

Research Article

Object Boundary Detection Using Active Contour Model via Multiswarm PSO with Fuzzy-Rule Based Adaptation of Inertia Factor

Ajay Khunteta and D. Ghosh

Department of Electronics & Communication Engineering, Indian Institute of Technology Roorkee, Uttarakhand 247667, India

Correspondence should be addressed to Ajay Khunteta; ajaykhunteta@gmail.com

Received 17 November 2015; Revised 2 June 2016; Accepted 14 July 2016

Academic Editor: Yasar Becerikli

Copyright © 2016 A. Khunteta and D. Ghosh. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Active contour models, colloquially known as snakes, are quite popular for several applications such as object boundary detection, image segmentation, object tracking, and classification via energy minimization. While energy minimization may be accomplished using traditional optimization methods, approaches based on nature-inspired evolutionary algorithms have been developed in recent years. One such evolutionary algorithm that has been used extensively in active contours is the particle swarm optimization (PSO). However, conventional PSO converges slowly and gets trapped in local minimum easily which results in inaccurate detection of concavities in the object boundary. This is taken care of by using proposed multiswarm PSO in which a swarm is set for every control point in the snake and then all the swarms search for their best points simultaneously through information sharing among them. The performance of the multiswarm PSO-based search process is further enhanced by using dynamic adaptation of the inertia factor. In this paper, we propose using a set of fuzzy rules to adjust the inertia weight on the basis of the current normalized snake energy and the current value of inertia. Experimental results demonstrate the effectiveness of the proposed method compared to conventional approaches.

1. Introduction

Active contours are deformable models that were introduced by Kass et al. for automatic object boundary detection [1]. The active contour model is an energy minimizing problem induced by specific image features such as the edge and shape information. The active contour model gradually evolves the final contour of the desired object by minimizing energy terms such as the internal and the external energy. In a digital image, these energy terms are computed from object features at some discrete points known as the control points; the contour is a spline curve fitted through these control points. As the energy terms computed at these points are minimized, they gradually move towards the desired object boundary. Thus, starting from an initial contour, the contour deforms iteratively and in the end shrinks to wrap around the object.

One of the many concerns with the active contour model is the local minimum problem. The energy minimization

procedure depends on the initial choice of the control points and hence, in case of inappropriate initialization, is highly prone to getting trapped in local minima. There are two issues associated with initialization and convergence of the active contour algorithm. First, the initial control points must generally be chosen close to the true boundary. Otherwise, the contour is likely to converge to a wrong result. The second problem is that active contours generally find it difficult to capture concave boundaries. In view of this, various approaches for snake implementation have been proposed in the literature that aim at enlarging the search space so as to enable the active contours to progress into the object concavities thereby capturing the true object boundary.

Williams and Shah [2] did an early attempt in implementing active contour using greedy algorithm for faster convergence. Another early work in this direction was proposed by Cohen [3] in which the external forces that push the curve to the edges are modified to give a more stable result. The model

makes the curve behave like a balloon which is inflated by an additional force. Wang et al. [4] presented multistage, optimal active contour model by using spline representation and dividing the energy minimization process into multiple stages. Yezzi Jr. et al. [5] proposed geometric snake model for segmentation of medical images. This method is based on defining feature-based metrics on a given image. An active contour algorithm with the objective of better localization was developed by Xu and Prince [6]. This method uses a new external force for active contour, called the gradient vector flow (GVF), which is computed as the diffusion of the gradient vectors of the edge map derived from the image. A “Segmented Snake” approach was proposed by Wong et al. in [7]. In this, the problem of global optimization of the closed snake curve is converted into local optimization of a number of open Segmented Snake curves. A “Directional Snake” model is presented in [8] which considers both the gradient strength and the gradient direction of the image to guide the snake towards the appropriate edges. Chan and Vese proposed active contour without edges in [9]. This model detects objects whose boundaries are not defined by gradient. Energy minimization is done as a case of minimal partition problem using level set formulation. In [10], Park and Keller presented a method for initialization of the snake using watershed transformation following which a coarse-to-fine strategy is used to finally capture the object boundary. Goldenberg et al. proposed a fast version of the geodesic active contour model in [11]. This object segmentation scheme is useful for tracking moving objects in a sequence of images. Adaptive snakes using expectation maximization (EM) algorithm were presented in [12]. The EM algorithm is used to update the confidence degrees associated with edge points and to estimate the object contour. Kim et al. [13] proposed object contour tracking by insertion of new points and deletion of unnecessary points to better describe and track the object’s boundary. They used more points in highly curved parts of the contour and fewer points in less curved parts. Venkatesh et al. [14] proposed multiple contour extraction using artificial neural network by splitting a single contour into as many subcontours as the objects in the image.

Although the above modifications to the traditional optimization procedure work well, they are plagued by increased computational cost. The extra energy terms and/or the complex processes involved in these methods increase the computational cost. This has motivated looking for some nonconventional techniques in optimizing the active contour model. Recently, nature-inspired evolutionary algorithms like particle swarm optimization (PSO), bacterial foraging algorithm (BFA), ant colony optimization (ACO), genetic algorithm (GA), and so forth, have been applied in solving many complex problems including the active contour model. GA approach to active contour [15, 16], ACO-based active contour [17], active contour using dynamic evolutionary algorithm [18], and active contour with honey bee mating optimization [19–22] employing PSO are some of the examples. Wang et al. [17] use ACO to search for the best path in a constrained region and reach the best feasible boundary with minimum energy function value. Horng et al. [19] used honey bee mating optimization for active contour. Each

active contour contains a chromosome that includes several genes as well as the control points of active contour. These control points are moved iteratively by minimizing the total energy of the active contour. Novo et al. [23] proposed the segmentation model that uses Differential Evolution as an evolutionary method that minimizes the decisions of the designer. Discrete particle swarm optimization algorithm to solve the branch-cut phase unwrapping problem of MRI data has been proposed by He et al. [24]. Cruz-Aceves et al. [25] proposed a novel Cardiac medical image segmentation method based on multiple active contours driven by particle swarm optimization. Xia et al. [26] proposed an improved active contour without edges to detect boundary in a given image, based on the techniques of curve evolution. A novel hybrid region-based active contour model is presented in [27] to segment medical images with intensity inhomogeneity. Boonnuk et al. [28] proposed active contour model with edge flow for texture segmentation to give better quality. Bilqis and Widita [29] studied segmentation comparison between fast marching and geodesic active contours. Results proved that the active contours method has slightly higher accuracy and sensitivity values than the fast marching method.

We recently proposed active contour model using multi-swarm PSO in which the inertia factor is adaptively varied using fuzzy rules [30]. This eventually facilitates gradual transition from global to local search as the control points move nearer to the object boundary. In this paper, we give a detailed description of our earlier proposed active contour model. Theory of the basic active contour model is given in Section 2 followed by existing PSO-based active contour models in Section 3. Section 4 presents our proposed method of multiswarm PSO and fuzzy-rule based adaptation of inertia factor in the multiswarm PSO process. Section 5 gives detailed description of implementation steps of multiswarm PSO with fuzzy adaptive inertia. Experimental results in the support of our proposed method and comparison are given in Section 6 while we draw our conclusion in Section 7.

2. Active Contour Model

Basic snake model [1] is a spline under the influence of internal and external image forces. Image forces push the snake towards the image salient features such as edges and lines. Snake is a closed curve and representing the position of points parametrically by $v(s) = (x(s), y(s))$ and energy functional can be written as

$$\begin{aligned}\xi_{\text{snake}} &= \int_0^1 \xi_{\text{snake}}(v(s)) ds \\ &= \int_0^1 [\xi_{\text{int}}(v(s)) + \xi_{\text{ext}}(v(s))] ds,\end{aligned}\tag{1}$$

where ξ_{int} represents the internal energy of spline and ξ_{ext} generates image forces. The internal spline energy and the external image energy are represented as

$$\begin{aligned}\xi_{\text{int}} &= \frac{1}{2} [\alpha(s) |v_s(s)|^2 + \beta(s) |v_{ss}(s)|^2], \\ \xi_{\text{ext}} &= -\Gamma |\Delta I(v(s))|^2,\end{aligned}\tag{2}$$

where α , β , and Γ are weights of the energy components, $\Delta I(v(s))$ is the gradient of the image intensity at $v(s)$, and $v_s(s)$ and $v_{ss}(s)$ represent the first and the second derivatives of $v(s)$, respectively. In case of digital images, spatial discretization of the contour at N distinct control points is done by the standard finite difference operators. Accordingly, internal and external energy terms are redefined in discrete domain using these transformations as

$$v_s(s) = x_s(s)^2 + y_s(s)^2 = (x_i - x_{i-1})^2 + (y_i - y_{i-1})^2, \quad (3)$$

$$v_{ss}(s) = x_{ss}(s)^2 + y_{ss}(s)^2 = (x_{i-1} - 2x_i + x_{i+1})^2 + (y_{i-1} - 2y_i + y_{i+1})^2, \quad (4)$$

$$\xi_{\text{ext}} = -\Gamma |\Delta I(x_i, y_i)|^2. \quad (5)$$

Finally, the discrete form of the snake energy is written as the sum of internal and external energies as

$$\xi_{\text{snake}} = \sum_{i=1}^N \xi_i, \quad (6)$$

where ξ_i is the local energy at the i th control point, defined as

$$\begin{aligned} \xi_i &= \frac{1}{2} \alpha ((x_i - x_{i-1})^2 + (y_i - y_{i-1})^2) \\ &+ \frac{1}{2} \beta ((x_{i-1} - 2x_i + x_{i+1})^2 + (y_{i-1} - 2y_i + y_{i+1})^2) \\ &- \Gamma |\Delta I(x_i, y_i)|^2. \end{aligned} \quad (7)$$

3. Existing PSO-Based Active Contour Models

Particle swarm optimization (PSO) is a population based evolutionary algorithm developed by Kennedy and Eberhart [31]. In this, the candidate solution, termed particles, explores the search space thoroughly to reach the point in the search space at which the objective function is minimum. In this process, velocity is constantly adjusted according to the concerned particles self-experience and global best information available from neighboring particles. Motivated by this idea of PSO, Zeng and Zhou [20] proposed an active contour model (ACM) that is based on invariant snake topology driven by PSO. This method uses swarm particles as control points on the contour to capture the object boundary. Later, Tseng et al. proposed using multipopulation PSO [21] for active contours so as to enhance the concavity searching ability. In another piece of work, snake driven PSO is used for segmenting medical images [22]. In this, snake movement around the object of interest is guided by modified velocity update equation of PSO.

In multipopulation PSO method for active contour model, control points are initialized around the boundary of the object in a manner same as in any conventional method for active contour model. Let the total number of control points around the object be N and each control point is designated as $\mathbf{q}_i = (x_i, y_i)$. The total snake energy is calculated as per (6) and (7) and subsequently minimized

using particle swarm optimization. A swarm \mathcal{Q}_i , having S number of particles, is initialized around control point \mathbf{q}_i . During the optimization process, the swarm \mathcal{Q}_i takes part in the energy minimization process. This is done by iterating PSO over the swarm \mathcal{Q}_i at the i th control point \mathbf{q}_i for some fixed number of generations.

In mathematical terms, PSO dynamics for the j th swarm particle of the i th swarm (corresponding to the i th control point) is expressed as follows:

$$\begin{aligned} \mathbf{V}_{i,j}(t+1) &= \omega \cdot \mathbf{V}_{i,j}(t) + c_1 \cdot r_1 [\mathbf{pbest}_{i,j}(t) - \mathbf{p}_{i,j}(t)] \\ &+ c_2 \cdot r_2 [\mathbf{gbest}_i(t) - \mathbf{p}_{i,j}(t)], \quad (8) \\ \mathbf{p}_{i,j}(t+1) &= \mathbf{p}_{i,j}(t) + \mathbf{V}_{i,j}(t+1), \end{aligned}$$

where ω is the inertia factor of the swarm, c_1 and c_2 are positive constants, r_1 and r_2 are random numbers between $[0, 1]$, $\mathbf{p}_{i,j}(t)$ is the position of the j th particle in the i th swarm after the t th iteration, $\mathbf{V}_{i,j}(t)$ is velocity of the j th particle in the i th swarm during t th iteration, $\mathbf{pbest}_{i,j}(t)$ denotes the best position of the j th particle in the i th swarm till t th iteration, and $\mathbf{gbest}_i(t)$ denotes the best position of the whole population of the i th swarm till t th iteration. Since the swarm is set in the two-dimensional space corresponding to the image plane, the swarm particles move along the image plane and hence the velocity $\mathbf{V}_{i,j}(t)$ consists of two components corresponding to the displacements along the horizontal (x -axis) and vertical (y -axis) directions in the image plane. The cost associated with the j th swarm particle in the i th swarm ($\mathbf{p}_{i,j}$) is defined as

$$\begin{aligned} \xi_{i,j} &= \frac{1}{2} \alpha \|\mathbf{q}_{i+1} - \mathbf{p}_{i,j}\|^2 + \frac{1}{2} \beta \|\mathbf{q}_{i+1} - 2 \cdot \mathbf{p}_{i,j} + \mathbf{q}_{i-1}\|^2 \\ &- \Gamma |\Delta I(\mathbf{p}_{i,j})|^2. \end{aligned} \quad (9)$$

Once the swarm \mathcal{Q}_i converges to the final \mathbf{gbest}_i , the location of the control point \mathbf{q}_i is updated to this \mathbf{gbest}_i position. This process is performed for all the N control points sequentially. That is, swarms are set around the control points $\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \dots, \mathbf{q}_N$ and the PSO process is repeated for every swarm. This way, one round of optimization is performed. Further rounds of this process are repeated for further energy minimization starting from control point \mathbf{q}_1 and continued to \mathbf{q}_N in every round. These rounds of optimization (snake energy minimization) are carried on for a predecided fixed number of rounds or till there is no more significant decrease in the total snake energy ξ_{snake} . The final object contour is obtained by passing a spline through the coordinates of the N control points (final \mathbf{gbest}_i location for all $i = 1, 2, \dots, N$) at the end of all rounds of optimization.

4. Proposed Method of Multiswarm PSO with Fuzzy Adaptive Inertia

Tseng et al.'s multipopulation PSO-based scheme encounters the problem due to the serial nature of the algorithm in which the position of the control points is updated sequentially one after the other while keeping the position of the other

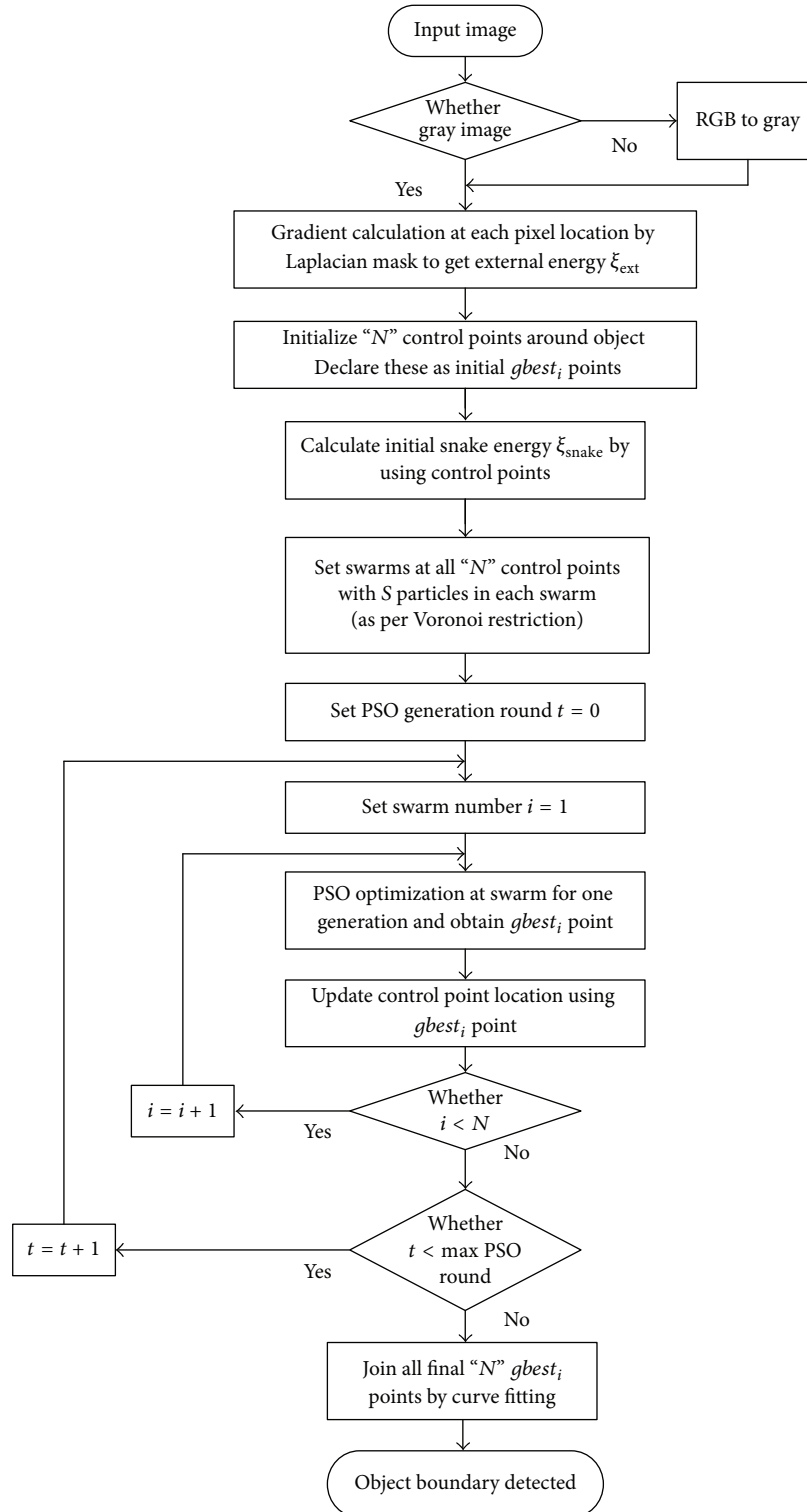


FIGURE 1: Flow chart of proposed multiswarm PSO-based active contour model.

control points fixed. In order to alleviate this problem, it seems to be reasonable, justified, and effective if the control points are updated simultaneously. In view of this, we propose a multiswarm PSO-based active contour model in which multiple swarms, one swarm at every control point, look for

the best position of the control points simultaneously via cooperation and information sharing among themselves.

Flow chart illustrating our proposed multiswarm PSO-based active contour model is given in Figure 1 and drawn as per [32].

Traditional active contour model has limitation in searching for boundary concavities. Our proposed PSO-based active contour model provides better movement of particles to enhance search ability but still the search capability is restricted due to the nonadaptive nature of the inertia value. Inertia plays key role in the PSO implementation. Inertia factor is used to balance the global and local search ability. High value of inertia causes the swarm particles to fly/swim a larger distance from one generation to the next thereby aiding in better and wider exploration of the search space. This is particularly important where the search space is large. On the other hand, it is desired that the swarm particles make very small displacements when the swarm is very close to the point of global minimum. This guarantees better stabilization and convergence of the algorithm. Thus, using a fixed high or low inertia factor, as used in [21], may not result in good convergence. Accordingly, we propose improving the performance of the search process by using dynamic adaptation of the inertia factor. We propose using a set of fuzzy rules to adjust the inertia weight on the basis of the current normalized snake energy and the current value of inertia of the swarm particles. This eventually facilitates gradual transition from global to local search as the control points move nearer to the object boundary.

As discussed above, the inertia factor ω plays a vital role in achieving better convergence. High value of inertia is used for better global exploration in search of the minimum energy point. Nevertheless, such high value is less desired near points where energy minimization is almost complete so that the contour stabilizes for better capture of the object. On the other hand, low inertia values during the initial phase may lead to the contour trapped in local minimum. Hence, an adaptive PSO with adjustable inertia factor is desired. One popularly used strategy is to reduce the inertia at a linear rate as the search progresses so that local exploration is preferred over global search in the later stage of the search process. A variation to this is to use a nonlinear change in inertia, as proposed in [33]. However, setting the rate of change in inertia to an appropriate value is again a question. Fuzzy adaptive particle swarm optimization introduced in [34, 35] uses a set of fuzzy rules for dynamic adaptation of the inertia. This way, inertia of all particles in a swarm (ω_i) is modified dynamically as per the current search result. Following the idea in [34, 35], we propose a fuzzy-rule based adaptation to the multiswarm PSO technique for active contour model given in [21]. In our work, we define a set of fuzzy rules to adjust the inertia weight on the basis of the current normalized snake energy and the current value of inertia.

In our proposed approach, we first calculate the normalized energy at all control points and the same is calculated after every round of PSO generation as follows:

$$\xi_{\text{norm}} = \frac{\xi_{\text{snake}}}{\xi_{\text{max}}}, \quad (10)$$

where ξ_{snake} is calculated using (6) after every PSO generation and ξ_{max} is the snake energy due to the initial configuration of the control points. This normalized energy value ξ_{norm} and the current value of the inertia ω for the swarm are then

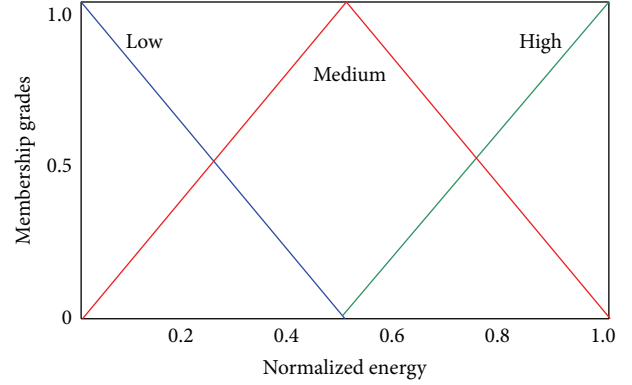


FIGURE 2: Membership function plots for fuzzy sets corresponding to *low*, *medium*, and *high* normalized energy (ξ_{norm}).

combined using fuzzy reasoning to determine the necessary change in the inertia value for use in the next round of PSO generation. For this, we formulate a set of nine “IF . . . THEN . . .” fuzzy rules, as stated below. Our proposed method for fuzzy adaptive PSO uses deductive fuzzy inference system based on this set of nine fuzzy rules.

Rule 1. IF ξ_{norm} is *low* AND ω is *low* THEN $\Delta\omega$ is *zero*.

Rule 2. IF ξ_{norm} is *low* AND ω is *medium* THEN $\Delta\omega$ is *negative*.

Rule 3. IF ξ_{norm} is *low* AND ω is *high* THEN $\Delta\omega$ is *negative*.

Rule 4. IF ξ_{norm} is *medium* AND ω is *low* THEN $\Delta\omega$ is *positive*.

Rule 5. IF ξ_{norm} is *medium* AND ω is *medium* THEN $\Delta\omega$ is *zero*.

Rule 6. IF ξ_{norm} is *medium* AND ω is *high* THEN $\Delta\omega$ is *negative*.

Rule 7. IF ξ_{norm} is *high* AND ω is *low* THEN $\Delta\omega$ is *positive*.

Rule 8. IF ξ_{norm} is *high* AND ω is *medium* THEN $\Delta\omega$ is *zero*.

Rule 9. IF ξ_{norm} is *high* AND ω is *high* THEN $\Delta\omega$ is *negative*.

Thus, our fuzzy inference system is a two-input and one-output system: two input variables (antecedent clauses) are the normalized energy ξ_{norm} term and the current inertia ω ; and the resultant output variable (consequent clause) is the desired change in the inertia value $\Delta\omega$. Antecedent clauses are linked by fuzzy AND operators. Three fuzzy sets for the antecedents and three fuzzy sets for the consequent clauses are used, as represented in Figures 2, 3, and 4. Fuzzy sets for the antecedents are labeled as “*low*,” “*medium*,” and “*high*.” Fuzzy sets for the consequent clauses are “*negative*,” “*zero*,” and “*positive*” representing corresponding change in inertia value. The range of values for all three fuzzy sets has been taken as per [35]. Normalized energy is represented between

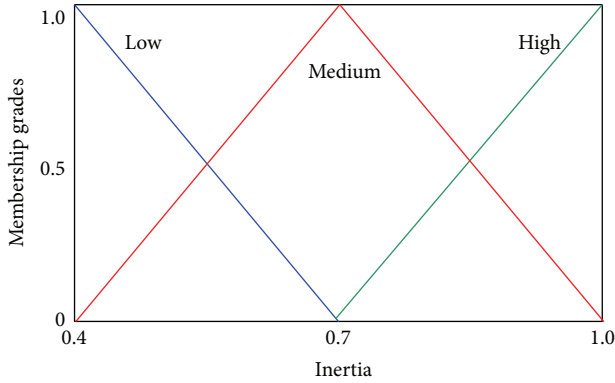


FIGURE 3: Membership function plots for fuzzy sets corresponding to *low*, *medium*, and *high* inertia (ω).

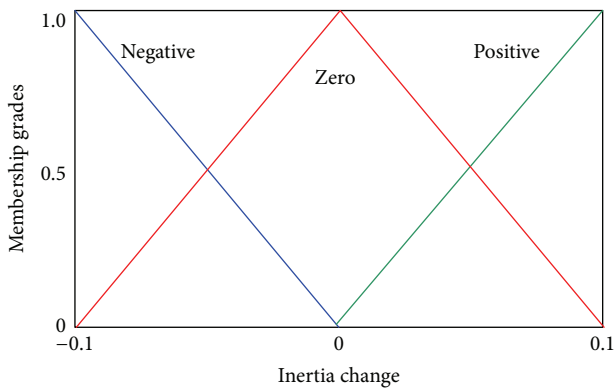


FIGURE 4: Membership function plots for fuzzy sets corresponding to *negative*, *zero*, and *positive* inertia change ($\Delta\omega$).

0 and 1 and inertia value is represented between 0.4 and 1.0. Output variable is in the form of change in inertia value and this variation is done between -0.1 and $+0.1$. Thus, the extent by which the inertia value needs to be updated is estimated on the basis of the present snake energy and the inertia value. Thus, this strategy of varying inertia value adaptively gives a trade-off between global and local exploration. The above set of fuzzy rules are eventually combined using Mamdani's max-min deductive fuzzy inference mechanism followed by centroid-based defuzzification to obtain the value of the desired $\Delta\omega$. A detailed description of the inference mechanism can be found in [36]. Flow chart of our proposed active contour model by using multiswarm PSO with fuzzy adaptive inertia factor is drawn in Figure 5.

5. Implementation Steps of Multiswarm PSO with Fuzzy Adaptive Inertia

- (1) Set PSO parameters. Calculate the image gradient at each pixel location by using the Laplacian operator. This gives the value of the external snake energy at each pixel location, as defined in (5).
- (2) Initialize N number of control points in a clockwise order and number these control points accordingly.

These control points are declared as initial $gbest_i$ points for the corresponding swarms, $i = 1, 2, \dots, N$.

- (3) Calculate initial snake energy ξ_{snake} as per (6) and (7).
- (4) Set a swarm (containing S number of particles) at each control point \mathbf{q}_i in such a manner that the position of the j th swarm particle $\mathbf{p}_{i,j}$ associated with the i th swarm is adequately close to \mathbf{q}_i so as to ensure that the control points do not cross over as the snake moves in the next step. In our algorithm, in order to ensure that no cross-over takes place, every particle of a swarm is taken such that the following condition is satisfied:

$$\|\mathbf{p}_{i,j}, \mathbf{q}_i\| \leq \|\mathbf{p}_{i,j}, \mathbf{q}_k\|, \quad \forall k \neq i, \quad (11)$$

where $\|\mathbf{u}, \mathbf{v}\|$ denotes the Euclidean distance between vectors \mathbf{u} and \mathbf{v} . This condition essentially divides the whole image plane into N nonoverlapping Voronoi regions, each region corresponding to every control point \mathbf{q}_i , $i = 1, 2, \dots, N$. Thus, all the particles in the i th swarm \mathcal{Q}_i are always restricted within the i th Voronoi region that also contains the i th control point \mathbf{q}_i . Since, the position of the particles of the swarm \mathcal{Q}_i are the candidates for the new location of the control point \mathbf{q}_i in the next step, it is guaranteed that no cross-over will take place as the snake moves from one step to the next step.

- (5) Run PSO for the first generation by taking all initial particle positions as $pbest_{i,j}$ and update $gbest_i$ values and control point \mathbf{q}_i locations.
- (6) Calculate velocity and then move swarm particles for the next generation, starting from the first swarm associated with the first control point and continuing till the last control point, as per (8). Based on these new positions of the particles, check fitness of all particles using (9) and accordingly update $pbest_{i,j}$ and $gbest_i$ values. Declare position of $gbest_i$ particle as the updated location of i th control point.
- (7) Calculate snake energy at all control points ($gbest_i$ points) and ξ_{norm_i} for all control points. Next, using ξ_{norm_i} and ω_i values together with the set of fuzzy rules, as described above, update $\omega = \omega + \Delta\omega$ for all control points. These new values of swarm inertia are to be used in the next generation of PSO.
- (8) For starting the next PSO generation round at all control points, go back to Step (6) and run PSO using the velocity and position of the particles obtained in the previous generation. Repeat till maximum number of generations are complete or until total snake energy stabilizes.

6. Experimental Results and Comparison

In our experiments, we applied our proposed fuzzy adaptive multiswarm PSO method on four images and our method is compared with the traditional snake [1], multi-population PSO-based method, as given in [21], and our

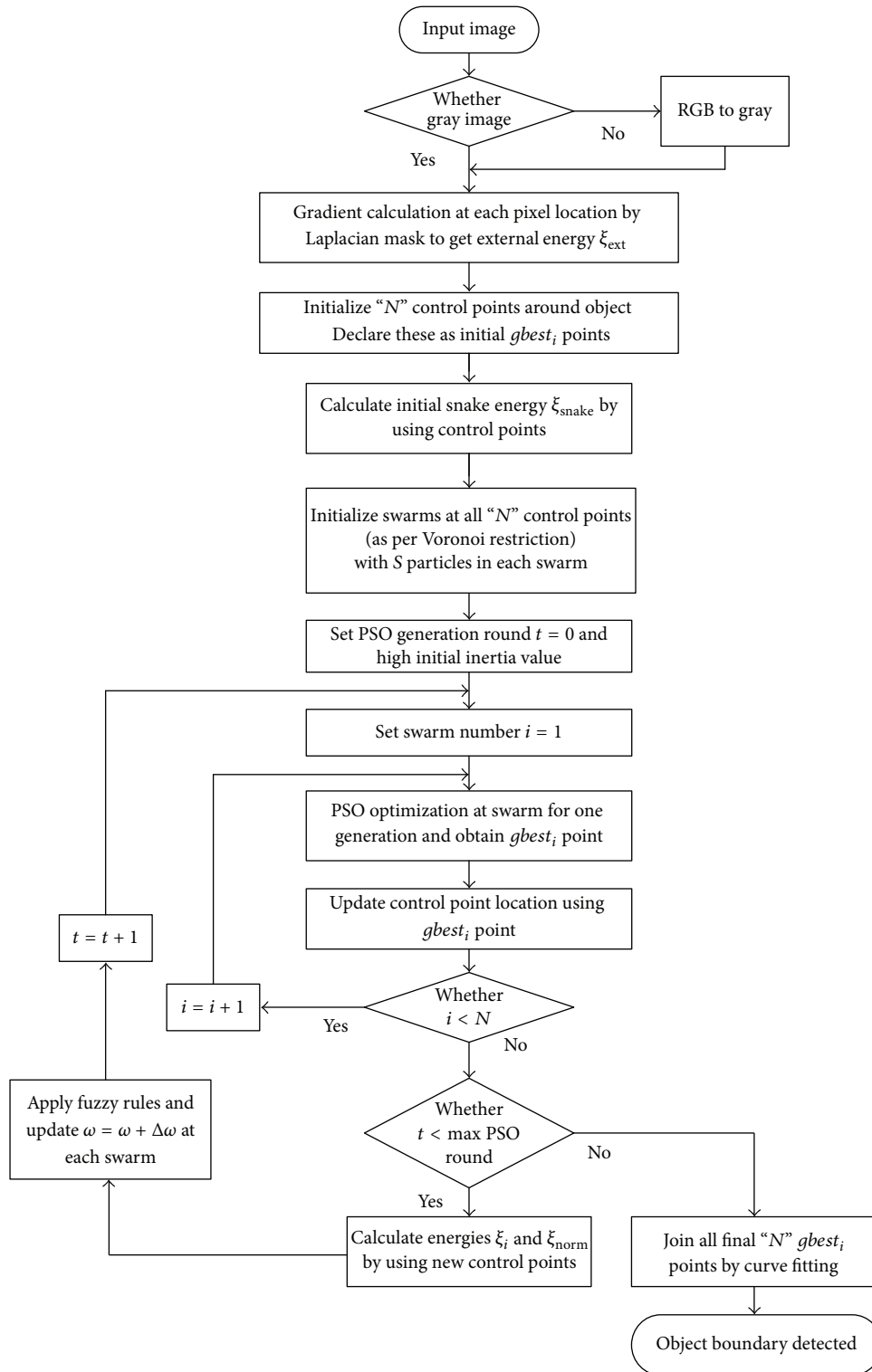


FIGURE 5: Flow chart of proposed active contour model by using multiswarm PSO with fuzzy adaptive inertia factor.

proposed multiswarm PSO method. In the first set of experiments, four images were taken. “Star” and “Vase” images are of sharp boundaries. “Peppers” and “MRI” images are taken as complex real images. The final snake energies obtained in all these cases are tabulated in Table 1.

As far as results are concerned, our proposed method is able to achieve better energy minimization compared to the other methods. This verifies the efficacy of our proposed method for catching up with concave image boundaries. Results show that our proposed multiswarm PSO method is better than

TABLE 1: Minimum snake energy values obtained by different methods.

Test images	Minimum energy by different methods		
	Basic snake method	Multipopulation method	Proposed multiswarm PSO method
Star	32.6735	30.0565	27.8738
Vase	312.3285	286.8435	268.9538
Peppers	2312.7590	2265.8750	2216.6342
MRI	71.5205	69.8676	63.8672

TABLE 2: Minimum snake energy values obtained by “multipopulation PSO” method for different cases of inertia.

Test images	Minimum snake energy values for different inertia cases					
	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
Star	32.8845	29.8747	28.9897	32.1865	26.7459	26.9856
Vase	340.8169	287.0056	292.3181	276.8824	307.9734	259.6250
Peppers	3087.5735	2237.9973	2540.4562	2316.3786	2398.0388	2147.1753
MRI	73.7734	70.8345	65.5727	66.8063	69.7296	63.7169

TABLE 3: Minimum snake energy values obtained by “proposed multiswarm PSO” method for different cases of inertia.

Test images	Minimum snake energy values for different inertia cases					
	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
Star	30.8645	27.1587	26.4797	32.1745	26.8359	24.6754
Vase	335.2819	286.4523	291.9311	268.6326	307.6539	243.1750
Peppers	3080.6115	2138.3984	2542.7400	2307.3176	2396.2976	1969.1973
MRI	70.3700	63.9098	65.1297	66.6803	69.0526	57.1506

the traditional snake and the multipopulation PSO-based methods. Our proposed multiswarm PSO helps in more stable convergence and better minimization of snake energy due to the parallel nature of the algorithm.

In the next set of experiments, we demonstrate the effectiveness of the proposed fuzzy based inertia adaptation scheme. For this, we compare our proposed PSO-based active contour models with multipopulation PSO method [21] using six different cases of inertia factor variations, as listed below.

Case 1. Fixed low inertia value ($\omega = 0.4$).

Case 2. Fixed high inertia value ($\omega = 0.95$).

Case 3. Linearly varying inertia using index $\delta = 1.0$ in (12).

Case 4. Nonlinear inertia variation, index $\delta = 0.7$ in (12).

Case 5. Nonlinear inertia variation, index $\delta = 1.3$ in (12).

Case 6. Fuzzy adaptive inertia variation.

Nonlinear variation of inertia factor for dynamic adaptation in PSO has been discussed in [33] in which the inertia factor is adapted as

$$\omega_{\text{iter}} = \frac{(\text{iter}_{\text{max}} - \text{iter})^\delta}{(\text{iter}_{\text{max}})^\delta} * (\omega_{\text{initial}} - \omega_{\text{final}}) + \omega_{\text{final}}, \quad (12)$$

where linear or nonlinear variation is decided by proper choice of the index value “ δ ”. In our experiments, $\omega_{\text{initial}} = 0.95$, $\omega_{\text{final}} = 0.4$, and iter is the iteration number in the PSO algorithm. Here one PSO iteration means one complete round of PSO optimization through all control points for multiswarm implementation. In case of multiswarm PSO, in each iteration, PSO runs for one generation only at each control point. To maintain fair comparison, region of swarm initialization as well as number of control points and number of particles in each swarm are taken same in all the above listed cases for one image.

Experiments have been performed on the same set of four images by using three methods, namely, multipopulation PSO, proposed multiswarm PSO, and proposed fuzzy adaptive multiswarm PSO-based method, with all the above six cases of inertia. Final contour energies obtained in these methods are given in Tables 2-3. Total contour energy versus computation time plots for all the above six cases for three images are plotted in Figures 6–11. Results obtained by multipopulation PSO-based method are shown in Figures 12–15 and results obtained by our proposed multiswarm PSO-based method are shown in Figures 16–19.

High or low inertia values are to some extent capable of capturing concavities. Results are generally better when inertia factor is varied linearly or nonlinearly in comparison to the fixed low or high value. Results show that our proposed fuzzy adaptive multiswarm PSO-based method is always able to provide better energy minimization than the

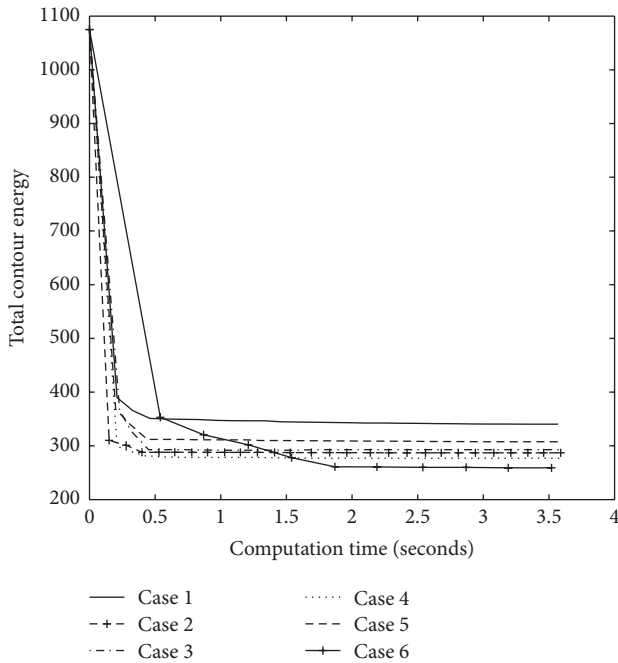


FIGURE 6: Contour energy versus computation time plot for “multipopulation PSO” method with six different cases of inertia in case of “Vase” image.

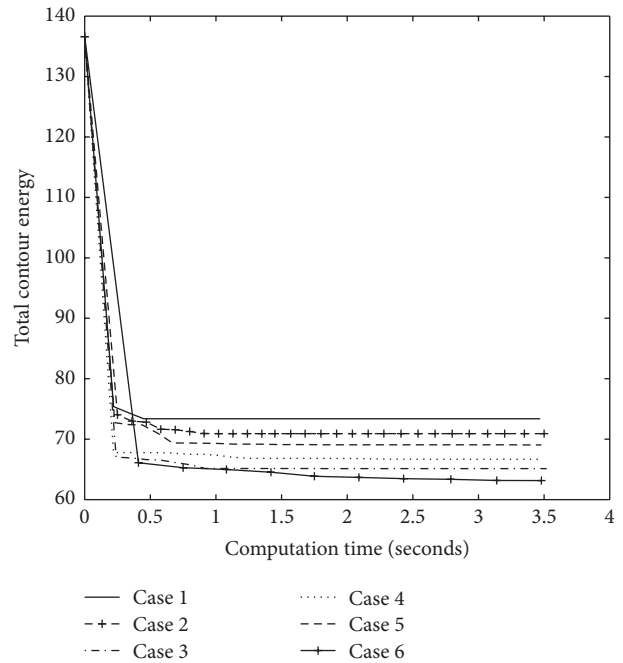


FIGURE 8: Contour energy versus computation time plot for “multipopulation PSO” method with six different cases of inertia in case of “MRI” image.

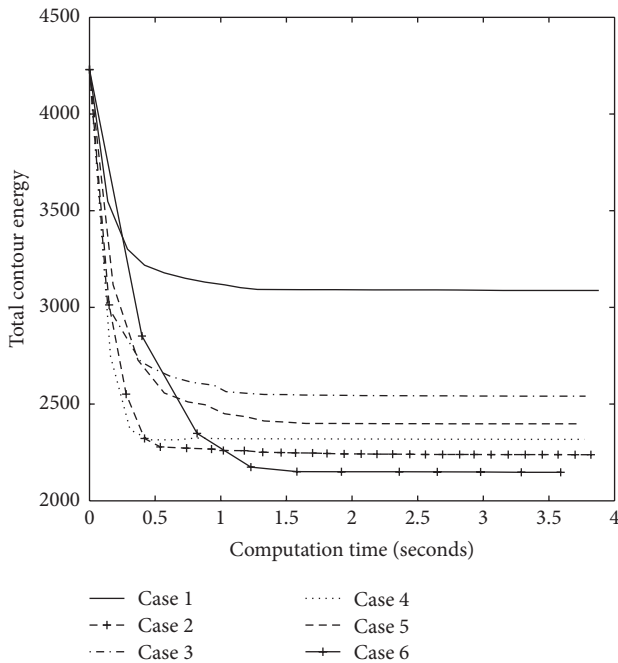


FIGURE 7: Contour energy versus computation time plot for “multipopulation PSO” method with six different cases of inertia in case of “Peppers” image.

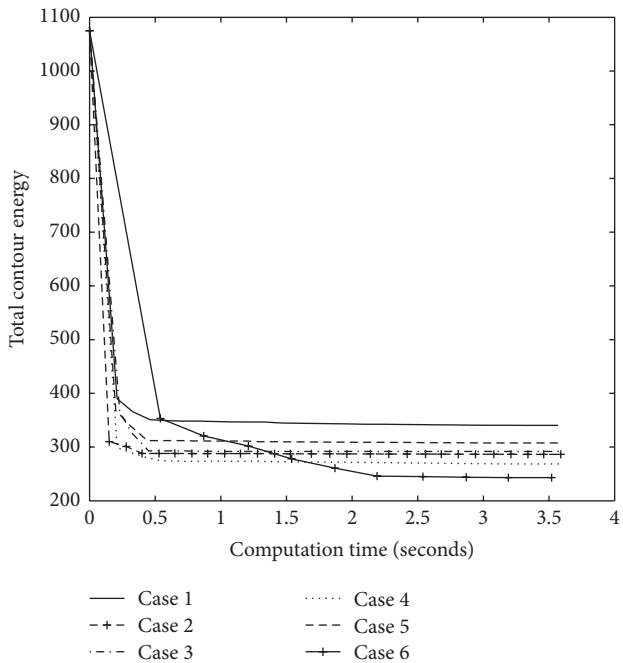


FIGURE 9: Contour energy versus computation time plot for “proposed multiswarm PSO” method with six different cases of inertia in case of “Vase” image.

multipopulation PSO method. Our proposed fuzzy adaptive multiswarm PSO method is capable of capturing the true boundaries including the concavities at the cost of increased computational time.

As we observe in the contour energy versus computation time plots given in Figures 6–11, our proposed fuzzy

adaptive multiswarm PSO-based method takes more number of iterations to converge but converges with lesser contour energy. Contour energy versus computation time plot have been drawn by choosing number of PSO rounds (for different cases of comparison) in such a manner that all the cases of

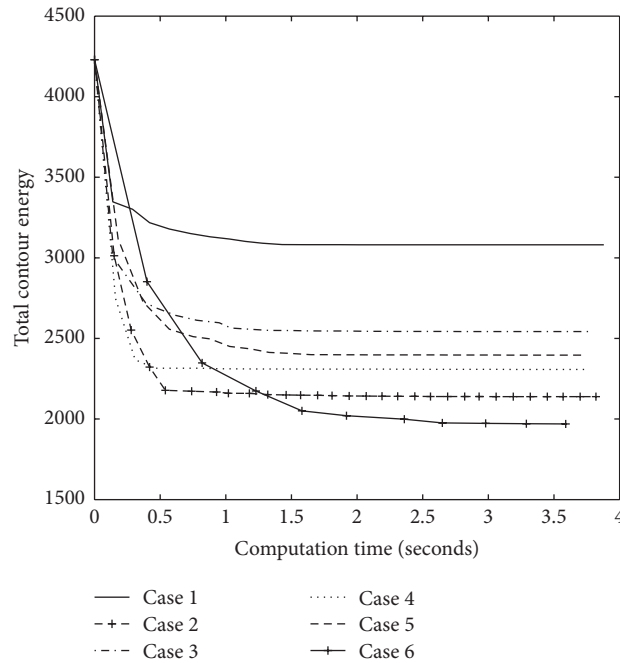


FIGURE 10: Contour energy versus computation time plot for “proposed multiswarm PSO” method with six different cases of inertia in case of “Peppers” image.

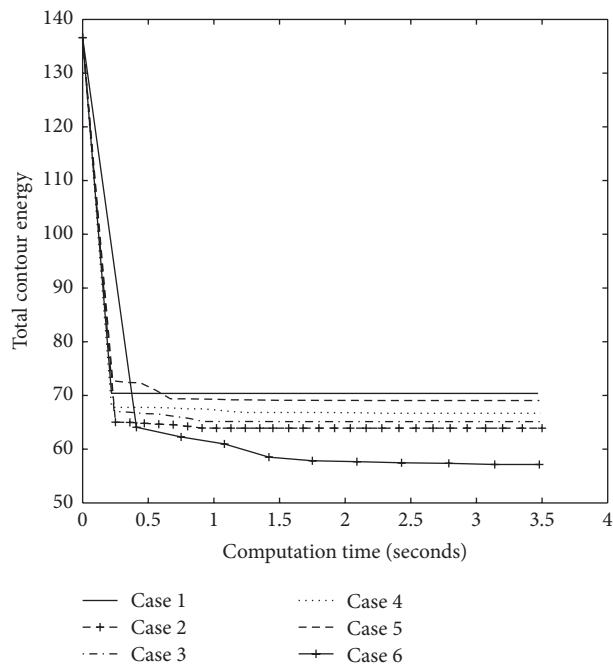


FIGURE 11: Contour energy versus computation time plot for “proposed multiswarm PSO” method with six different cases of inertia in case of “MRI” image.

the experiment run for almost similar time during the energy minimization process.

PSO parameters used in our implementation are as follows:

- (i) Population size (particles in each swarm) = 20.
- (ii) Number of generations = 10.
- (iii) Acceleration constants $c_1 = c_2 = 1.5$.

7. Conclusion

PSO-based active contour model has been used extensively for image segmentation. However, the existing multipopulation PSO-based active contour model gets trapped in local minimum easily. The problem of local minimum results in inaccurate detection of concavities in the object boundary. In our proposed active contour model, we used multiswarm

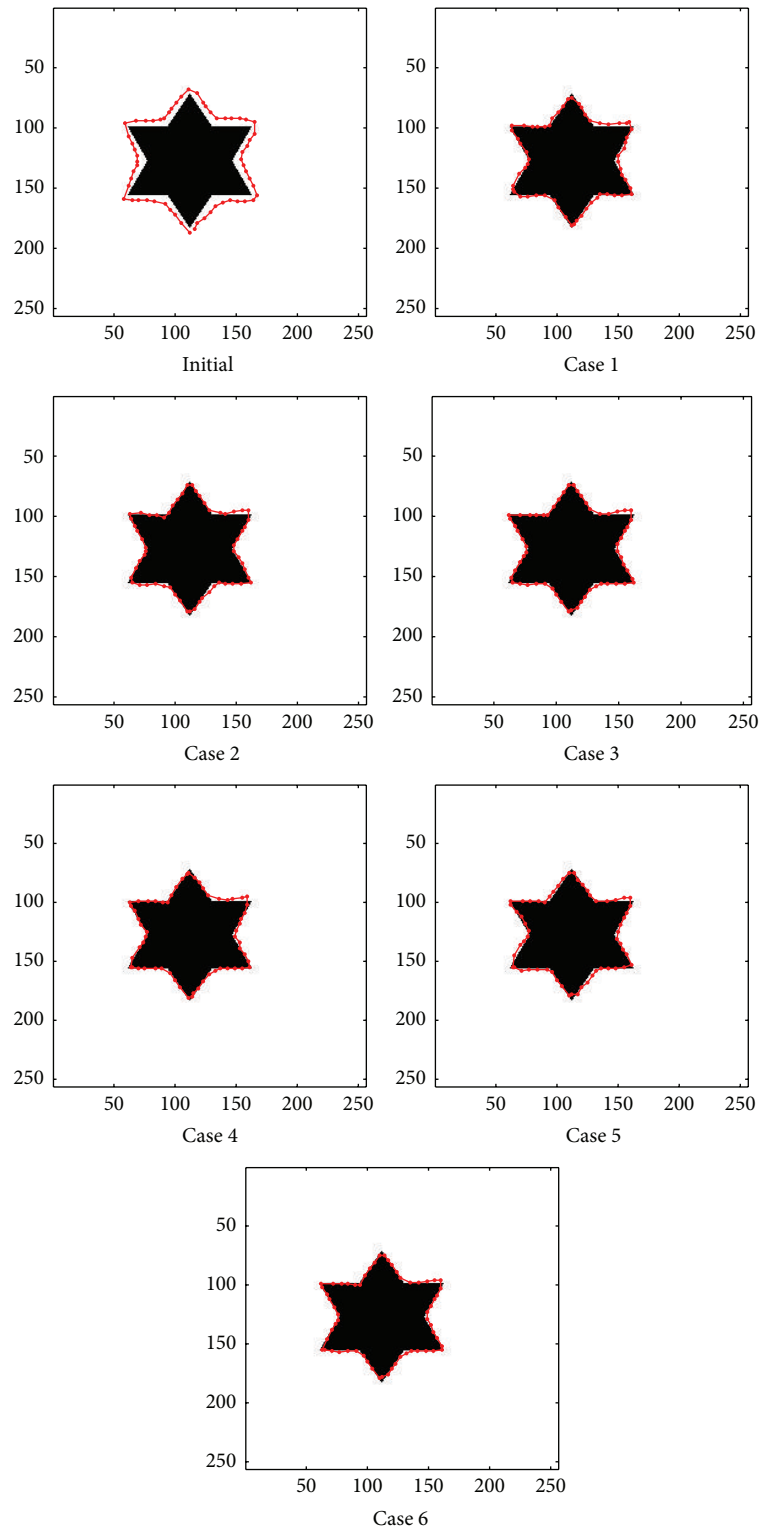


FIGURE 12: Active contour model implementation by “multipopulation PSO” method for different cases of inertia. Results for “Star” image with control points = 60, $\alpha = 0.05$, $\beta = 0.025$, and $\Gamma = 0.16$.

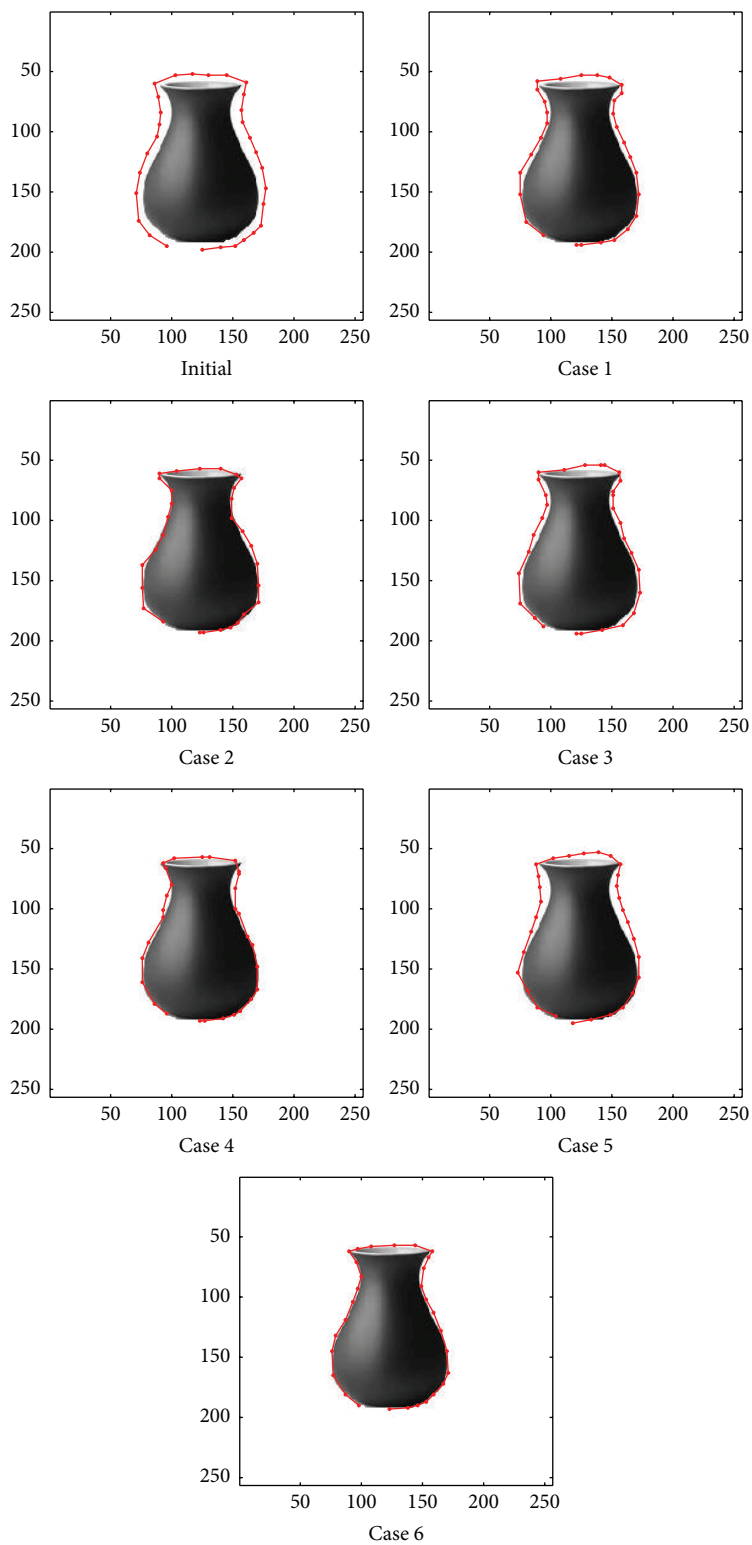


FIGURE 13: Active contour model implementation by “multipopulation PSO” method for different cases of inertia. Results for “Vase” image with control points = 30, $\alpha = 0.321$, $\beta = 0.054$, and $\Gamma = 0.003$.

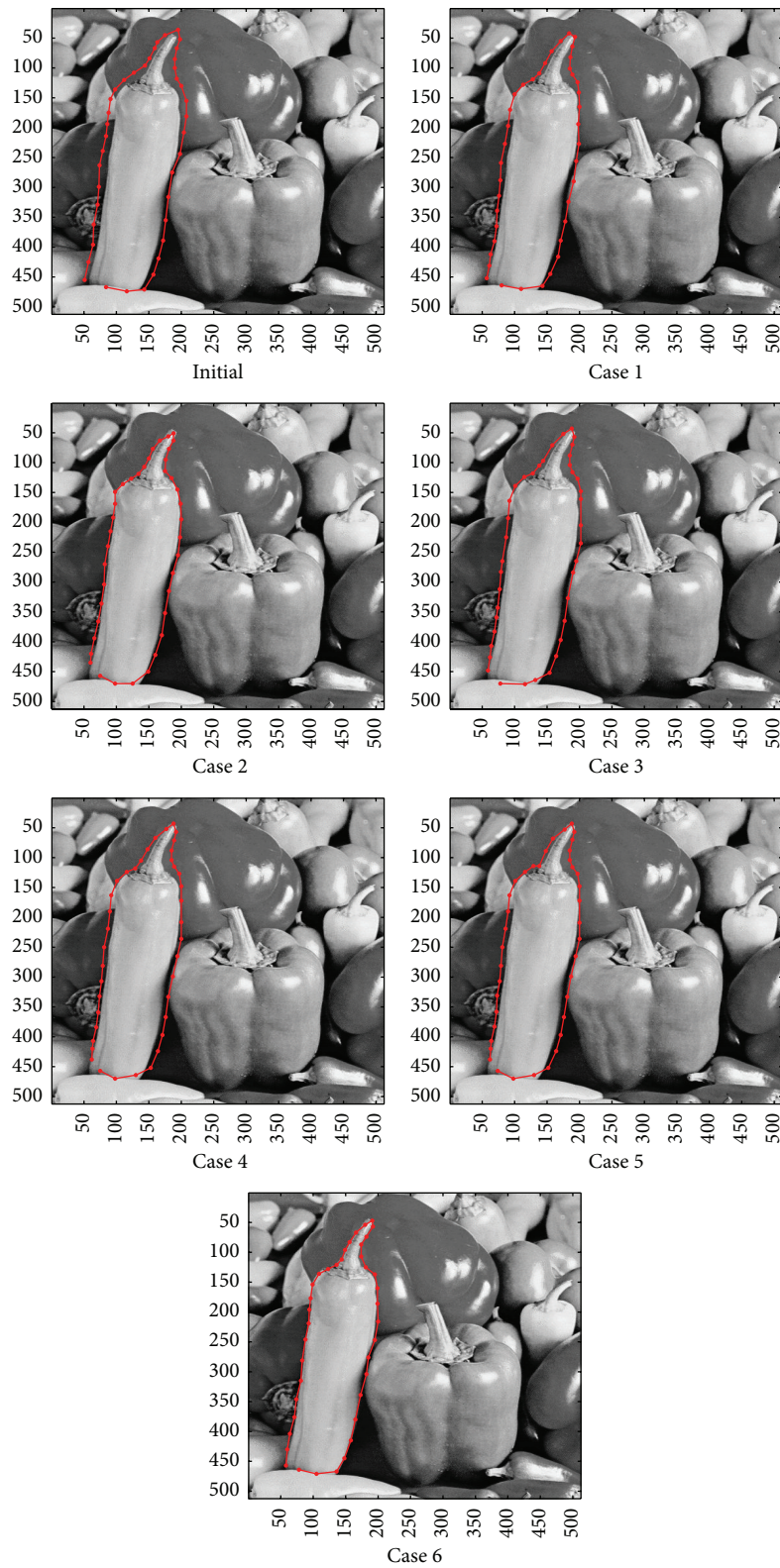


FIGURE 14: Active contour model implementation by “multipopulation PSO” method for different cases of inertia. Results for “Peppers” image with control points = 40, $\alpha = 0.25$, $\beta = 0.22$, and $\Gamma = 0.50$.

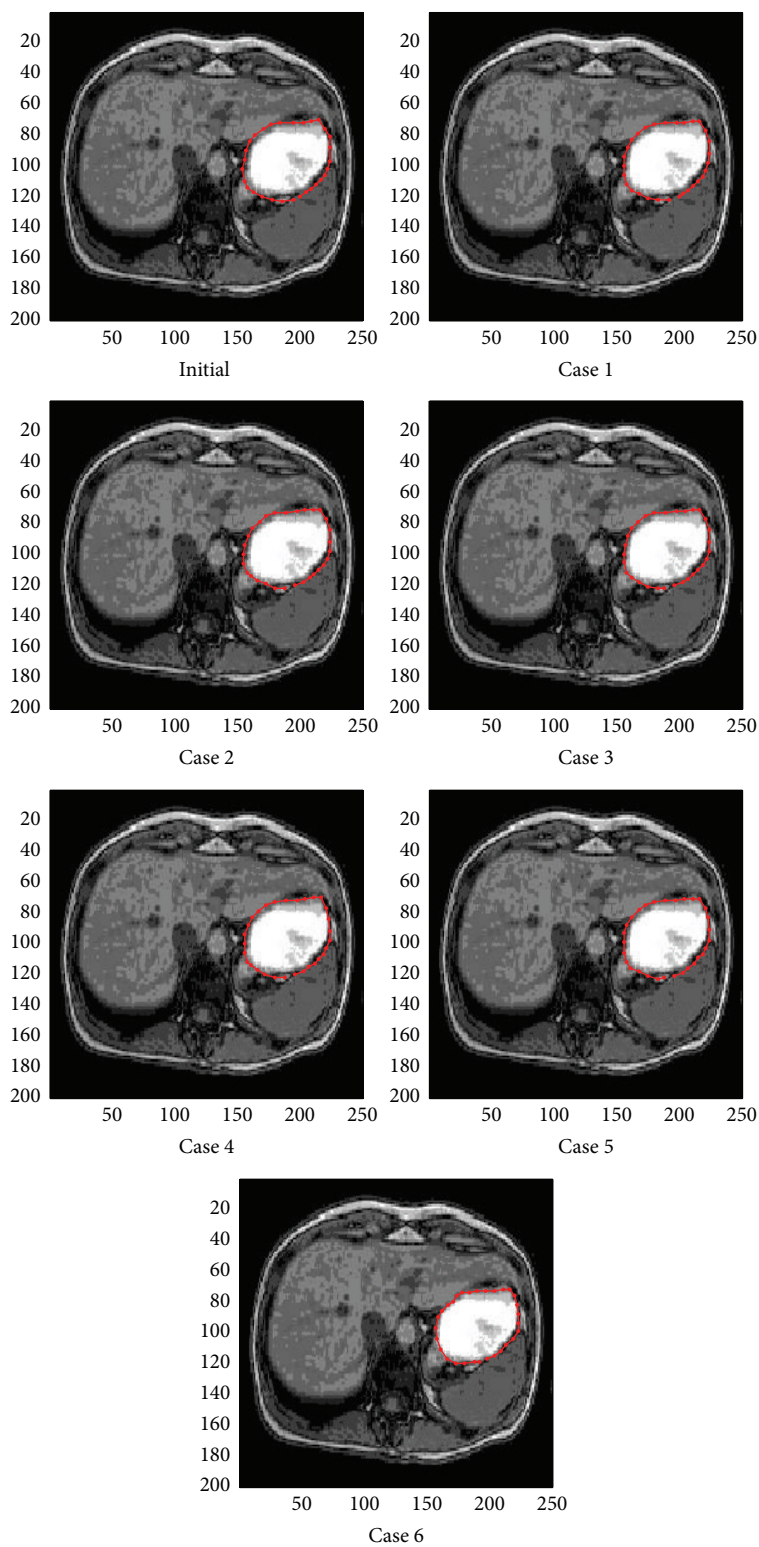


FIGURE 15: Active contour model implementation by “multipopulation PSO” method for different cases of inertia. Results for “MRI” image with control points = 30, $\alpha = 0.17$, $\beta = 0.15$, and $\Gamma = 0.20$.

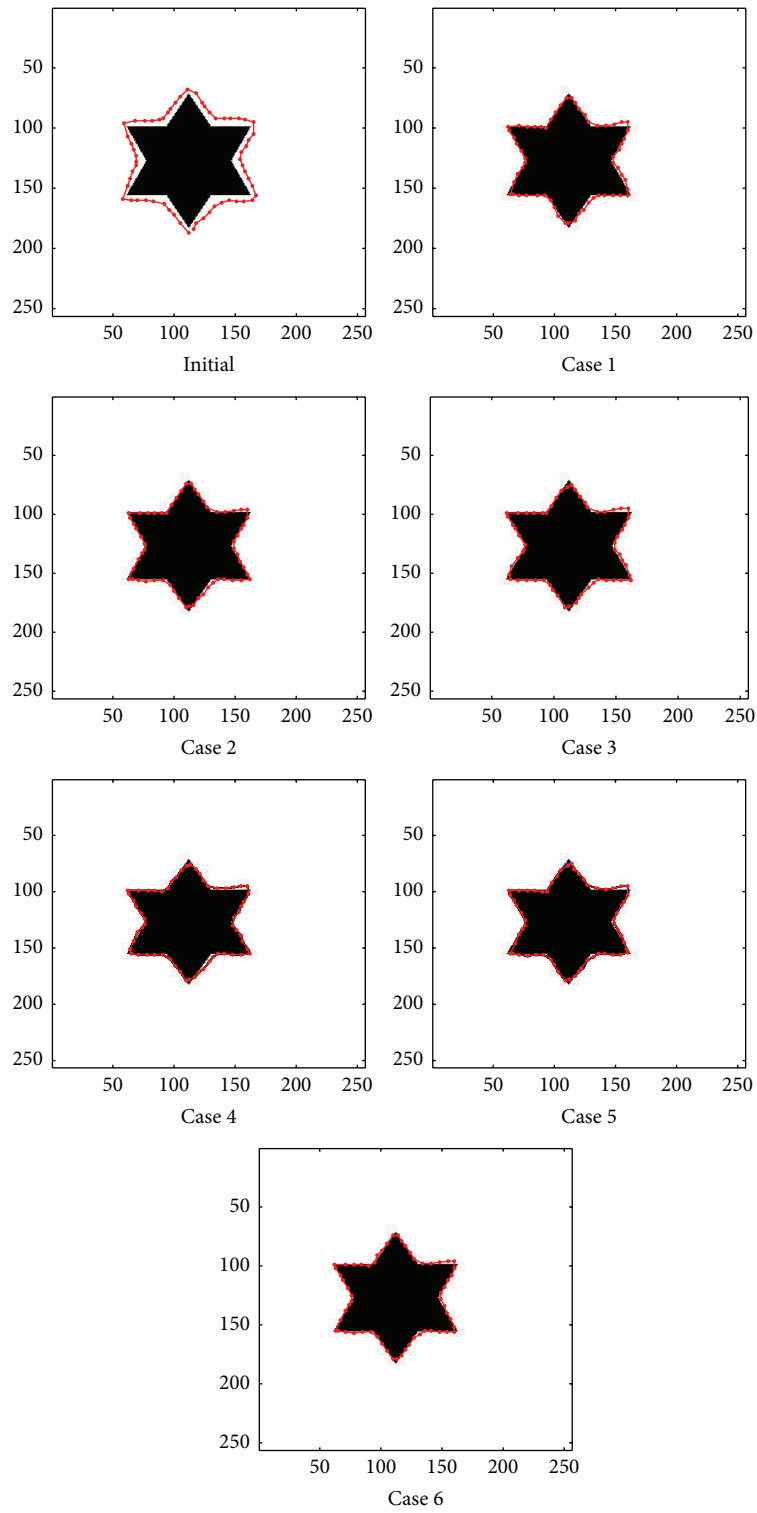


FIGURE 16: Active contour model implementation by “proposed multiswarm PSO” method for different cases of inertia. Results for “Star” image with control points = 60, $\alpha = 0.05$, $\beta = 0.025$, and $\Gamma = 0.16$.

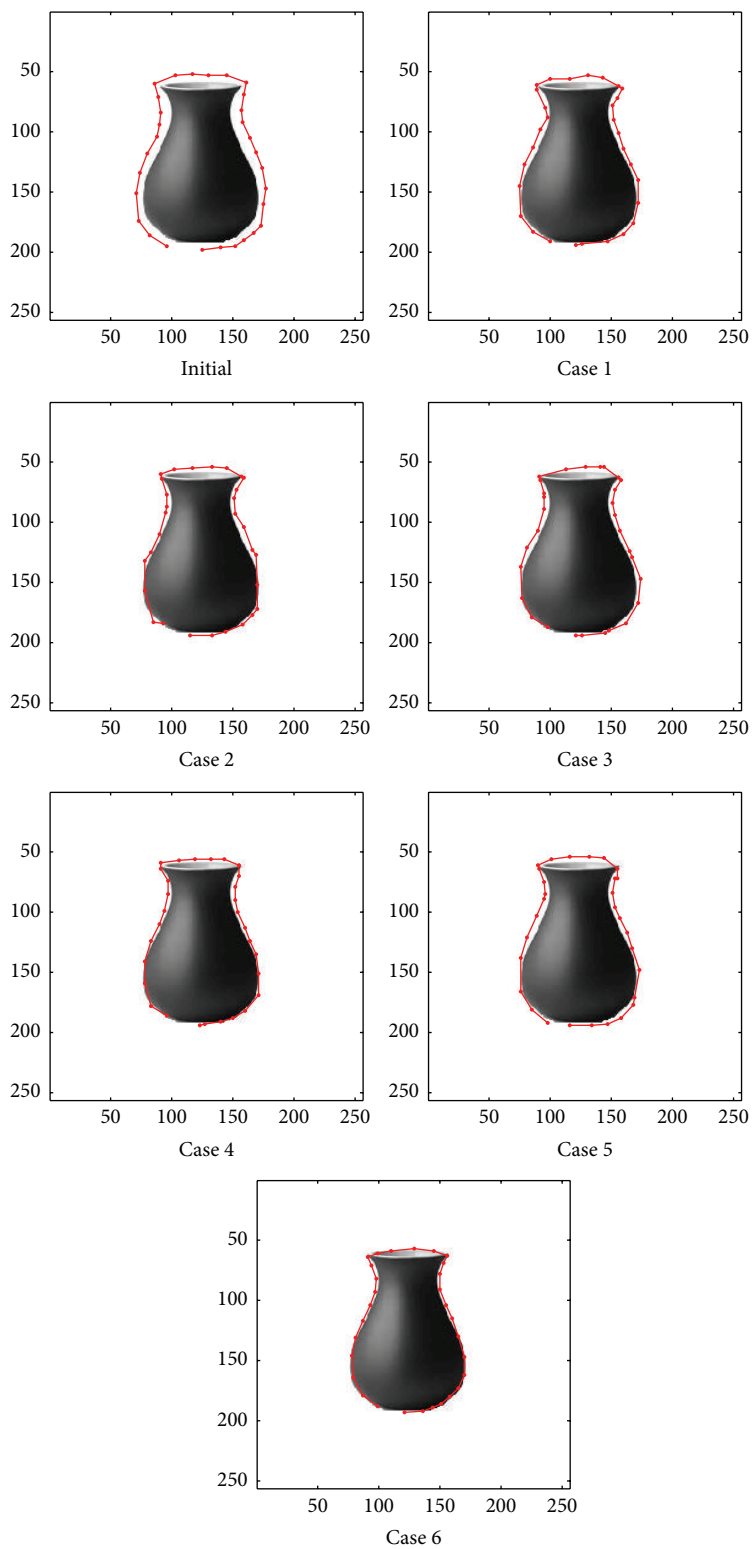


FIGURE 17: Active contour model implementation by “proposed multiswarm PSO” method for different cases of inertia. Results for “Vase” image with control points = 30, $\alpha = 0.321$, $\beta = 0.054$, and $\Gamma = 0.003$.

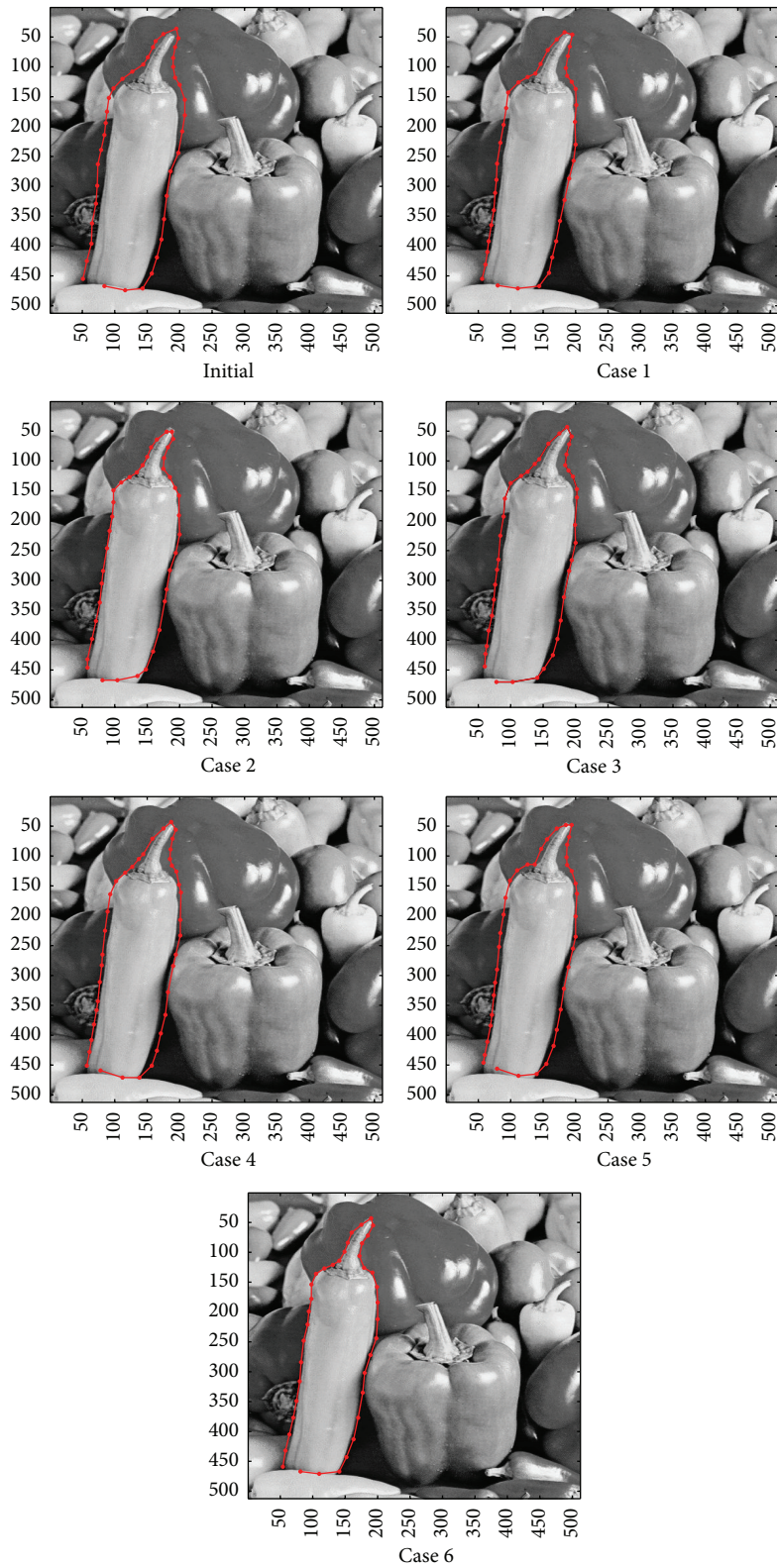


FIGURE 18: Active contour model implementation by “proposed multiswarm PSO” method for different cases of inertia. Results for “Peppers” image with control points = 40, $\alpha = 0.25$, $\beta = 0.22$, and $\Gamma = 0.50$.

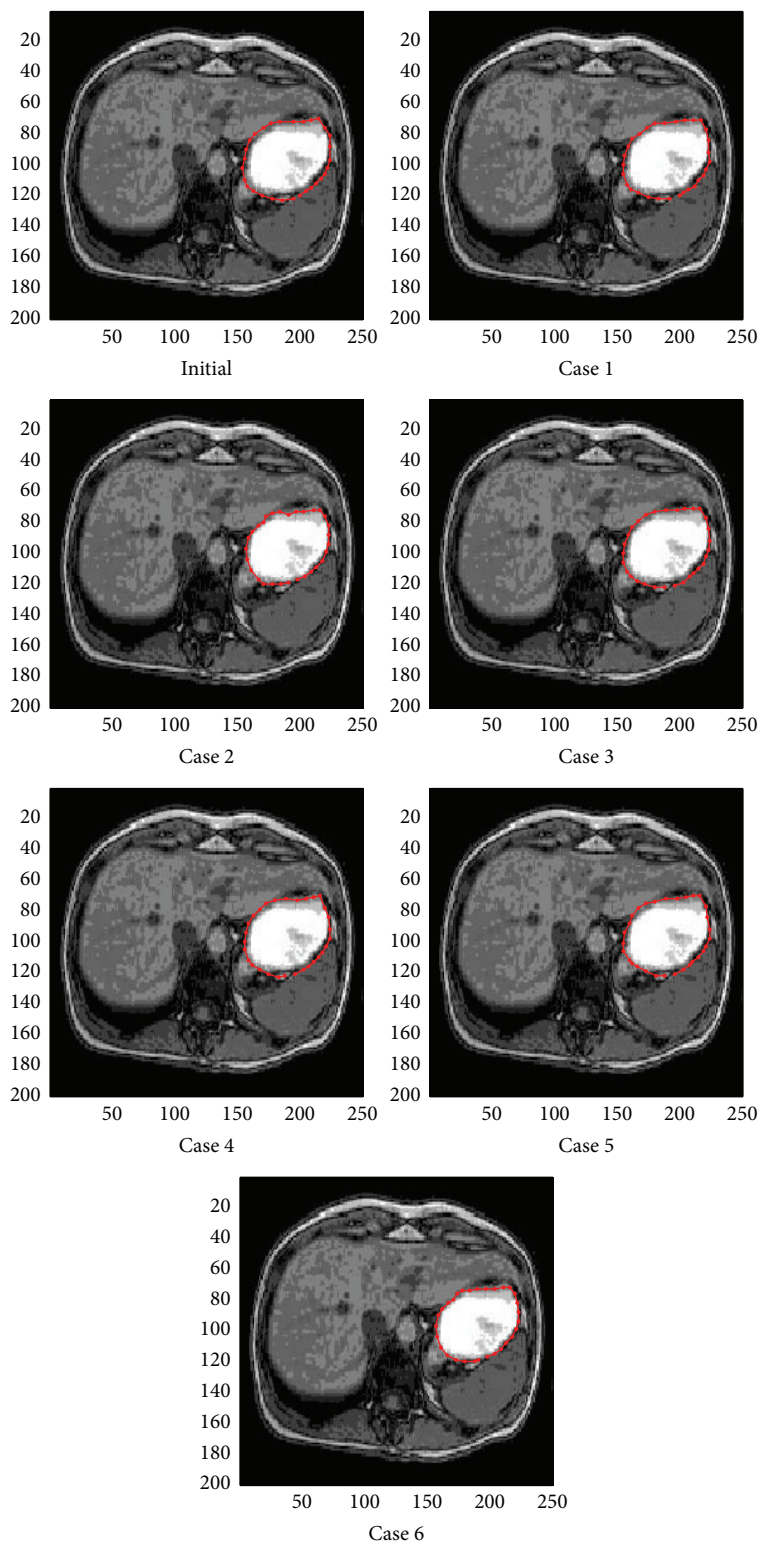


FIGURE 19: Active contour model implementation by “proposed multiswarm PSO” method for different cases of inertia. Results for “MRI” image with control points = 30, $\alpha = 0.17$, $\beta = 0.15$, and $\Gamma = 0.20$.

PSO in which a swarm is set in the two-dimensional image space for every control point in the snake and then all the swarms search for their best positions simultaneously and collaboratively through information sharing among them by keeping all swarms active during the whole process of energy minimization. The proposed multiswarm PSO-based active contour model alleviates the convergence problem encountered in the previously existing multipopulation PSO-based method in two ways: (1) parallel implementation of PSO over multiple swarms, one swarm at every control point, and (2) continuous updating of the control point positions in every generation of the PSO. Results show that our proposed multiswarm PSO-based method is capable of providing minimum contour energy in most of the cases. This verifies efficiency of our proposed method for images having concavities.

Active contour model using multiswarm PSO with fuzzy-rule based inertia adaptation is presented. The proposed method is able to provide better energy minimization than other PSO implementations due to two factors: multiswarm optimization and inertia adaptation. Multiswarm PSO helps in more stable convergence due to the parallel nature of the algorithm. At the same time, inertia updating based on the current state of the algorithm gives a good compromise between global and local search, as necessary. The proposed method is not only able to capture concavities more accurately but also free from manual tuning of the inertia weight. However, the proposed method is computationally more complex. Nevertheless, since this method offers better energy minimization of active contours, it is useful where more precision is required such as in cases of medical image segmentation.

Competing Interests

The authors declare that they have no competing interests.

References

- [1] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.
- [2] D. Williams and M. Shah, "A fast algorithm for active contours," in *Proceedings of the 3rd International Conference on Computer Vision*, pp. 592–595, Osaka, Japan, December 1990.
- [3] L. D. Cohen, "On active contour models and balloons," *CVGIP: Image Understanding*, vol. 53, no. 2, pp. 211–218, 1991.
- [4] M. Wang, J. Evans, L. Hassebrook, and C. Knapp, "A multistage, optimal active contour model," *IEEE Transactions on Image Processing*, vol. 5, no. 11, pp. 1586–1591, 1996.
- [5] A. Yezzi Jr., S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum, "A geometric snake model for segmentation of medical imagery," *IEEE Transactions on Medical Imaging*, vol. 16, no. 2, pp. 199–209, 1997.
- [6] C. Xu and J. L. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 359–369, 1998.
- [7] Y. Y. Wong, P. C. Yuen, and C. S. Tong, "Segmented snake for contour detection," *Pattern Recognition*, vol. 31, no. 11, pp. 1669–1679, 1998.
- [8] H. Park, T. Schoepflin, and Y. Kim, "Active contour model with gradient directional information: directional snake," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 2, pp. 252–256, 2001.
- [9] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.
- [10] J. Park and J. M. Keller, "Snakes on the watershed," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 10, pp. 1201–1205, 2001.
- [11] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Fast geodesic active contours," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1467–1475, 2001.
- [12] J. C. Nascimento and J. S. Marques, "Adaptive snakes using the EM algorithm," *IEEE Transactions on Image Processing*, vol. 14, no. 11, pp. 1678–1686, 2005.
- [13] S.-H. Kim, A. Alattar, and J. W. Jang, "Object contour tracking using the optimization of the number of snake points," in *Proceedings of the International Conference on Computational Intelligence and Security (ICCIAS '06)*, pp. 1591–1596, Guangzhou, China, November 2006.
- [14] Y. V. Venkatesh, S. K. Raja, and N. Ramya, "Multiple contour extraction from graylevel images using an artificial neural network," *IEEE Transactions on Image Processing*, vol. 15, no. 4, pp. 892–899, 2006.
- [15] L. MacEachern and T. Manku, "Genetic algorithms for active contour optimization," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 4, pp. 229–232, Monterey, Calif, USA, May-June 1998.
- [16] A. Mishra, P. K. Dutta, and M. K. Ghosh, "A GA based approach for boundary detection of left ventricle with echocardiographic image sequences," *Image and Vision Computing*, vol. 21, no. 11, pp. 967–976, 2003.
- [17] X.-N. Wang, Y.-J. Feng, and Z.-R. Feng, "Ant colony optimization for image segmentation," in *Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC '05)*, pp. 5355–5360, Guangzhou, China, August 2005.
- [18] D. Wang, Y. Li, and L. Wang, "A new dynamical evolutionary algorithm for active contour optimization," in *Proceedings of the 3rd International Conference on Natural Computation (ICNC '07)*, vol. 5, pp. 571–575, IEEE, Haikou, China, August 2007.
- [19] M.-H. Horng, J.-Y. Chen, and R.-J. Liou, "Honey bee mating optimization scheme for active contour model," in *Proceedings of the 2009 9th International Conference on Hybrid Intelligent Systems (HIS '09)*, vol. 1, pp. 174–179, Shenyang, China, August 2009.
- [20] D. Zeng and Z. Zhou, "Invariant topology snakes driven by particle swarm optimizer," in *Proceedings of the 3rd International Conference on Innovative Computing Information and Control (ICICIC '08)*, pp. 38–42, Dalian, China, June 2008.
- [21] C.-C. Tseng, J.-G. Hsieh, and J.-H. Jeng, "Active contour model via multi-population particle swarm optimization," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5348–5352, 2009.
- [22] E. Shahamatnia and M. M. Ebadzadeh, "Application of particle swarm optimization and snake model hybrid on medical imaging," in *Proceedings of the IEEE 3rd International Workshop on Computational Intelligence in Medical Imaging (CIMI '11)*, pp. 1–8, Paris, France, April 2011.

- [23] J. Novo, J. Santos, and M. G. Penedo, "Topological active models optimization with differential evolution," *Expert Systems with Applications*, vol. 39, no. 15, pp. 12165–12176, 2012.
- [24] W. He, Y. Cheng, L. Xia, and F. Liu, "A new particle swarm optimization-based method for phase unwrapping of MRI data," *Computational and Mathematical Methods in Medicine*, vol. 2012, Article ID 475745, 9 pages, 2012.
- [25] I. Cruz-Aceves, J. G. Avina-Cervantes, J. M. Lopez-Hernandez, and S. E. Gonzalez-Reyna, "Multiple active contours driven by particle swarm optimization for cardiac medical image segmentation," *Computational and Mathematical Methods in Medicine*, vol. 2013, Article ID 132953, 13 pages, 2013.
- [26] H. Xia, Y. Yan, S. Huang, and W. Xiao, "Boundary detection using an improved active contour model without edges," *Journal of Computational Information Systems*, vol. 10, no. 24, pp. 10603–10609, 2014.
- [27] T. Liu, H. Xu, W. Jin, Z. Liu, Y. Zhao, and W. Tian, "Medical image segmentation based on a hybrid region-based active contour model," *Computational and Mathematical Methods in Medicine*, vol. 2014, Article ID 890725, 10 pages, 2014.
- [28] T. Boonnuk, S. Srisuk, and T. Sripramong, "Texture segmentation using active contour model with edge flow vector," *International Journal of Information and Electronics Engineering*, vol. 5, no. 2, pp. 107–111, 2015.
- [29] A. Bilqis and R. Widita, "Comparison of segmentation using fast marching and geodesic active contours methods for bone," *Journal of Physics: Conference Series*, vol. 694, no. 1, Article ID 012044, 2016.
- [30] A. Khunteta and D. Ghosh, "Active contour model via multi-swarm PSO with fuzzy rule-based adaptation of inertia factor," in *Proceedings of the International Conference on Signal Processing and Communication (ICSC '13)*, pp. 191–196, Noida, India, December 2013.
- [31] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks, University of Western Australia*, vol. 4, pp. 1942–1948, Perth, Western Australia, December 1995.
- [32] H. Chen, Y. Zhu, and K. Hu, "Discrete and continuous optimization based on multi-swarm coevolution," *Natural Computing*, vol. 9, no. 3, pp. 659–682, 2010.
- [33] A. Chatterjee and P. Siarry, "Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization," *Computers and Operations Research*, vol. 33, no. 3, pp. 859–871, 2006.
- [34] Y. Shi and R. C. Eberhart, "Fuzzy adaptive particle swarm optimization," in *Proceedings of the Congress on Evolutionary Computation*, vol. 1, pp. 101–106, Seoul, Republic of Korea, May 2001.
- [35] P. Bajpai and S. N. Singh, "Fuzzy adaptive particle swarm optimization for bidding strategy in uniform price spot market," *IEEE Transactions on Power Systems*, vol. 22, no. 4, pp. 2152–2160, 2007.
- [36] T. J. Ross, *Fuzzy Logic with Engineering Applications*, John Wiley & Sons, New York, NY, USA, 3rd edition, 2010.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

