3.7(b) shows the result of Canny edge detection. Figure 3.7(c) shows the result of non-learning based level-set ACM method. Figure 3.7(d) shows the result of our SOLAC method. Figure 3.8 shows segmentation results obtained to segment sheep appearing in a different testing image than the training image.

Thus, Canny edge detection is a less effective choice for applications of object identification and recognition since this method detects all of the edges in the image along with the edges of
Figure 3.7: Comparison of segmentation results of different algorithms; (a) input image (b) Canny edge detector; (c) level-set ACM method; (d) SOLAC segmentation method result.

Figure 3.8: Segmentation result of SOLAC in a different testing image than the training image; (a) input image (b) segmentation output.
the object/objects of interest. Although, level-set ACM method shows great results in the cases where the object/objects of interest are not appearing in a complex background, this method still suffers from producing false positives due to the local minima problem, specially when the object of interest is non-homogeneous in color and texture. Our proposed SOLAC approach is superior compared to both the Canny edge detection and non-learning based level-set ACM methods. Figure 3.9 illustrates segmentation result of the proposed SOLAC method for person detection. This image is also considered challenging since the object of interest (the person) and the surrounding background both consist of a large variety of shades and colors. Our method shows great performance in extracting the person’s region and boundary despite missing the face region and producing some false positives as shown in Figure 3.9(b).

![Figure 3.9: Segmentation using SOLAC for human detection application](image)

(a) (b)

The same previous image is used for shadow segmentation. This is performed by training the foreground SOM to detect the shadow as our object of interest and training the background SOM to
detect anything else other than the shadow regions. Figure 3.10 demonstrates the result of shadow segmentation.

![Image](image.png)

Figure 3.10: Segmentation result using SOLAC for shadow detection application; (a) input image; (b) result using our method.

Table 3.1 illustrates segmentation metrics produced after applying our SOLAC method in a 100 randomly selected natural images from the PASCAL-2011 dataset [1].

<table>
<thead>
<tr>
<th>RE %</th>
<th>SP %</th>
<th>P %</th>
<th>FPR %</th>
<th>FNR %</th>
</tr>
</thead>
<tbody>
<tr>
<td>69</td>
<td>55</td>
<td>42</td>
<td>44</td>
<td>30</td>
</tr>
</tbody>
</table>

3.3.3 Application of SOLAC to Different Types of Imagery

In order to evaluate our SOLAC segmentation approach we also conducted experiments using three other types of imagery viz. hyperspectral imagery, medical imagery and thermal imagery.
Object segmentation in Hyperspectral imagery

Hyperspectral sensors provide the spectrum for each pixel in the image, helping computer vision researchers in many areas like object detection, segmentation, identifying materials, and many more applications [53–56]. Hyperspectral imagery is chosen to be one of our testing sets to evaluate our proposed SOLAC approach. We use the Resonon Pika II hyperspectral camera to capture natural images. This camera provides 240 spectral channels that range from 400-900nm with 2.1nm spectral resolution. Knowing the fact that hyperspectral images consist of a large amount of bands, this type of imagery needed preprocessing before applying our SOLAC technique. We start with feature extraction from raw hyperspectral images by employing Principal Component Analysis (PCA) transformation, and this actually helps in reducing dimensionality and aids in selecting the best sets of the significant spectral bands. Then, our SOLAC segmentation is applied on the optimal number of spectral bands determined by the PCA [10]. In the following set of Figures, we illustrate the effectiveness of applying our SOLAC approach in both hyperspectral and regular RGB images. The only difference between the two imagery types is that the hyperspectral images have been preprocessed by applying PCA approach as explained earlier. Figure 3.11 shows a set of results for segmenting bushes in an outdoor scene. Figure 3.11(a) shows the RGB bands of the input hyperspectral image, Figure 3.11(b) and 3.11(c) are the results of using the proposed SOLAC method in both the RGB and hyperspectral versions of this image, respectively. We achieved better results with less false segmentation in the hyperspectral version of this image. In other words, efficiently utilizing hyperspectral information boosts the segmentation results. Figure 3.12 shows another a set of results for segmenting vegetation in an outdoor scene. Figure 3.12(a) shows the RGB bands of the input hyperspectral image, Figure 3.12(b) and 3.12(c) are the results of using the proposed SOLAC method in both the RGB and hyperspectral versions of this image, respectively. Although both results show
Figure 3.11: Segmentation results in a hyperspectral image (a) RGB bands of input hyperspectral image; (b) segmentation result of the RGB image; (c) segmentation result of the hyperspectral image.
acceptable detection of the objects of interest, more accurate segmentation is accomplished in the hyperspectral version of the image.

Table 3.2 illustrates segmentation metrics produced after applying our SOLAC method in five hyperspectral images. The table show high performance of our proposed SOLAC approach.

Table 3.2: Average segmentation statistical results for five hyperspectral images.

<table>
<thead>
<tr>
<th>RE %</th>
<th>SP %</th>
<th>P %</th>
<th>FPR %</th>
<th>FNR %</th>
</tr>
</thead>
<tbody>
<tr>
<td>93</td>
<td>99</td>
<td>98</td>
<td>7.0</td>
<td>6.0</td>
</tr>
</tbody>
</table>

Object segmentation in medical imagery

There exist several medical imagery types ranging from simple X-rays to more advanced categories such as CT, MRI, and fluoroscopic techniques for visualization in real-time. In order to achieve accurate segmentation results for this critical type of imagery, our SOLAC approach is morphed with Seeded Region Growing (SRG) approach forming a new segmentation framework [11] shown in Figure 3.13.

The proposed algorithm was tested on a 50 image dataset composed of X-ray imagery of bones, lungs, livers, and brains. The datasets were selected from the publicly available online servers hosted by The Cancer Image Archives (TCIA) [51] and by Life In The Fast Lane (LITFL) [52] Radiology Image Database. Each image in the database was truthed by medical students who have completed their radiology clinical rotation. Figure 3.14 illustrates the several types of imagery used in the experiments. The regions marked in red are identified as areas of potential carcinoma and therefore regions of interest.

In addition to varying the type of imagery, we tested the algorithm on a series of lung scans taken from four patients. This allowed us to assess the feasibility of training the algorithm using
Figure 3.12: Segmentation results in a hyperspectral image (a) RGB bands of input hyperspectral image; (b) segmentation result of the RGB image; (c) segmentation result of the hyperspectral image.
Figure 3.13: Flowchart of the SOLAC segmentation method in medical imagery.
Figure 3.14: Several experimental imagery types; (a) Bone X-ray with abnormal tissue growth; (b) Region of interest is marked in red; (c) Brain CT scan with visible dense tissue regions; (d) Regions of interest, potential carcinoma, is marked in red in the CT image. Raw imagery obtained from [52] and [51].

Figure 3.15: (a), (b), and (c) are CT scans of three patients with tumors in their left lung. The shape and characteristics of each scan can drastically differ between patients. Raw imagery obtained from [52] and [51].

one set of scans and testing using the scans of the other three patients. As can be seen in Figure 3.15,
Figure 3.16: (a)-(c) Lung tumour in right lung. Manual truthing is illustrated in red, the computed boundary is marked in green, and the binary image mask indicates the size and area of the detected tumour.

The shape of the lungs and characteristics of the tumor areas can significantly differ from person to person.

In Figure 3.16, we illustrate the lung scan with the tumor marked in red. In Figure 3.16b the proposed algorithm computes a boundary around the tumor marked in green. By transforming the boundary image into a binary image mask we obtain the final tumor mask in Figure 3.16c.

Similarly, we illustrate the results obtained in segmenting carcinoma in the liver scan as shown in Figure 3.17a. In Figure 3.17b the algorithm computes a boundary around the tumor marked in green. The binary image mask is shown in Figure 3.17c. Segmentation of carcinoma in brain scan is illustrated in Figure 3.18. In Figure 3.18b the algorithm computes a boundary around the tumor marked in green. The final tumor mask is illustrated in Figure 3.18c. Finally, segmentation of carcinoma in bone scan is depicted in Figure 3.19. The boundary of tumor region is illustrated in Figure 3.19b and the final binary mask is shown in Figure 3.19c.
Figure 3.17: (a)-(c) Abdominal CT scan of a liver with intrahepatic lesion.

Figure 3.18: (a)-(c) CT scans of the brain showing large anomaly in the frontal lobe.

In addition to visual inspection, we evaluate our algorithm by computing the true positive and true negative percentages of the detected and segmented regions along with segmentation evaluation metrics. The metrics are showing in Tables 3.3 and 3.4. From those tables we can conclude that the
proposed algorithm produces promising results in liver and brain and has shortcomings in segmenting the bones due to the complexity of the regions of interest, which will be further analyzed and improved in future work.
Object segmentation in thermal imagery

Thermal images are exploited in many areas of pattern recognition and classification applications [57–60]. Our proposed SOLAC approach is applied in thermal images taken by a FLIR T650sc thermal camera, which provides high sensitivity of temperature measurements when capturing real-life images. Figure 3.20 demonstrates segmentation result with the person being as object of interest. Figure 3.20(a) is the input thermal image, and Figure 3.20(b) is the result obtained using the proposed SOLAC approach. This result can be used for human detection applications. Figure 3.21 shows another thermal image segmentation with cars being as objects of interest. Furthermore, Figure 3.22 shows people segmentation results. Table 3.5 illustrates average statistical values of the segmentation results of 66 thermal images.

Figure 3.20: Thermal imagery segmentation for person detection (a) input image; (b) result using SOLAC method.
Figure 3.21: Results of car segmentation; (a) input thermal image; (b) segmentation result; (c) segmentation mask.

Figure 3.22: Results of people segmentation; (a) input thermal image; (b) segmentation result; (c) segmentation mask.

Table 3.5: Average segmentation statistical results for 66 thermal images.

<table>
<thead>
<tr>
<th>RE %</th>
<th>SP %</th>
<th>P %</th>
<th>FPR %</th>
<th>FNR %</th>
</tr>
</thead>
<tbody>
<tr>
<td>67</td>
<td>94</td>
<td>59</td>
<td>5.20</td>
<td>36</td>
</tr>
</tbody>
</table>
3.4 Summary

In this chapter, a novel learning-based active contour segmentation technique for accurate object boundary and region extraction was presented. The technique makes use of prior information of the object/objects of interest and uses local image fitting technique in the active contour cost function. By incorporating SOM within curve evolution of the cost function, the contour is guided towards the specific object/objects of interest. The technique showed promising results in segmenting a variety of objects in a different types of imagery.

Our tests of the performance of the proposed method included comparing the results of the proposed SOLAC method with other leading segmentation techniques. In addition, four different categories of imagery were used to assess our method’s performance. In spite of the remarkable progress achieved for accurate object boundary and region extraction using our proposed SOLAC approach, learning-based ACM approaches still consumes relatively more time compared to other ACM approaches that do not use prior information of objects. As a result, the interest in a faster and accurate segmentation inspires us for an optimization approach that will be illustrated in the next chapter.
CHAPTER IV

A SELF-ORGANIZED LATTICE BOLTZMANN ACTIVE CONTOUR (SOLBAC) SEGMENTATION

In classical level-set methods, the cost function is solved using the gradient descent method to characterize curve evolution. This leads to solving a Partial Differential Equation (PDE). Although this scheme showed great promise, the need to simplify and speed up the computations had led us to use the Lattice Boltzmann Method (LBM) as an alternative approach for solving the Level-Set Function (LSF) [13, 14]. One of the merits of using the LBM is that it implicitly manipulates the curvature term in the LSF and therefore speeds up the process of curve evolution convergence.

In this dissertation research, we propose a novel level-set segmentation approach, exploiting the capabilities of SOMs and the LBM. Using the object of interest and background region characteristics as prior knowledge, we design our level-set function which can effectively segment objects in complex background environments. The convergence criteria is optimized using the LBM approach [12] in order to speed up curve evolution of the level-set function.

4.1 The Proposed SOLBAC Approach

We propose a novel Self-Organizing Lattice Boltzmann Active Contour (SOLBAC) segmentation approach for object segmentation. Our segmentation method integrates neural networks within
the Local Image Fitting (LIF) level-set function for boundary and region extraction of objects. In addition, the proposed method leverages the lattice Boltzmann method to reduce computational time. The diagram in figure 4.1 provides an explanation of our proposed method. A functional framework of the proposed segmentation method is shown in figure 4.2.

The proposed segmentation approach is based on four algorithmic steps. The first step incorporates clustering the intensity information of both the object of interest and its background through an unsupervised SOM technique to produce dual SOMs; one SOM network to represent the object of interest, and another SOM network representing the background region. The trained networks are employed into the second stage of our segmentation process to map the intensity levels of an input testing image. The mapped neurons are then utilized to evolve the energy function of LIF level-set protocol in the third stage of our proposed approach. Finally, LBM is used to solve the curve evolution problem and to optimize the convergence of the contour to the most optimum object/objects boundaries.

In classical level-set methods, the problem of minimizing the cost function is handled using calculus of variations [4, 61]. Our method utilizes LBM to solve this problem and this gives us the advantage of reduced computational time. Our modified level-set function is based on a local image fitting energy function [4]. We embed the clustered data determined from the dual SOMs into the level-set function. We use the following energy equation:

$$F^{SOM}(x, y) = (I(x, y) - I^{SOM}(x, y))(s_1 - s_2)\delta_\varepsilon(\phi), \quad (x, y) \in \Omega$$

where $I$ is the input image in domain $\Omega$, $\delta_\varepsilon(\phi)$ is the regularized Dirac delta function given in Equation (2.17), and $I^{SOM}$ is the self-organizing map fitted image which is defined in Equation (3.2) and we recall it here:

$$I^{SOM} = s_1 H_\varepsilon(\phi) + s_2 (1 - H_\varepsilon(\phi))$$

68
Figure 4.1: SOLBAC system diagram.
Figure 4.2: The functional framework of the proposed SOLBAC segmentation system.
where $H_{\varepsilon}(\phi)$ is the regularized Heaviside function given in Equation (2.15), $\phi$ is the level-set function, $s_1$ and $s_2$ are level-set foreground and background parameters, respectively, that are defined in Equations (3.3) and (3.4). We recall them here:

$$s_1 = W_o - \text{mean}(I \in \{(x, y) \in \Omega | \phi(x, y, t) < 0 \} \cap G_k(x, y)), \quad \text{(4.3)}$$

$$s_2 = W_b - \text{mean}(I \in \{(x, y) \in \Omega | \phi(x, y, t) > 0 \} \cap G_k(x, y)), \quad \text{(4.4)}$$

where $W_o$ and $W_b$ are the best matching neurons of the foreground object region and background region respectively, $G_k(x, y)$ is a square Gaussian window with width $k$.

We achieve optimization by introducing our LIF level-set function into the LBM general equation that is defined in Equation (2.30), our modified LBM evolution equation becomes:

$$f_i(x, y + e_i, t + 1) = F^{SOM}(x, y) \left( f_i(x, y, t) + \frac{1}{\tau} \left[ f_i^{eq}(x, y, t) - f_i(x, y, t) \right] + \sigma_c \right) + F^{SOM}(x, y) f_i(x, y, t) \quad \text{(4.5)}$$

Where $\sigma_c$ is the convection coefficient. In this dissertation research, a $D2Q5$ lattice is used (Figure 4.3), in which, $A_i = 1/3$ for the zero central link, and $A_i = 1/6$ for the axial links.

![Figure 4.3: D2Q5 lattice structure.](71)
4.2 Results and Evaluation

The proposed SOLBAC approach is evaluated and compared to similar leading algorithms. Section 4.2.1 provides a series of segmentation results applied to different natural imagery and compared to the state-of-the-art learning-based segmentation methods. The visible imagery are taken from challenging PASCAL 2011 database [1]. The evaluation strategies are identical to the ones described in section 3.3.1. The performance of the proposed SOLBAC approach is also tested on other imagery types viz. hyperspectral imagery, medical imagery and thermal imagery. The hyperspectral imagery are natural images taken using Resonon Pika II hyperspectral camera, which provides 240 spectral channels that range from 400-900nm with 2.1nm spectral resolution. The medical imagery are taken from datasets composed of X-ray imagery of bones, lungs, livers, and brains. The datasets are selected from the publicly available online servers hosted by The Cancer Image Archives (TCIA)[51] and by Life In The Fast Lane (LITFL)[52] Radiology Image Database. The thermal imagery are real-life images that are taken by a FLIR T650sc thermal camera, which provides high sensitivity of temperature measurements.

4.2.1 Evaluation of SOLBAC on Visible Imagery

This section shows the performance of applying the proposed SOLBAC approach on visible imagery. In addition, we compare our method with other state-of-the-art SOM-based ACM approaches such as CSOM-CV method [6], edge-based SOM-ACM method (EB-SOM-ACM) [7], TASOM-ACM method [8], and SOLAC method. When performing segmentation of objects, the enclosed boundary around the regions of interest is transformed into a binary image mask which can be directly compared to the ground truth of the segmented regions.

In Figures 4.4 - 4.7, we depict a series of images. The first column shows input image, second column illustrates the result using EB-SOM-ACM method, third column exhibits the result using
CSOM-CV method, forth column demonstrates the result obtained using TASOM-ACM approach, fifth column shows the result using our SOLAC method, and finally the sixth column demonstrates our SOLBAC method result.

In Figure 4.4, we illustrate the segmentation of an airplane as the object of interest. The segmentation result obtained using EB-SOM-ACM shows that the wings of the airplane are totally missed and portions of the landing gears are missed. Specifically, the straits attaching the wheels to the fuselage are overlooked. CSOM-CV and TASOM-ACM results show that major portions of the wing are missing. The result achieved using SOLAC approach miss part of the wings. The result obtained using our SOLBAC method shows the entire region of the airplane without missing any part.

We illustrate the results obtained in segmenting people in Figure 4.5. Results obtained using EB-SOM-ACM and CSOM-CV miss parts of the heads of the people. TASOM-ACM handles those missed parts better, but introduces false positives. Our SOLAC and SOLBAC methods segment those portions of the head missed by other methods and introduce less false positives.

We demonstrate the results obtained in segmenting a dog in Figure 4.6. The segmentation result obtained using EB-SOM-ACM shows that parts of the face and neck of the dog are missing. The result obtained using CSOM-CV shows satisfying results while missing some pixels from the nose of the dog. TASOM-ACM result shows missing parts of the neck of the dog with some false positives. SOLAC result shows missing parts of the nose as well. The result obtained using our SOLBAC method shows all parts of the dog including the very detailed parts of the nose with some false positives.

Moreover, we illustrate the results obtained in segmenting sheep in Figure 4.7. The result obtained using EB-SOM-ACM shows that this method could not provide satisfactory results for this image. The result obtained using CSOM-CV shows better results compared to EB-SOM-ACM
Figure 4.4: Results of an air plane segmentation; (a) input image; (b) segmentation result of EB-SOM-ACM method; (c) segmentation result of CSOM-CV method; (d) segmentation result of TASOM-ACM method; (e) segmentation result of SOLAC method; (f) segmentation result of SOLBAC approaches.

method but missed parts of the ears. The result obtained using TASOM-ACM also shows unsatisfying results for this image. SOLAC result is quit simillar to the result obtained using CSOM-CV method with more false positives. The result obtained using our SOLBAC method shows much satisfying segmentation results, even though there are some false positives.

Table 4.1 shows the execution time in seconds by the different segmentation methods for extracting the regions in images shown in Figures 4.4 - 4.7. Compared to all three SOM-based ACM methods, Table 4.1 shows that SOLBAC approach is the most computationally efficient.

In addition to visual inspection, we substantiate and validate the proposed technique on a set of 100 randomly selected images from the PASCAL-2011 dataset. We evaluate our algorithm by
Figure 4.5: Results of people segmentation; (a) input image; (b) segmentation result of EB-SOM-ACM method; (c) segmentation result of CSOM-CV method; (d) segmentation result of TASOM-ACM method; (e) segmentation result of SOLAC method; (f) segmentation result of SOLBAC approaches.

Table 4.1: Execution time for various segmentation methods applied in images of Figures 4.4 - 4.7.

<table>
<thead>
<tr>
<th>Image</th>
<th>Size(pixels)</th>
<th>EB-SOM-ACM</th>
<th>CSOM-CV</th>
<th>TASOM-ACM</th>
<th>SOLAC</th>
<th>SOLBAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.4</td>
<td>444 × 500</td>
<td>3.67</td>
<td>1.87</td>
<td>6.21</td>
<td>10.0</td>
<td>1.48</td>
</tr>
<tr>
<td>4.5</td>
<td>500 × 389</td>
<td>4.68</td>
<td>3.35</td>
<td>5.36</td>
<td>8.72</td>
<td>3.27</td>
</tr>
<tr>
<td>4.6</td>
<td>474 × 500</td>
<td>5.70</td>
<td>2.67</td>
<td>6.63</td>
<td>8.12</td>
<td>2.02</td>
</tr>
<tr>
<td>4.7</td>
<td>375 × 500</td>
<td>4.24</td>
<td>3.54</td>
<td>5.34</td>
<td>9.51</td>
<td>3.19</td>
</tr>
</tbody>
</table>
computing metrics for our method along with the EB-SOM-ACM, CSOM-CV, and TASOM-ACM approaches. We had computed the average values of recall (RE), specificity (Sp), precision (P), false positive Rate (FPR), false negative rate (FNR), and Haussdrof distance [62]. Higher values of the first three metrics and lower values of the last three metrics indicate a better segmentation result. These metrics are shown in Table 4.2 and Figure 4.8, from which we can observe that the proposed SOLBAC approach competes with the state-of-the-art methods.

Thus, from Figures 4.4- 4.8 and 4.12 and Tables 4.1 - 4.2, we can observe that the proposed SOLBAC algorithm produces promising segmentation results even though there are false positives in some cases due to the complexity of the background regions.
Figure 4.7: Results of sheep segmentation; (a) input image; (b) segmentation result of EB-SOM-ACM method; (c) segmentation result of CSOM-CV method; (d) segmentation result of TASOM-ACM method; (e) segmentation result of SOLAC method; (f) segmentation result of SOLBAC approaches.

Table 4.2: Average statistics for a set of 100 randomly selected natural images from PASCAL 2011 dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>RE %</th>
<th>SP %</th>
<th>P %</th>
<th>FPR %</th>
<th>FNR %</th>
<th>Haussdorf distance %</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSOM-CV [6]</td>
<td><strong>73</strong></td>
<td><strong>54</strong></td>
<td><strong>45</strong></td>
<td>43</td>
<td><strong>26</strong></td>
<td>14.30</td>
</tr>
<tr>
<td>SOLAC</td>
<td>69</td>
<td>55</td>
<td>42</td>
<td>44</td>
<td>30</td>
<td>14.33</td>
</tr>
<tr>
<td>SOLBAC</td>
<td>70</td>
<td>54</td>
<td>42</td>
<td>45</td>
<td>29</td>
<td><strong>14.23</strong></td>
</tr>
</tbody>
</table>
4.2.2 Application of SOLBAC to Different Types of Imagery

Moreover, the SOLBAC approach is tested on hyperspectral imagery, medical imagery, and thermal imagery too. Figure 4.9 shows hyperspectral image segmentation using both of the proposed segmentation methods the SOLAC and SOLBAC approaches.

Figure 4.10 shows medical image segmentation using both SOLAC and SOLBAC approaches. Furthermore, in Figure 4.11 we illustrate thermal image segmentation using both SOLAC and SOLBAC methods.

4.2.3 Computation Speed

All experiments were conducted on a personal computer with an Intel Core I5, 2.53GHz processor, 64-bit operating system having 4.00 GB of RAM and running MATLAB R2014a.
Figure 4.9: Segmentation results of a hyperspectral image; (a) Input image; (b) segmentation result using SOLAC; (c) segmentation result using SOLBAC.

Time consumption is measured for all methods. SOLBAC method performed segmentation with reduced processing time compared to process time of the CSOM-CV, EB-SOM-ACM, TASOM-ACM, and SOLAC methods. Figure 4.12 shows the average time consumption comparison for
Figure 4.10: Segmentation results of a medical image; (a) Input image; (b) segmentation result using SOLAC; (c) segmentation result using SOLBAC.

Figure 4.11: Segmentation results of a thermal image; (a) Input image; (b) segmentation result using SOLAC; (c) segmentation result using SOLBAC.

It can be observed that the proposed SOLBAC approach performs the segmentation process in a lower time compared to the state-of-the art learning-based segmentation methods. Because the

all four methods when applied to 100 randomly selected images from PASCAL-2011 database. The SOLBAC approach consumes less computation time compared to other learning-based ACM approaches.
Figure 4.12: Average time consumption comparison for a set of 100 images using different learning-based segmentation techniques.

Table 4.3: Average time consumption comparison for a set of 100 images using different learning based segmentation techniques.

<table>
<thead>
<tr>
<th>Method</th>
<th>Time (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOLBAC</td>
<td>3.06</td>
</tr>
<tr>
<td>EB-SOM-ACM [7]</td>
<td>30.76</td>
</tr>
<tr>
<td>TASOM-ACM [8]</td>
<td>51.61</td>
</tr>
<tr>
<td>SOLAC</td>
<td>88.53</td>
</tr>
</tbody>
</table>

proposed SOLBAC approach leverage LMB in the convergence process of the evolved contour, it can minimize the problem of time consumption. More specifically, the LBM approach is capable of solving the complex parts of the evolving curve implicitly because of the fact that this method
discretized space and time as described earlier in chapter II, and this helps in tremendous reduction of processing time.

4.3 Summary

In this chapter a new fast learning-based segmentation approach (SOLBAC) was presented. This method starts with seed patches from object/objects of interest and the background to learn a dual SOMs. The learned SOMs are utilized to retrieve the prior information and integrate it in the ACM cost function. The convergence of the ACM is achieved using LBM. The proposed framework showed effective segmentation in terms of accuracy and significant reduction of time consumption.
CHAPTER V

CONCLUSION AND FUTURE WORK

In this dissertation we presented a novel learning-based segmentation approach exploiting SOM and ACM for object region extraction, preserving the fine details that are critical for object identification and scene understanding applications. The proposed method is established using learning-based level-set active contour models that utilize prior information from the object of interest and its background to guide the active contours towards desired object/objects of interest. The learning-based active contours utilize SOM neural networks to retrieve information about object/objects of interest. Our proposed SOLAC framework contains the representational power necessary for accurate object boundary and region extraction in complex background environments. Unlike traditional segmentation methods that do not utilize prior knowledge of objects, the presented technique plays an important role in semantic segmentation for object identification and recognition. Additionally, we introduced the SOLBAC technique, which is an optimized version of the SOLAC method, to speed up the segmentation process by utilizing LBM. The proposed SOLBAC technique was shown to effectively segment objects with lower computation time compared to other state-of-the-art learning based active contour techniques. Both of our presented methods remarkably reduced the effect of over segmentation problem caused from sticking at local minima in active contours.

The two presented frameworks were tested qualitatively and quantitatively in order to evaluate their performances. The qualitative evaluation was done by testing the presented SOLAC and
SOLBAC methods on a variety of different types of imagery ranging between visible imagery, hyperspectral imagery, medical imagery, and thermal imagery. Moreover, the performances of the proposed SOLAC and SOLBAC schemes were compared to the state-of-the-art learning-based segmentation methods and shown to perform competitively. The quantitative assessment was done by calculating metrics that measure the accuracy of the segmentation results along with processing time consumption. Those measurements were also compared with metrics obtained from state-of-the-art learning-based segmentation methods. These results proved that the proposed approaches, particularly the SOLAC and SOLBAC methods, are able to cope with segmentation objects in complex environments. Although SOLAC approach consumes relatively long processing time, by leveraging LBM to reduce time consumption, the SOLBAC method performs accurately and with much lower computational time. To sum up, our SOLAC and SOLBAC approaches proved good performance and accurate object segmentation. The goal is to generate learning-based ACM techniques for real-time applications. So far, this task remains unaccomplished, although preliminary efforts had started to appear.

Further improvements and future directions of this work include:

- Explore the use of other neural networks to retrieve the information of the object/objects of interest. Time adaptive self-organizing map (TASOM) is one possible alternative in which adaptive learning rate and adaptive neighborhood function for each neuron of the neural network that change with time are being used. The adaptive parameters are automatically adjusted for each neuron independently [63]. TASOM is being used for adaptive object segmentation.

- Explore the use of a 3D lattice of LBM method along with 3D level-set functions to further enhance the segmentation performance.
• Explore the use of parallelization ability of the LBM method to speed up processing time. The current trend to increase computational capacity is based on clustering CPUs. Hence, in order to decrease the computational time required for computation, parallelization would be one of the most optimum solutions. The main advantage of LBM is that the information from neighboring nodes is only needed for evolving variables. Therefore, only local exchange of lattice nodes is needed and this property makes the LBM have almost no communication overheads between CPUs and therefore ideal for parallel computing.

• Explore the use of GPU-accelerated computing, in which graphical processing unit (GPU) together with a CPU to accelerate segmentation process.

• Consider other types of descriptors to be used as prior information about the object/objects of interest like texture, shape, or motion.

• Test the framework with additional challenging databases.
PUBLICATIONS

JOURNALS


CONFERENCE PROCEEDINGS


ABSTRACTS


BIBLIOGRAPHY


92


