13

Unsupervised Medical Image Segmentation using Stochastic Optimization Methods

Ivan Cruz-Aceves[1], Juan Gabriel Avina-Cervantes[1], Juan Manuel Lopez-Hernandez[1], Ma. de Guadalupe Garcia-Hernandez[1], Miguel Torres-Cisneros[1], Manuel Ornelas-Rodriguez[2].

[1]University of Guanajuato, Electronics Department (DICIS). Salamanca, Guanajuato, Mexico
[2]Leon Institute of Technology, Graduate Department (ITL). Leon, Guanajuato, Mexico

Outline:

Introduction 335
Image segmentation and optimization methods 336
Active Contour Models (ACM) 336
The ACM Algorithm in General 337
Stochastic Optimization Methods 337
Particle Swarm Optimization (PSO) 337
Differential Evolution 338
Estimation of Distribution Algorithms (EDAs) 339
Application of PSO, DE, and UMDA Algorithms 340
Proposed Medical Image Segmentation Methods 343
Segmentation through Constrained Polar Sections (CPS) 343
Pseudocode and Parameter Selection 345
Segmentation by using Maximum Euclidean Distance (MED) 347
Computational Experiments 348
Application to Human Heart on Computed Tomography Images 348
Application to White Blood Cells on Microscope Images 350
Concluding Remarks 353
Acknowledgements 353
References 354

Introduction

In medical image analysis, the automatic segmentation is an important and challenging problem for the diagnosis and monitoring of different diseases. The objective of image segmentation is to separate objects of interest from a given image based on different attributes such as shape, colour, intensity or texture. In recent years, several methods have been proposed for this purpose such as adaptive local multi-atlas in human heart [1], suppressed fuzzy c-means in brain magnetic resonance images [2], improved watershed transform in mammograms [3], graph cut in multiple human organs [4, 5], rule optimization with region growing in pelvic injuries [6] and active contour models (ACM) in human prostate [7], lungs from magnetic resonance images of the torso [8] and intravascular ultrasound images [9], to name a few.

The Active Contour Model was introduced by Kass et al. [10] and it is an energy-minimizing spline curve composed of control points also known as snaxels. This spline evolves through evaluation of internal and external forces being attracted towards features as edges of a target object. The classical implementation of ACM presents the shortcomings of sensitivity to initial positioning of the control points, which must be close to the target object and the propensity to be trapped into local minima. To solve these drawbacks different improvements have been proposed to adapt diverse methods working together with ACM such as Finite-element [11], graph cut [12], statistical methods [13, 14] and population-based methods including Genetic algorithms [15, 16], Differential Evolution (DE) [17], Estimation of Distribution Algorithms (EDAs) [18] and Particle Swarm Optimization (PSO) [19, 20, 21]. The use of population-based methods working together with the classical ACM becomes more stable and efficient in the local minima, which is highly suitable for medical image segmentation problems.

Population-based methods are an effective way to solve different optimization problems. Three of the most popular methods are the PSO, DE and EDAs because of their robustness in local minima and efficiency solving global optimization problems with nondifferentiable functions. Since these techniques are not computationally demanding and highly efficient, they have been used in many real-world applications including, tumour classification [22], cancer chemotherapy optimization [23] and parameter estimation in human immunodeficiency virus (HIV) [24].

This chapter introduces novel unsupervised image segmentation methods based on three different stochastic optimization techniques. These unsupervised methods use the theory of Active Contour Model working together with Differential Evolution, Estimation of Distribution Algorithms and Particle Swarm Optimization to perform the segmentation task. These methods are explored and applied in the segmentation of white blood cells (leukocytes) from microscope images and for segmentation of the human heart from Computed Tomography images, where the human heart results are evaluated using the Jaccard and Dice indexes with respect to regions outlined by experts.

Image segmentation and optimization methods

This section introduces the basis of the Active Contour Model for image segmentation, and the fundamentals of the stochastic optimization methods of Differential Evolution, Estimation of Distribution Algorithms and Particle Swarm Optimization.

Active Contour Models (ACM)

The classical Active Contour Model, also called snake, is represented by a parametric curve which can move within the spatial domain of an image where it was assigned [10]. This parametric curve is defined by *p(s, t) = (x(s, t), y(s, t)), s* ∈ *[0, 1]*, where *t* is the time parameter whereby the curve evolves to minimize the total energy function given by Equation (1).

$$E_{snake} = \int_0^1 \left[E_{int}\big(p(s,t)\big) + E_{ext}(p(s,t))\right] ds \tag{1}$$

This energy function has two components, the internal energy E_{int} and the external energy E_{ext}. Internal energy is presented in Equation (2), which is used to maintain the search within the spatial image domain, and to control the shape modification of the parametric curve using the first and second derivatives of ***p(s)***, the curve tension parameter ***α(s)*** and the rigidity parameter ***β(s)***.

$$E_{int}\big(p(s,t)\big) = \frac{1}{2}\left[\alpha(s)\left|\frac{\partial p(s)}{\partial s}\right|^2 + \beta(s)\left|\frac{\partial^2 p(s)}{\partial s^2}\right|^2\right] \tag{2}$$

The external energy defined by Equation (3) is given by the particular features of the image, where $\boldsymbol{\nabla I}(p(s))$ is the surface gradient computed at *p(s)* and γ is a weight parameter. The optimal solution is acquired by solving the Euler equation (Equation (4)), when both external and internal energies become stable.

$$E_{ext}\big(p(s)\big) = -\gamma|\nabla I(p(s))|^2 \tag{3}$$

$$\nabla E_{ext} - \alpha\frac{\partial^2 p(s)}{\partial s^2} + \beta\frac{\partial^4 p(s)}{\partial s^4} = 0 \tag{4}$$

In the computational implementation of the traditional ACM, the parametric curve is composed by a set of n control points $\{p_i \mid i = 1,2,\dots,n\}$, also called snaxels. The internal and external energies are approximated by Equation (5) and Equation (6) respectively. In both energies $q_{i,j}$ represents the current control point p_i in the j index within its searching window. Accordingly, the local energy function given by Equation (7) is iteratively evaluated to minimize the k_i index by using Equation (8), where W_i is the predefined searching window for the control point p_i [21].

$$E_{int} = \frac{1}{2}\left[\alpha(s)\big|q_{i,j} - p_{i-1}\big|_2^2 + \beta(s)\big|p_{i-1} - 2q_{i,j} + p_{i+1}\big|_2^2\right] \tag{5}$$

$$E_{ext} = -\gamma|\nabla I(q_{i,j})|_2^2 \quad (6)$$

$$E_{i,j} = E_{int} + E_{ext} \quad (7)$$

$$E_{snake} = \sum_{i=i}^{n} E_{i,k_i}, k_i = arg\ min_j(E_{i,j}), j \in w_i \quad (8)$$

This classical ACM implementation presents two main weaknesses. Firstly, sensitivity to the initial position of the control points which must be close to target object otherwise failure of convergence will occur; secondly, these control points are prone to be trapped into local minima problem deflecting the parametric curve of the optimum edge. A suitable alternative to overcome the aforementioned drawbacks is to use robust stochastic optimization methods. In our experiments, three different population-based methods were used, which are described in the following Section 2.2.

The ACM Algorithm in General

According to the previous description in Section 2.1, the classical ACM algorithm can be implemented by using the procedure in Box 1.

1. Initialize the control points $\{p_i \mid i = 1,2,\dots,n\}$
2. For each control point p_i:
3. Calculate $E_{i,j}$ in searching window W_i using Equation (7)
4. Find the best index k_i according to Equation (8)
5. Set $P_i = q_{i,k_i}$
6. Compute E_{snake}
7. If E_{snake} becomes stable, then stop, otherwise repeat steps (2) to (7)

BOX 1
Algorithm of Active Contour Model

Stochastic Optimization Methods

Stochastic Optimization methods are used to solve numerical optimization problems based on different strategies. In this work, we focus on Differential Evolution, Estimation of Distribution Algorithms and Particle Swarm Optimization.

Particle Swarm Optimization (PSO)

Particle Swarm Optimization is a computational intelligence technique proposed by Eberhart et al. [25] and improved by [26] to solve numerical optimization problems. Similar to evolutionary techniques, PSO uses a set also called swarm of potential solutions referred to as particles to perform the optimization task. Each particle represents a solution in an *N*-dimensional space $\boldsymbol{X}_i = \{x_{i1}, x_{i2}, \dots, x_{iN}\}$, which moves through hyperspace to a new position according to the following velocity equation:

$$v_i(t+1) = \varphi v_i(t) + kr_1\big(p_{best} - x_i(t)\big) + kr_2(p_{gbest} - x_i(t)) \quad (9)$$

where $v_i(t)$ is the current velocity of the particle x_i in the time step (t), φ is the inertia weight, k represents the learning factor, $r_1, r_2 \sim U\ (0,1)$ where U is a uniform distribution, p_{best} is the current best solution found by x_i, and p_{gbest} is the best solution found by the best particle of the whole swarm. Subsequently, assuming that the new velocity of the particle has been updated, Equation (10) is used to calculate its new position within the search space.

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (10)$$

According to the aforementioned description, the PSO algorithm can be implemented by using the following procedure.

1. Initialize iterations *G*, and swarm size N_p
2. Initialize each particle X_i by generating random positions and velocities.
3. For each particle $X_{i,g}$, where $g = \{1, \dots G\}$:
4. Evaluate $X_{i,g}$ in fitness function and update its P_{best}, if the new fitness is better.
5. Find the best particle in the swarm and update P_{gbest}, if the fitness value found is better.
6. Stop if the convergence criterion is satisfied (e.g., stability or number of iterations).
7. Update velocity of all particles using Equation (9).
8. Update position of all particles using Equation (10), then repeat steps 3-8.

BOX 2
Algorithm of Particle Swarm Optimization

Differential Evolution

Differential evolution (DE) is a stochastic real-parameter heuristic from the family of evolutionary algorithms proposed by Storn et al. [27, 28] for numerical global optimization problems. DE starts with a number N_p of randomly initialized potential solutions, also known as individuals $X = \{x_1, x_2, \dots, x_{N_p}\}$. These individuals are iteratively improved through different variation operators and the solution is chosen to be the individual with the best fitness according to an objective function.

The main idea of DE method consists of three evolutionary operators: mutation, crossover and selection on the floating-point encoding.

Mutation: Creates a mutant vector $V_{i,g+1}$ at each generation g based on the distribution of the current population $\{X_{i,g} | i = 1,2, \dots, N_p\}$ using the following mutation strategy:

$$V_{i,g+1} = X_{r1,g} + F\big(X_{r2,g} - X_{r3,g}\big), \quad r1 \neq r2 \neq r3 \neq i \quad (11)$$

where F is the differentiation factor also called scaling or mutation factor parameter, and $r1$, $r2$ and $r3$ are the indexes of three individuals mutually different and uniformly randomly selected from the set $\{1, \dots, N_p\}$.

Crossover: This operator is used to create the trial vector $U_{i,g+1}$ via the following:

$$U_{i,g+1} = \begin{cases} V_{i,g+1}, & if\ r \leq CR \\ X_{i,g}, & if\ r > CR \end{cases} \quad (12)$$

where a uniform random value r on the interval $(0,1)$ is generated to be compared with the crossover rate parameter CR. If r is bigger than CR, the current information of individual $X_{i,g}$ is preserved, otherwise the values from the mutant vector $V_{i,g+1}$ are copied to the trial vector $U_{i,g+1}$.

Selection: This operator is applied using Equation (13) to optimization process. It selects according to a fitness function, the better one between the current individual $X_{i,g}$ and the trial vector $U_{i,g+1}$ to be used to replace the current individual in the next generation.

$$X_{i,g+1} = \begin{cases} U_{i,g+1}, & if\ f(U_{i,g+1}) < f(X_{i,g}) \\ X_{i,g}, & otherwise \end{cases} \quad (13)$$

According to the previous description of the traditional DE algorithm, it is described by using the following procedure.

1. Initialize generations *G*, population size N_p, differentiation F and crossover rate CR
2. Initialize each individual X_i by generating random candidate solutions
3. For each individual $X_{i,g}$, where $g = \{1, \dots G\}$:
4. Compute $V_{i,g+1}$by using the mutation step in Equation (11)
5. Assign $U_{i,g+1}$ according to the crossover operator via Equation (12)
6. Update $X_{i,g+1}$, if $U_{i,g+1}$ is better than $X_{i,g}$ using Equation (13)
7. If stopping criterion is satisfied (e.g., stability or number of generations), then stop

BOX 3
Algorithm of Differential Evolution

Estimation of Distribution Algorithms (EDAs)

Estimation of Distribution Algorithms are population-based methods that incorporate statistical information to solve optimization problems [29, 30, 31]. EDAs are from the family of evolutionary algorithms since they use a population of individuals, selection operators and binary encoding replacing the crossover and mutation operators by probabilistic models based on the global statistical information inferred from the current solutions. One of EDAs that works perfectly for linear problems or problems with not many significant dependencies is the Univariate Marginal Distribution Algorithm (UMDA)[32, 33]. This algorithm works on binary strings to infer statistical dependencies between the variables using a probability vector $\boldsymbol{p} = (p_1, p_2, \dots, p_n)^T$ to construct the probabilistic model, where p_i represents the probability of obtaining a 1 in position i. UMDA approximates the actual probability distribution of the individuals in $\mathbb{P}_t$ using the product of the

univariate frequencies computed from the selected population and assuming that all variables are independent [34]. The first step of UMDA consists of the selection probability s, which is computed using Equation (14) and it is performed after the chromosomes in the search space **Ω** have been sorted according to fitness.

$$\mathbb{P}^s(x) = \frac{\mathbb{P}(x)f(x)}{\sum_{\bar{x}\in\Omega}\mathbb{P}(\bar{x})f(\bar{x})} \tag{14}$$

In the second step a joint probability $\mathbb{P}$ is calculated through the following:

$$\mathbb{P}(x) = \prod_{i=1}^{n}\mathbb{P}(X_i = x_i) \tag{15}$$

where $x = (x_1, x_2, \ldots, x_n)^T$ is the binary value of the ith bit in the chromosome, and X_i is the ith component of the random vector X. Finally, the third step of UMDA is used to generate a new population of individuals from the probabilistic model, to be evaluated according to a fitness function in the next generation. These three steps of UMDA are iteratively performed until the termination criteria are satisfied, and the solution is chosen to be the individual with the best fitness in the entire population.

Based on the previous description, the UMDA algorithm can be implemented through the following procedure:

1. Initialize generations *T*, population size N_p.
2. Initialize each individual X_i by generating random candidate solutions
3. For each individual $X_{i,t}$, where t= $\{1, \ldots T\}$:
4. Select a subpopulation *S* of individuals according to a selection method.
5. Compute the univariate marginal probabilities $p_i^S(x_i, t)$ of *S*.
6. Generate n new individuals according to $p(x, t+1) = \prod_{i=1}^{n} p_i^S(x_i, t)$.
7. Stop if convergence criterion is satisfied, otherwise, repeat steps (3)-(7).

BOX 4
Algorithm of the Univariate Marginal Distribution Method

Application of PSO, DE, and UMDA Algorithms

Since the aforementioned algorithms are optimization techniques, they can be applied to minimize the following function:

$$\min f(x) = x^2, \ [-64,64] \tag{16}$$

Figure 13.1(a) presents the graph of the function (16), in which the optimization process will be explained. The first step in the optimization methods is the initialization of their particular parameters followed by generating random potential solutions. These solutions referred as

individuals or particles must be initialized within the search space defined by the function. In our example, the search space is in the range [-64, 64], and 8 individuals have been randomly initialized, as shown in Figure 13.1(b) and Table 13.1. The binary representation used for EDAs has 7-bits, since the range of the function ([-64, 64]) can be rearranged to establish a new range in [0, 128].

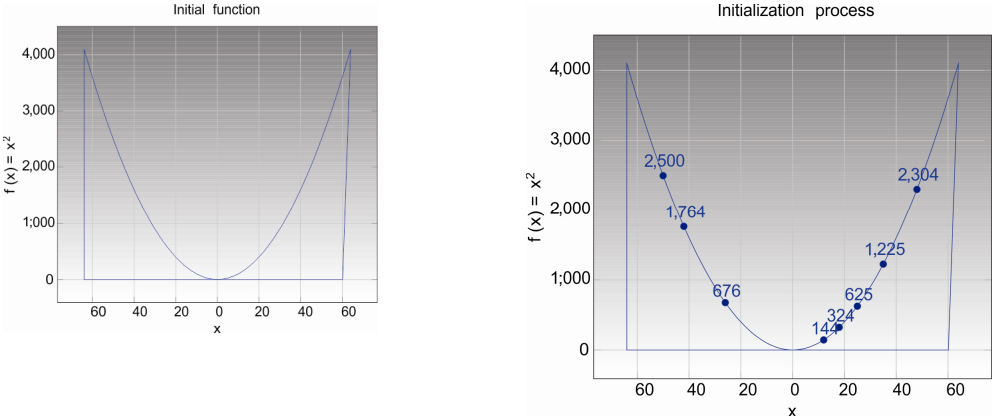


FIGURE 13.1
(a) Plot of function $f(x) = x^2$ and (b) Initialization of potential solutions

TABLE 13.1
Initialization of potential solutions according to Figure 13.1(b)

individuals	Real encoding (PSO/DE)	Binary encoding (EDAs)	Fitness
1	25	1011001	625
2	48	1110000	2304
3	18	1010010	324
4	-50	0001110	2500
5	-42	0010110	1764
6	12	1001100	144
7	35	1100011	1225
8	-26	0100110	676

While EDAs works on the binary encoding, PSO and DE use real encoding to perform their search strategy. The strategy search of each optimization method is applied through iterations over the whole set of potential solutions. For instance, PSO uses the velocity and position equations, DE works with mutation, crossover and selection operator and UMDA uses selection and univariate marginal probabilities. When the optimization process is finished, in general taking into account the number of iterations, the optimal solution (Figure 13.2) is chosen to be the individual or particle with the best fitness in the whole set of potential solutions. These methods are easy to extend for working with n-dimensional problems just by modifying their codification, since the search strategies have been proven to be robust and efficient in different applications [23, 24, 35]. In

order to adapt the codification presented in Table 13.1 and extending the function (16) to two-dimensions (function (17)), in the following Table 13.2 and Figure 13.3 the adaptation is illustrated.

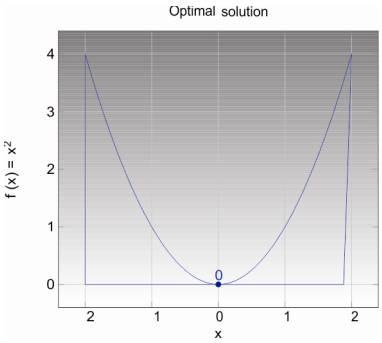


FIGURE 13.2
Optimal solution of function (16)

$$\min f(x,y) = x^2 + y^2, \ [-64,64] \quad (17)$$

TABLE 13.2
Adaptation of potential solutions to minimize the function 17 shown in Figure 13.3

	Real encoding		Binary encoding		
Individual	**X-axis**	**Y-axis**	**X-axis**	**Y-axis**	**Fitness**
1	25	3	1011001	1000011	634
2	48	21	1110000	1010101	2745
3	18	-10	1010010	0110110	424
4	-50	-32	0001110	0100000	3524
5	-42	46	0010110	1101110	3880
6	12	-50	1001100	0001110	2644
7	35	-20	1100011	0101100	1625
8	-26	-2	0100110	0111110	680

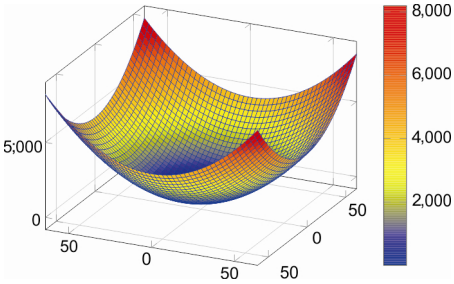


FIGURE 13.3
Surface of spherical function $f(x, y) = x^2 + y^2$

In the following Section 3, two medical image segmentation frameworks based on different strategies working together with the optimization methods described above, are explained in detail.

Proposed Medical Image Segmentation Methods

The first method uses constrained polar sections to perform the segmentation process. These constrained sections make it possible to adapt different optimization methods preserving their original search strategies as it is described in Section 3.1. The second method uses the maximum Euclidean distance between potential solutions to perform the segmentation task. The search is not constrained via polar sections; instead, Euclidean distance is used as it is illustrated in Section 3.3.

Segmentation through Constrained Polar Sections (CPS)

In this method, different optimization techniques can be adopted without any modification for guiding the convergence of multiple active contours on a polar coordinate system to perform the segmentation task [36, 37]. Due to the local minima disadvantage of the traditional ACM discussed above, this method uses population-based techniques to solve it. Since the methodology makes it possible to apply the traditional implementation of the optimization techniques, the advantages of robustness, low computational time and efficiency are inherently acquired. The procedure of the segmentation method is illustrated below in Figure 13.4, which consists of the Preprocessing, Initialization and Segmentation steps.

In the preprocessing stage, we remove noise from the image by using a 2-D median filter (3x3 window size), followed by the Canny edge detector to separate the regions of interest from the background. The last step in this stage is to produce the Euclidean Distance Map, where low potential values (ideally zero) are assigned to pixels located close to the target object and high potential values to pixels far from the object [11].

In the initialization stage, a polar coordinate system on the Distance Map is generated via an interactively determined seed point composed by the $\boldsymbol{x}$ and $\boldsymbol{y}$ coordinates. This coordinate system divides the target object through $\theta = {}^{2\pi}/_{g}$, where g denotes the degrees of each constrained

polar section S, in which one edge sectional solution must exist. Subsequently, the object of interest has to be confined by the spatial domain of the n initial contours, which can be generated in a circular or elliptical shape according to the pattern of the object. After the n contours are produced, n equidistant control points (snaxels) are generated and assigned as potential solutions (individuals or particles) to conform one population O_i for each polar section S_i. The third stage represents the segmentation process, where for section S_i, the optimization method is applied individually to minimize the corresponding edge sectional solution. Once the optimization process is finished, the final segmented object is acquired by connecting the best solution of each polar section to each other.

The proposed method presents the following advantages on the initialization process. Firstly, the initial contours can be easily defined in different shapes according to the pattern of the target object. Secondly, the number of control points (snaxels) per contour, can be modified according to the number of polar sections in which the target object is divided. The last feature is the seed point, which is used to generate all the snaxels automatically on the constrained spatial domain of the object of interest allowing extend the present method to work with sequential images just reproducing the origin point through the set of images.

Since the segmentation results directly depend from the appropriate parameters selection of the presented method, they are individually described in the following Section.

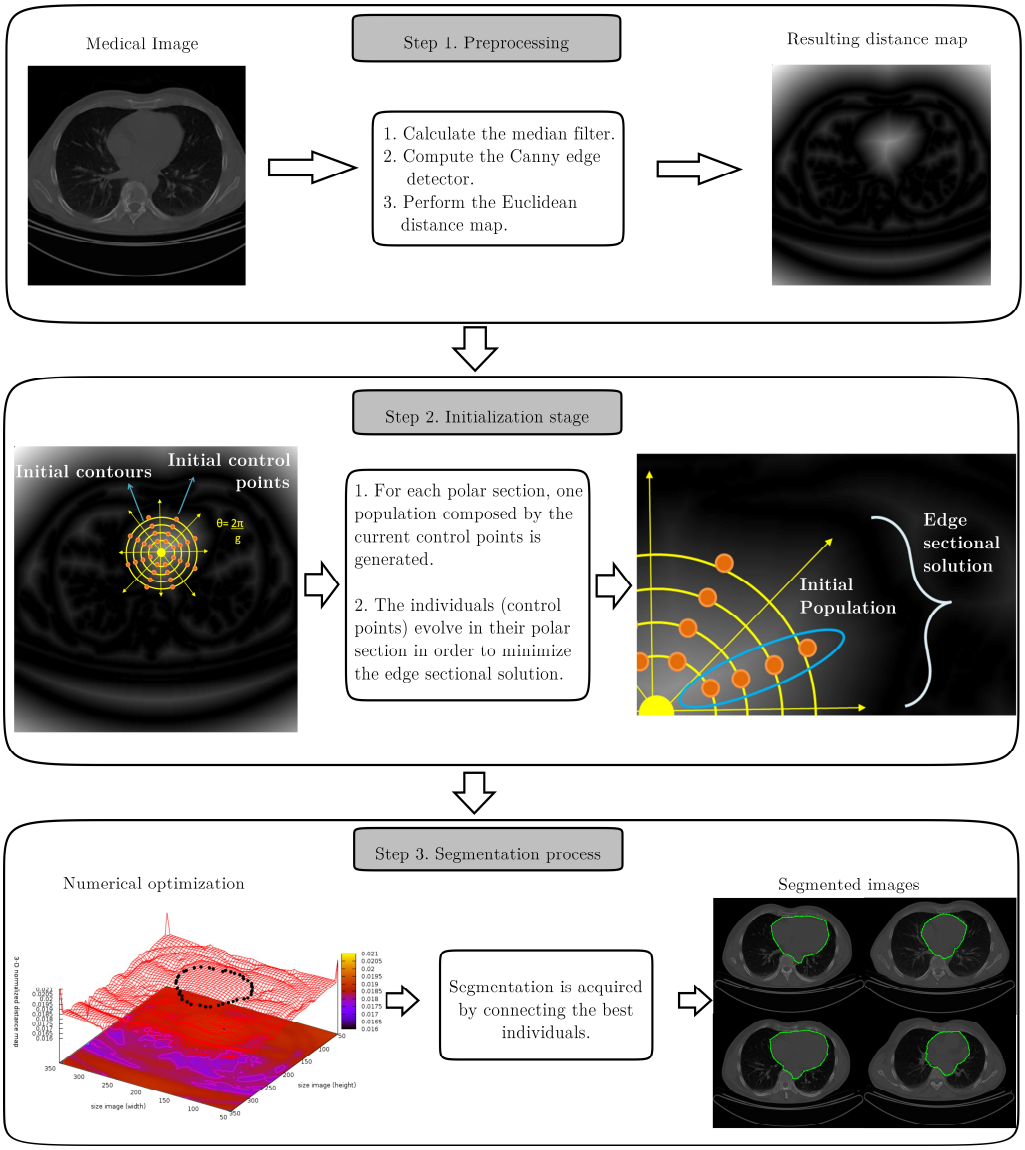


FIGURE 13.4
Workflow of the Constrained Polar Sections method, consisting on the steps of preprocessing of medical image, initialization of evolutionary methods, and segmentation process through numerical optimization. Reproduced with permission from I. Cruz-Aceves, J. G. Avina-Cervantes, J. M. Lopez-Hernandez, et al., "Multiple Active Contours Guided by Differential Evolution for Medical Image Segmentation", Computational and Mathematical Methods in Medicine, Volume 2013, Article ID 190304, 14 pages. ©Hindawi Publishing Corporation

Pseudocode and Parameter Selection

In this section, the procedure as pseudocode and the parameters description of the image segmentation method based on constrained polar sections are described as follows:

1. Initialize coordinates (x, y) from the origin point, degrees g and number of snakes.
2. Initialize parameters of optimization method such as number of iterations and population size.
3. Generate one population or swarm O_i for each polar section S_i.
4. For each population or swarm O_i:
5. Apply restriction of the search space to ignore improper solutions.
6. Perform the particular search strategy of the optimization method.
7. Stop if convergence criterion is satisfied, otherwise, repeat steps (4)-(7).

BOX 5
Algorithm of the method based on Constrained Polar Sections

The parameters play an essential role in the success of any optimization algorithm. In our approach, the parameters have been experimentally tuned taking into account the number of different potential solutions generated through the iterations, and also by considering the number of improper solutions to perform local exploitation instead exploration.

Preprocessing step: These parameters have been experimentally tuned to preserve the real edges in the image, since these can affect the segmentation result. Particularly, the parameters in the Canny edge detector were used as $\sigma = 1.3, T_l = 10.0$ and $T_h = 30.0$.

Degrees: This parameter g represents the degrees of each constrained polar section S. In our experiments g was statistically tuned in the range [13, 16] degrees, since the relation between computational time and segmentation results is suitable.

Number of snakes: It has to be defined assuming that the target object is confined within the region of the initial snakes. According to the human heart segmentation experiments, the number of snakes was set experimentally between 12 and 15 contours.

Number of snaxels: It depends of the relation $2\pi/g$. The number of snaxels of each snake determines the number of polar sections in which the target object is divided.

Seed point: This seed point is created interactively by the user, and it is composed by the x and y coordinates of the pixel where it was assigned.

Iterations: In experiments no more than 10 or 20 iterations are required to become stable, since the segmentation problem is reduced to minimize constrained polar sections, which are generally unimodal, computed through the distance map.

The constant parameters in this method are suitable for other medical images, taking into account that the optimization techniques are directly applied to minimize one edge sectional solution for each polar area.

Segmentation by using Maximum Euclidean Distance (MED)

In this method, the Maximum Euclidean Distance is used instead the Constrained Polar Sections to perform the segmentation task. The key aspect of the method is illustrated below in Figure 13.5, which consists on the incorporation of the $Dmax$ parameter.

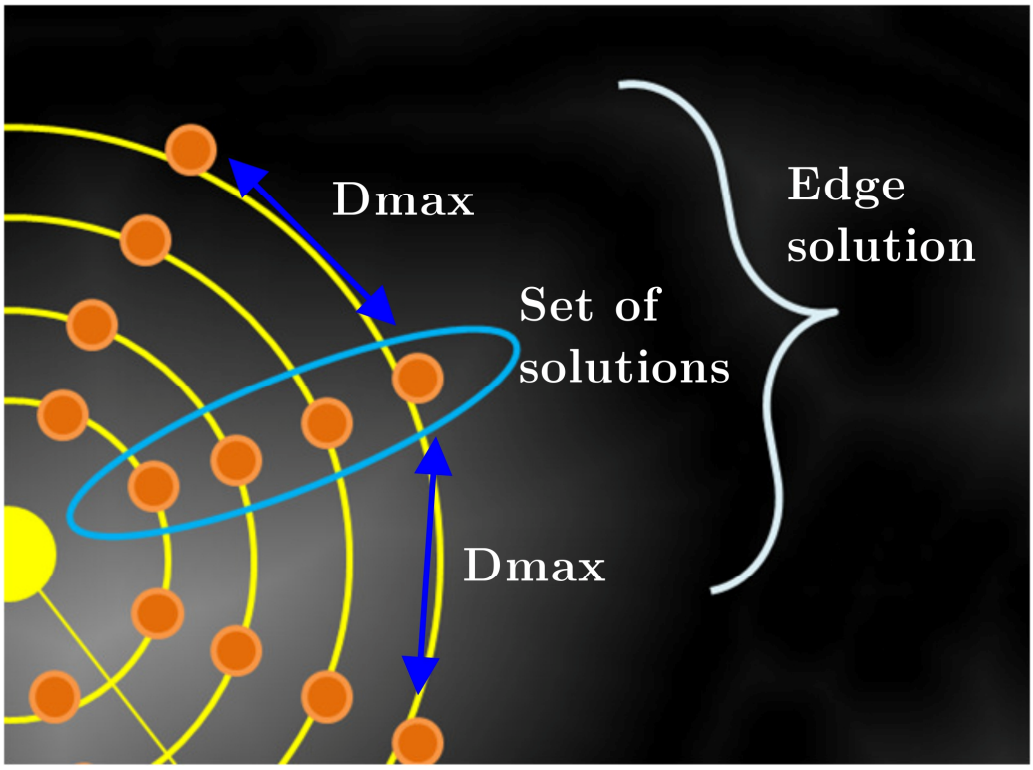


FIGURE 13.5
Incorporation of $Dmax$ in the Maximum Euclidean Distance method

This method also makes it possible to apply different meta-heuristics directly in the segmentation process. The preprocessing step is similar to the process carried out by the method of polar sections, since we first remove noise from the image followed by an edge detection between the background and the target object, and finally compute the Euclidean distance map. The initialization step is different because the polar sections are replaced by the $Dmax$ parameter, which is used to keep the search of the control points within the search space in order to minimize the closest edge solution. The $Dmax$ parameter is initialized by the user in terms of pixel separation between control points, and it is iteratively evaluated using Equation (18) (Euclidean distance) as follows:

$$Distance = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2} \quad (18)$$

The third step of this method involves the numerical optimization and the segmentation result. The numerical optimization is performed on the intensity of the Euclidean distance map, which is used as fitness function. The optimization technique is applied for each population separately in order to be placed on the closest edge solution keeping $Dmax$ between best potential solutions. Finally, when the optimization process for each set of solutions is finished, the resulting segmented object is acquired by connecting the best individual of each set to each other.

The procedure of the image segmentation method based on the Maximum Euclidean Distance is described as follows:

1. Initialize coordinates (x, y) from the origin point, $Dmax$ parameter.
2. Initialize number n of active contours and number m of control points.
3. Initialize parameters of optimization method such as iterations and population size.
4. Generate one population or swarm O_i.
5. For each population or swarm O_i:
6. Apply restriction of the search space to ignore improper solutions.
7. Perform the particular search strategy of the optimization method.
8. Stop if convergence criterion is satisfied, otherwise, repeat steps (5)-(8).

BOX 6
Algorithm of the method based on Maximum Euclidean Distance

Computational Experiments

In this section, the abovementioned methods are applied in the segmentation of human heart on Computed Tomography images and white blood cells on microscope images.

Application to Human Heart on Computed Tomography Images

We have applied the Constrained Polar Sections method on a dataset composed of 144 Computed Tomography (CT) images (size 512 x 512 pixels) from different patients. In Figure 13.6 the human heart segmentation results on a subset of CT images are illustrated. Figure 13.6(a) presents the original test images of the subset in order to increase the human perception of the segmentation task. In Figure 13.6(b) the manual delineations of the human heart made by experts are presented. Figure 13.6(c) shows the segmentation results obtained through the traditional implementation of the Active Contour Model, where the fitting problem leads to an inaccurate convergence to heart boundary. The ACM parameters used in this simulation were set as 45 control points, α = 0.017, β = 0.86 and γ = 0.45, according to similar tests reported in [21]. Figures 6(d), 6(e) and 6(f) illustrate the segmentation results obtained via the CPS method using the UMDA, PSO and DE strategies, respectively. The general parameters were set as number of contours = 9, number of control points = 45 and generations = 10. The results obtained with the CPS method are in general more accurate

than the results obtained with the classical ACM, since it fit to the real human heart boundary accurately avoiding the local minima problem.

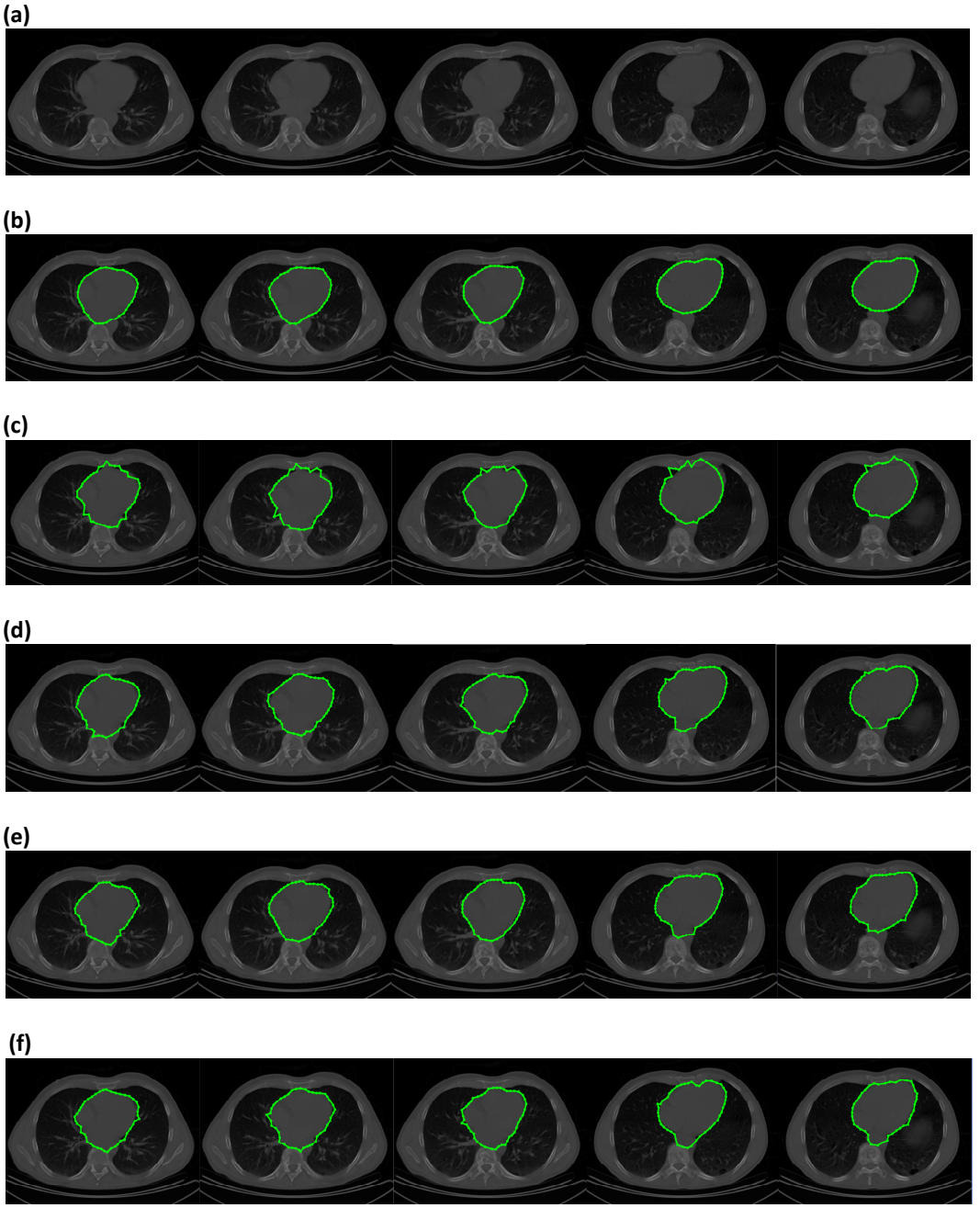


FIGURE 13.6
Human heart segmentation on CT images. (a) Test images, (b) Delineation by experts, (c) Segmentation results using the classical ACM implementation, (d) Segmentation results using UMDA, (e) Segmentation results using PSO, and (f) Segmentation results using DE

From the abovementioned dataset of CT images, in Table 13.3 a comparative analysis using the average of the segmentation results obtained from computational methods and the regions outlined by experts is introduced. The similarity measures used for this analysis are the Jaccard and Dice indexes, which evaluate the segmentation result in terms of overlapping regions and they are situated in the range [0, 1], according to [3]. This comparative analysis suggests that the CPS method using different optimization strategies can lead to more efficiency in human heart segmentation regarding the traditional Active Contour Model, which can be significantly help cardiologists in clinical practice.

TABLE 13.3
Average similarity measure among the regions segmented by ACM, and the constrained polar sections framework via UMDA, PSO and DE compared to those regions outlined by expert from the set of CT images

Comparative Studies	Similarity Measure	
	Jaccard index (J)	**Dice index (D)**
ACM vs Expert	0.5272	0.6904
UMDA vs Expert	0.7142	0.8333
PSO vs Expert	0.8260	0.9047
DE vs Expert	0.8666	0.9285

The use of optimization strategies in the CPS method provides robustness, accuracy and stability in the local minima problem, improving the segmentation accuracy. As shown in similarity analysis, the CPS method is able to detect the human heart with a high accuracy and effectiveness, which can help cardiologists to better analyze the medical images and increase their monitoring abilities.

Application to White Blood Cells on Microscope Images

The Maximum Euclidean Distance method is applied in the segmentation of white blood cells (also called leukocytes) on microscope images. The experimental data set includes images with one and two leukocytes to be detected, which are collected from the ALL-IDB database [38, 39] (http://homes.di.unimi.it/scotti/all/).

We can apply the MED method in different ways; for instance, by using the preprocessing procedure for the CPS method, or adopting techniques such as the Generalized Hough Transform [40] to detect circles avoiding the interactive seed point just applying a slight adaptation as in [41, 42]. This preprocessing step uses a binarization of the test image to separate the target objects from the background image, followed by applying the Sobel edge detector to introduce just the boundary pixels to the Generalized Hough Transform in order to detect the most promising circles close to the leukocytes. Finally, when the circle of the object of interest is detected, we can use the MED method to generate the potential solutions around this circle to adapt it to the shape of the target leukocyte.

In Figure 13.7, the preprocessing steps for the segmentation of white blood cells are illustrated. Figure 13.7(a) presents the test microscope images; Figure 13.7(b) shows the edge detection results using the Sobel operator. From the edge detection procedure, the Generalized Hough Transform is able to detect circular shapes (target leukocyte) in the image as it is shown in Figure 13.7(c). This

circle represents the input for the MED method generating multiple concentric circles as potential solutions. The best contour solution found by the method represents the final segmentation result. This MED method shows a high accuracy segmentation locating the edge of the white blood cells as it is illustrated in Figure 13.7(d).

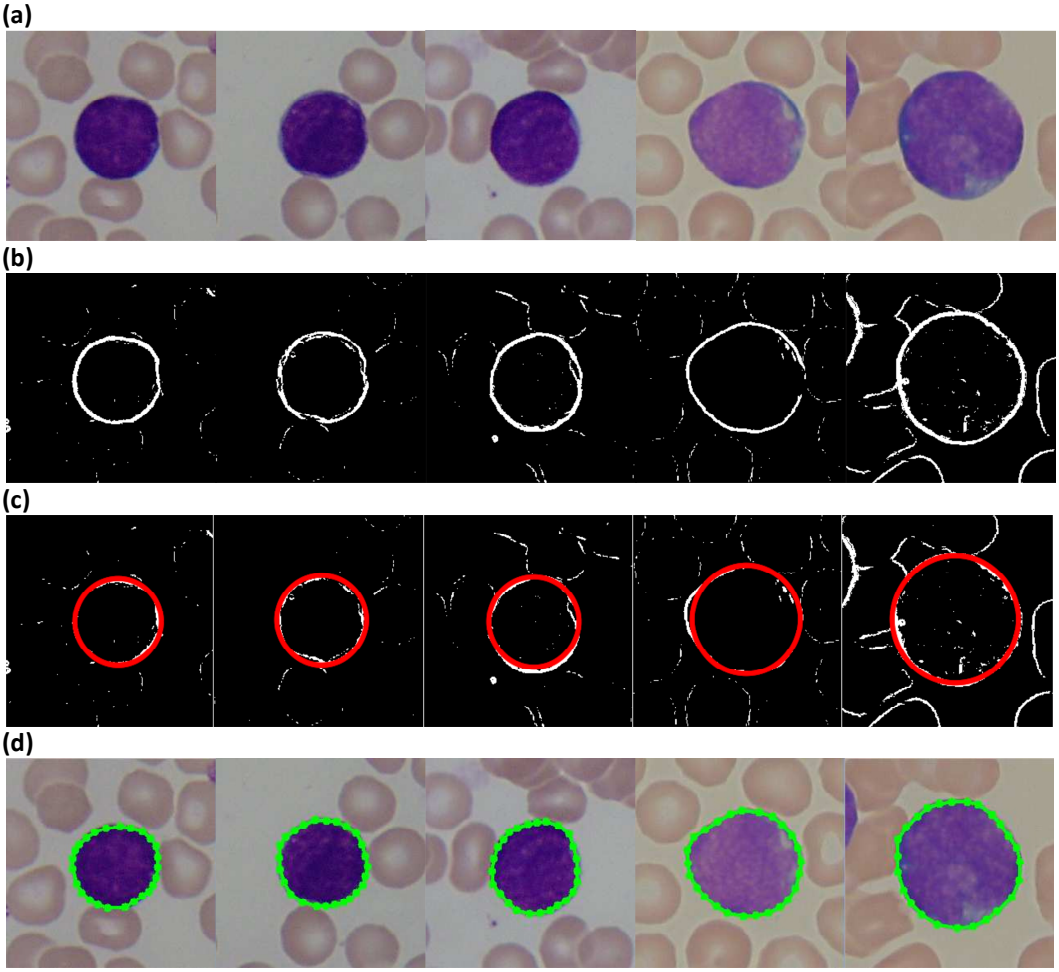
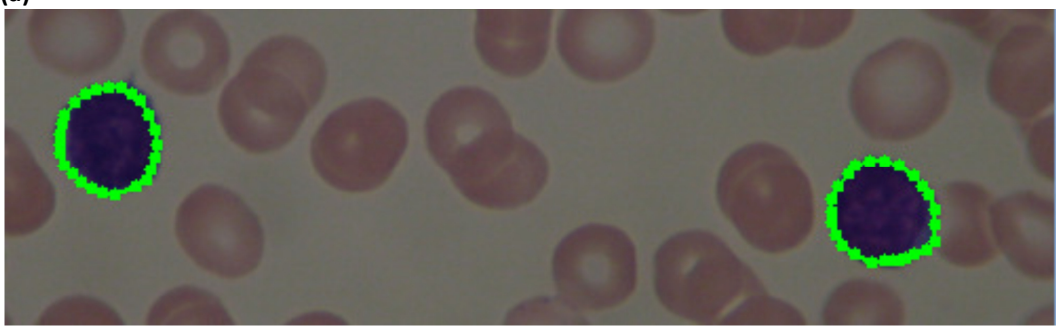


FIGURE 13.7
White blood cells segmentation. (a) Test images, (b) Edge detection, (c) Circle detection using Hough transform and (d) Segmentation results using the MED method

The aforementioned procedure to detect one leukocyte can be applied to detect multiple leukocytes in an image as it is shown in Figure 13.8(a) and (b), respectively. The procedure only depends on the number of circles which are detected by the Generalized Hough Transform.

(a)



(b)

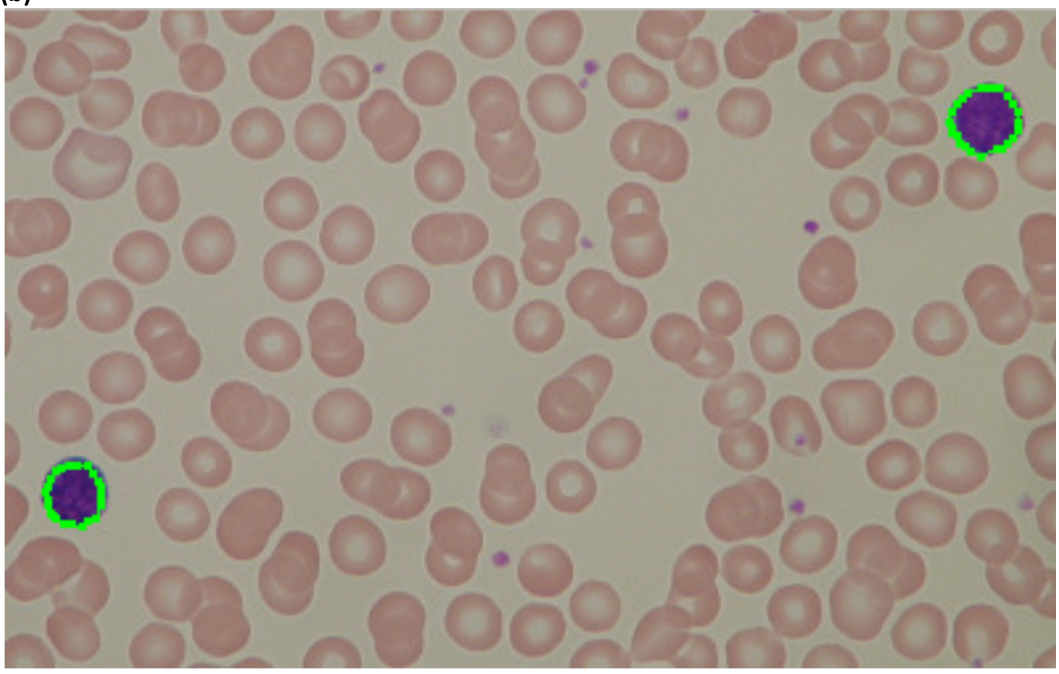


FIGURE 13.8
(a) and (b) Multiple white blood cells segmentation using the procedure illustrated in Figure 13.7

The segmentation results acquired by the CPS and MED methods show their flexibility to be applied in different problems in medical image segmentation only by using different preprocessing steps. Also, these methods are easy to adapt for working with different population-based methods, since the methodologies allow preserving the original implementation of the optimization techniques.

Concluding Remarks

In this chapter, we have presented two different frameworks based on the theory of active contour models for medical image segmentation. The first framework through an interactive initialization divides the target object in polar sections, in order to avoid the local minima and the sensitivity to initial positioning problems regarding the classical implementation of the Active Contour Model. This framework was applied in the human heart segmentation on Computed Tomography images using different optimization strategies such as Estimation of Distribution Algorithms, Particle Swarm Optimization and Differential Evolution. Moreover, the second framework involves a distance parameter between potential solutions instead the constrained polar sections, and it also introduces the Generalized Hough Transform to avoid the interactive seed point to perform the segmentation task. This second framework was applied in the white blood cells segmentation problem on microscope images. The experimental results revealed that by using optimization techniques as strategy search in both frameworks, it is possible to attain high accuracy, robustness and efficiency in the human heart and leukocytes segmentation. Finally, the experimental results have also shown that these frameworks are appropriate and efficient to be used in different medical segmentation problems, which is very useful for clinical decision support applications.

Acknowledgements

This work has been supported by the National Council of Science and Technology of México (CONACYT) under Grant 241224-218157. The authors wish to thank to the cardiology department of the Mexican Social Security Institute, UMAE T1 León for the precious clinical advice and for kindly providing us the sources of cardiac CT images.

References

1. van Rikxoort EM, Isgum I, Arzhaeva Y, Staring M, Klein S, Viergever MA, et al. Adaptive local multi-atlas segmentation: Application to the heart and the caudate nucleus. Medical image analysis. 2010 Feb, 14(1):39–49.
2. Nyma A, Kang M, Kwon YK, Kim CH, Kim JM. A Hybrid Technique for Medical Image Segmentation. Journal of Biomedicine and Biotechnology. 2012, 2012(830252):7.
3. Hsu WY. Improved watershed transform for tumor segmentation: Application to mammogram image compresion. Expert Systems with Applications. 2012, 39:3950–3955.
4. Boykov Y, Jolly MP. Interactive Organ Segmentation using Graph Cuts. Proceedings of Medical Image Computing and Computer-Assisted Intervention. 2000, p. 276–286.
5. Schmidt FR, Toppe E, Cremers D. Efficient Planar Graph Cuts with Applications in Computer Vision. IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2009.
6. Davuluri P, Wu J, Tang Y, et al. Hemorrhage Detection and Segmentation in Traumatic Pelvic Injuries. Computational and Mathematical Methods in Medicine. 2012, 2012(898430):12.
7. Liu X, Haider MA, Yetik IS. Unsupervised 3D Prostate Segmentation Based on Diffusion-Weighted Imaging MRI Using Active Contour Models with a Shape Prior. Journal of Electrical and Computer Engineering. 2011, 2011(410912):11.
8. Middleton I, Damper RI. Segmentation of magnetic resonance images using a combination of neural networks and active contour models. Medical Engineering & Physics. 2004 Jan, 26(1):71–86.
9. Zhu X, Zhang P, Shao J, Cheng Y, Zhang Y, Bai J. A snake-based method for segmentation of intravascular ultrasound images and its in vivo validation. Ultrasonics. 2011, 51:181–189.
10. Kass M, Witkin A, Terzopoulos D. Snakes: Active contour models. International Journal of Computer Vision. 1988, 1:321–331.
11. Cohen LD, Cohen I. Finite-Element Methods for Active Contour Models and Balloons for 2-D and 3-D Images. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1993 Nov, 15(11):1131–1147.
12. Chen X, Udupa JK, Bagci U, Zhuge Y, Yao J. Medical Image Segmentation by Combining Graph Cuts and Oriented Active Appearance Models. IEEE Transactions on Image Processing. 2012, 21(4):2035–2046.
13. Wang L, He L, Mishra A, Li C. Active contours driven by local Gaussian distribution fitting energy. Signal Processing. 2009, 89:2435–2447.
14. Liu B, Cheng HD, Huang J, Tian J, Tang X, Liu J. Probability density difference-based active contour for ultrasound image segmentation. Pattern Recognition. 2010, 43:2028-2042.
15. Ballerini L. Genetic snakes for medical images segmentation. In: Proceedings of the First European Workshops on Evolutionary Image Analysis, Signal Processing and Telecommunications. vol. 1596/1999 of EvoIASP '99/EuroEcTel '99. Springer-Verlag. 1999. p. 59–73.
16. Talebi M, Ayatollahi A, Kermani A. Medical ultrasound image segmentation using genetic active contour. Journal of Biomedical Science and Engineering. 2011, 4:105-109.
17. Novo J, Santos J, Penedo MG. Topological Active Models Optimization with Differential Evolution. Expert Systems with Applications. 2012, 00:1–28.
18. Cruz-Aceves I, Avina-Cervantes JG, Lopez-Hernandez JM, Garcia-Hernandez MG, Ibarra-Manzano MA. Unsupervised cardiac image segmentation via multi-swarm active contours

with a shape prior. Computational and Mathematical Methods in Medicine. 2013, 2013(909625):10.
19. Cruz-Aceves I, Avina-Cervantes JG, Lopez-Hernandez JM, et al. Automatic image segmentation using active contours with univariate marginal distribution. Mathematical Problems in Engineering. 2013, 2013(419018):12.
20. Shahamatnia E, Ebadzadeh MM. Application of Particle Swarm Optimization and Snake Model Hybrid on Medical Imaging. In: Proceedings of the third International Workshop on Computational Intelligence in Medical Imaging. IEEE service center. 2011. p. 1–8.
21. Tseng CC, Hsieh JG, Jeng JH. Active contour model via multi-population particle swarm optimization. Expert Systems with Applications. 2009, 36:5348–5352.
22. Abdi MJ, Hosseini SM, Rezghi M. A Novel Weighted Support Vector Machine Based on Particle Swarm Optimization for Gene Selection and Tumor Classification. Computational and Mathematical Methods in Medicine. 2012, 2012(320698):7.
23. Petrovsky A, A S, Mccall J. Optimising cancer chemotherapy using an estimation of distribution algorithm and genetic algorithms. Genetic and Evolutionary Computation Conference, GECCO-2006. 2006, p. 413–418.
24. Ho WH, Chan ALF. Hybrid Taguchi-Differential Evolution Algorithm for Parameter Estimation of Differential Equation Models with Application to HIV Dynamics. Mathematical Problems in Engineering. 2011, 2011(514756):14.
25. Eberhart RC, Kennedy J. A new optimizer using particle swarm theory. In Proceedings of the sixth international symposium on micro machine and human science. 1995, p.39–43.
26. Shi Y, Eberhart RC. A Modified Particle Swarm Optimizer. In Proceedings of the IEEE Congress on Evolutionary Computation. 1998, p. 69–73.
27. Storn R, Price KV. Differential Evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces. International Computer Sciences Institute, Berkeley, CA, USA, 1995. TR-95-012.
28. Storn R, Price KV. Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. Journal of Global Optimization. 1997, 11:341-359.
29. Larrañaga P, Lozano JA. Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation. Kluwer, Boston, MA, 2002.
30. Mühlenbein H, PaaB G. From recombination of genes to the estimation of distributions I. Binary parameters. Parallel Problem Solving from Nature. 1996, p. 178–187.
31. Pelikan M, Goldberg DE, Lobo F. A survey of optimization by building and using probabilistic models. Computational Optimization and Applications. 2002, 21:5–20.
32. Muehlenbein H. The equation for response to selection and its use of prediction. Evolutionary Computation. 1997, 5(3):303–346.
33. Bashir S, Naeem M, Shah SI. A comparative study of heuristic algorithms: GA and UMDA in spatially multiplexed communication systems. Engineering Applications of Artificial Intelligence. 2010, 23:95–101.
34. Lozada-Chang LV, Santana R. Univariate marginal distribution algorithm dynamics for a class of parametric functions with unitation constraints. Information Sciences. 2011, 181:2340–2355.
35. Neshat M, Sargolzaei M, Toosi AN, Masoumi A. Hepatitis Disease Diagnosis Using Hybrid Case Based Reasoning and Particle Swarm Optimization. ISRN Artificial Intelligence. 2012, 2012(609718):6.
36. Cruz-Aceves I, Avina-Cervantes JG, Lopez-Hernandez JM, Gonzalez-Reyna SE. Multiple Active Contours Driven by Particle Swarm Optimization for Cardiac Medical Image

Segmentation. Computational and Mathematical Methods in Medicine. 2013, 2013(132953):13.
37. Cruz-Aceves I, Avina-Cervantes JG, Lopez-Hernandez JM, Rostro-Gonzalez H, et al. Multiple Active Contours Guided by Differential Evolution for Medical Image Segmentation. Computational and Mathematical Methods in Medicine. 2013, 2013(190304):14.
38. Labati RD, Piuri V, Scotti F. ALL-IDB: the Acute Lymphoblastic Leukemia Image Database for Image Processing. in Proceedings of the 2011 IEEE International Conference on Image Processing (ICIP 2011). 2011 Sep, 2011:2045–2048.
39. Scotti F. Robust Segmentation and Measurements Techniques of White Cells in Blood Microscope Images. In Proceddings of the 2006 IEEE Instrumentation and Measurement Technology Conference (IMTC 2006). 2006 Apr, 2006:43–48.
40. Ballard DH. Generalizing the Hough transform to detect arbitrary shapes. Pattern Recognition. 1981, 13(2):111–122.
41. Cuevas E, Diaz M, Manzanares M, Zaldivar D, Perez-Cisneros M. An Improved Computer Vision Method for White Blood Cells Detection. Computational and Mathematical Methods in Medicine. 2013, 2013(137392):14.
42. Cuevas E, Oliva D, Diaz M, Zaldivar D, Perez-Cisneros M, Pajares G. White Blood Cell Segmentation by Circle Detection Using Electromagnetism-Like Optimization. Computational and Mathematical Methods in Medicine. 2013,2013(395071):15.