Identify The Optimal Values of the Geometric Deformable Models Parameters to Segment Multiple objects in Digital Images

Abdulmune'm S. Rahma
Department of Computer Science / University of Technology
Israa A. Alwan
Dept. of Computer Sci./College of Education for Pure Sciences(Ibn Al-Haitham)/ University of Baghdad

Received in: 5 September2012 , Accepted in: 20 November2012

Abstract

Accuracy in multiple objects segmentation using geometric deformable models sometimes is not achieved for reasons relating to a number of parameters. In this research, we will study the effect of changing the parameters values on the work of the geometric deformable model and define their efficient values, as well as finding out the relations that link these parameters with each other, by depending on different case studies including multiple objects different in *spacing, colors, and illumination.* For specific ranges of parameters values the segmentation results are found good, where the success of the work of geometric deformable models has been limited within certain limits to the values of these parameters.

Keywords

Multiple objects segmentation, geometric deformable models, level set method.

Introduction

Multiple objects segmentation in digital images is one of the important trends in scientific research because of the effect in the studies that adopt this topic, including digital images to track targets in automatic control systems for the security of buildings, where the item (the person) is tentatively selected to then identify the identity, digital images on the radar to identify objects for tracking and positioning, digital images in medical devices, and robots [1, 2]. The first phase for all these applications is limited for detecting and segmenting targets(desired objects) and the objective of this research will focus on that first common phase, where the geometric deformable models are considered. They are curves or surfaces that move within two-dimensional (2D) or three-dimensional (3D) digital images under the influence of both internal and external forces and user defined constraints [3, 4]. Generally, geometric deformable models have very useful properties in segmentation process, but with respect to the multiple objects segmentation the most important one is the "topology independent". This property means, that the geometric deformable models do not require any spatial strategy to handle multiple objects, where the evolving curves naturally split and merge allowing the simultaneous detection of several objects so the number of objects to be segmented are not required to be known prior in the image.

Geometric deformable models are independently introduced by Caselles *et al.* [5] and Malladi *et al.* [6], respectively. These models are based on *front evolution theory and the level set method* [3, 7, 8, 9]. The geometric deformable models have two main problems. *The first one* is the *re-initialization* technique of the traditional level set methods, which is used for periodically re-initializing the level set function to a signed distance function during the evolution. It is a complex and expensive procedure as it should be repeated every iteration to avoid errors [7, 8]. Fortunately, Li *et al* [8] solved this problem by proposing a new variational formulation that forces the level set function to be close to singed distance function. The new variational formulation consists of an internal energy term that penalizes the deviation of the level set function from a signed distance function, and an external energy term that drives the motion of the zero level set toward the boundaries of the desired object. Therefore completely eliminates the need of the costly re-initialization procedure. *The second* is the success of the segmentation process is not direct, and sometimes it fails. The reason for this, is that the geometric deformable models depend on a number of parameters and the appropriate or efficient values for these parameters are not given according to a fixed base, but by trial and error. So, in this research we will study the effect of changing the parameters values on the work of the geometric deformable model and define their efficient values, as well as finding out the relations that link these parameters with each other when using it in multiple objects segmentation .

Background on Traditional Level Set Methods

The level set method was introduced by Osher and Sethian [10]. It is used in geometric deformable models representation, where, instead actually parameterizing the front *S (t)*, represent it as the zero level set of a higher dimensional function ϕ, and calculate the evolution of this function as an initial value problem. This can be modeled as [7]:

$$\frac{\partial \phi}{\partial t} + F\left|\nabla \phi\right| = 0 \quad \text{............................(1)}$$

Given $\phi(s, t=0)$. At any instant, the position of front *S (t)* shall be given as the zero level set of the evolving function ϕ:

$$S(t) = \{(x, y) \in R^2 \mid \phi(x, y, t) = 0\} \quad \text{.......... (2)}$$

The speed scalar functions *F* depending on:

- The front local curvature.

- The image gradient.
- Additional propagation terms

The basic problem in traditional level set methods is, although F is a real, non-uniform function, defined only over F. The algorithm doesn't see it that way, considering F valid over the entire domain. It then updates all level sets of ϕ according to F, causing ϕ to be no longer a signed distance function where $|\nabla\phi|=1$. To avoid these problems, a common numerical scheme is to initialize the function ϕ as a signed distance function before the evolution, and then "reshape" (or "re-initialize") the function ϕ to be a signed distance function *periodically* during the evolution [7, 8]. This would require solving:

$$\frac{\partial\phi}{\partial t}=\operatorname{sign}(\phi_0)\left(1-|\nabla\phi|\right) \qquad \text{.......... (3)}$$

where ϕ_0 is the function to be re-initialized, and sign (ϕ) is the sign function.

Unfortunately, the above equation has drawbacks. If ϕ_0 is not smooth or ϕ_0 is much steeper on one side of the interface than the other, the zero level set of the resulting function ϕ can be moved incorrectly from that of the original function. Moreover, when the level set function is far away from a signed distance function, these methods may not be able to re-initialize the level set function to a signed distance function. In practice, the evolving level set function can deviate greatly from its value as signed distance in a small number of iteration steps, especially when the time step is not chosen small enough [8].

So, from the practical viewpoints, the re-initialization process can be quite complicated, expensive, and have subtle side effects. Moreover, most of the level set methods are fraught with their own problems, such as when and how to re-initialize the level set function to a signed distance function. Because of these shortcomings Li *et al* [8] introduced a new variational level set formulation without re-initialization.

Modified Geometric Deformable Model (MGDM) Based on Level Set Method without Re-initialization

The total functional energy proposed by Li *et al* [8] is:

$$E(\phi)=\mu\, p(\phi)+E_m\ (\phi) \qquad \text{.......... (4)}$$

The left term is the Penalizing function or the internal energy, where

$$p(\phi)=\int_\Omega \frac{1}{2}\left(|\nabla\phi|-1\right)^2 dxdy \qquad \text{.......... (5)}$$

It is a metric to characterize how close a function ϕ to a signed distance function in $\Omega\subset R^2$. This metric will play a key role in the new variational level set formulation. μ is a parameter controlling the effect of penalizing the deviation of ϕ from a signed distance function, it must be $\mu>0$. The right term is the external energy that would drive the motion of the zero level curve of ϕ. The external energy can be calculated as the following:

Let I be an image, and g be the edge indicator function defined by

$$g=\frac{1}{1+|\nabla G_\sigma * I|^2}$$

Where G_σ is the Gaussian kernel with standard deviation σ. The external energy for a function $\phi(x, y)$ is defined as below

$$E_{g,\lambda,v}(\phi)=\lambda L_g(\phi)+v A_g(\phi) \qquad \text{.......... (6)}$$

where λ > 0 and v can be positive or negative, depending on the relative position of the initial curve to the desired object external or internal respectively, The energy functional $L_g(\phi)$ is responsible for computing the *Length* of the zero level curve of ϕ and the energy functional $A_g(\phi)$ also called the weighted *Area* term is responsible for speeding up the curve evolution. The terms $L_g(\phi)$ and $A_g(\phi)$ are defined by

$$L_g(\phi) = \int_\Omega g\,\delta(\phi)|\nabla\phi|\,dxdy$$

and

$$A_g(\phi) = \int_\Omega g\,H(-\phi)\,dxdy \qquad \text{.......... (8)}$$

where δ is the univariate Dirac function, and H is the Heaviside function.

After computing the external energy, the total energy equation (4) can be rewritten as below

$$E(\phi)=\mu\, p(\phi)+E_{g,\lambda,v}(\phi) \qquad \text{.......... (9)}$$

Equation (9) must be minimized to find the solution. The gradient flow th is as below

$$\frac{\partial\phi}{\partial t}= \mu[\Delta\phi-\operatorname{div}(\frac{\nabla\phi}{|\nabla\phi|})]+\lambda\delta(\phi)\operatorname{div}(g\frac{\nabla\phi}{|\nabla\phi|})+v\,g\,\delta(\phi) \qquad \text{.......... (10)}$$

where Δ is the Laplacian operator. Equation (10) is the evolution equation of the level set function.

As for the proposed initialization process of the level set function, in this new formulation the level set function ϕ is no longer required to be initialized as a signed distance function, where it can be defined as the following:

$$\phi_0(x,y) = \begin{cases} -P, & (x,y)\in\ \Omega_0-\partial\Omega_0 \\ 0 & (x,y)\in\partial\Omega_0 \\ P & \Omega\ -\Omega_0 \end{cases} \qquad \text{.......... (11)}$$

Where Ω_0 is a subset in the image domain Ω , and $\partial\Omega_0$ be all the points on the boundaries of Ω_0 [8].

The Diagram of the MGDM

To start the segmentation process, an initial front must put around the objects to be segmented, and then the MGDM parameters must be inputted. Depending on values of these parameters, the given front will evolve and move towards the objects boundaries and then locks on them. Figure (1) shows the overview diagram of the MGDM.

Implementation of the MGDM

The MGDM depends on *eight* important parameters which are, the internal (penalizing) energy term parameter μ,the scale parameter in Gaussian kernel for smoothing (needed in edge indicator or stopping function) σ, the weighted length term parameter λ, the weighted area term parameter v, the smoothed Dirac function parameter ε, the initial level set function definition parameter p, the time step of iteration parameter T, and the evolution iteration number parameter. For studying the effect of changing these parameters values on the work of the MGDM and finding out their efficient values, the MGDM was tested on multiple objects with different cases of *spacing, colors, and illumination*; as clarified in the following sections. The value of ε in all experiments of this research was **1.5**, so, it isn't mentioned in its tables.

Contiguous Objects Different in Spaces

The MGDM is applied on objects in three different cases of spacing (far from, close, and very close) to each other, as in the next sections.

1. Objects Far From and Close to Each Other

The MGDM is applied on image of three weights are far from each other. When giving values for the parameters as in Table (1), we note that the MGDM has succeeded in segmenting these weights and did not face any difficulty, despite the large distances between them, see Figure (2). When using the same parameter values shown in Table (1) in segmenting weights are close to each other as in Figure (3), we note the success of the MGDM, although the distance between the weights is smaller than in Figure (2).

2. Objects Very Close to Each Other

The MGDM is applied on image of three weights which are very close to each other, and this means that there are long and narrow distances between them, and when using the same parameter values as shown in Table (1), the MGDM fails miserably in the process of segmenting, even when increasing the evolution iteration number to 400, see Figure (4). When trying to solve this problem by making the initial front permeates the distances between weights as in Figure (5, a), to facilitate the work of the MGDM in pulling the front to the inside, also note the failure of the MGDM in spite of the value of evolution iteration number was 400, see Figure (5, b, c).

So, overcoming this problem is not by increasing the value of evolution iteration number or change the shape of the initial front, but to make it pull strongly and quickly into those long, narrow distances, and this is possible when giving large values to σ, T, and v. When giving large values for these parameters as in Table (2), we note that the initial front pulled to the inside, but after a number of iterations gets shrink and then vanished, see Figure (6).

The cause of the failure of the proposed solution in Figure (6) is the length of the front disproportionate with its speed of evolution, where, the length of the front was very small in relation to the speed of evolution, where λ =3 while v =9. So the proposed solution becomes successful after giving a very large value to the λ as well. In order to fit the length of the front with its speed of evolution, the value of λ must be equal to the value of v. So, after regarding the values in Table (3), we note in Figure (7) that the MGDM has succeeded in segmenting the weights through 250 iterations, although the long and narrow distances between them, this number is much less when we try segmenting weights by making evolution iteration number = 400 However, the MGDM failed. Also, the initial front is regular (don't permeate the distances between weights). Values of parameters are shown in the Table (3) except evolution iteration number =200, were used in segmenting another image of objects very close to each other, also note the direct success of the MGDM and this confirms the correctness of the proposed values, see Figure (8). When comparing the successful values shown in Table (3) with the successful values shown in Table (1), we will find the following:

μ (with objects very close to each other) < μ (with objects far and close to each other)
σ (with objects very close to each other) > σ (with objects far and close to each other)
λ (with objects very close to each other) = λ (with objects far and close to each other) *3
v (with objects very close to each other) = v (with objects far and close to each other) *3
p (with objects very close to each other) = p (with objects far and close to each other)
T (with objects very close to each other) = T (with objects far and close to each other) + (3*10)

Overlapped Objects Different in Colors

The MGDM is applied on overlapped objects in two cases of colorific difference. The first case is overlapped objects with simple colorific difference. The second is overlapped objects with significant colorific difference as in the next sections.

1. Overlapped Objects with Simple Colorific Difference

The MGDM is applied on image of five circular objects located inside three combined circles with simple colorific difference, as in Figure (9). When focusing on Figure (9, a) note two things are very important in the success of the MGDM, *the first is,* the five circular desired objects are located in the center of the image, and this means that, they are very far from the initial front. *The second* is, the boundaries of the circles surrounding the five desired objects are clear, but not very strong. These two things mean, should increase the capture range of the MGDM in order to gain access to those five objects and segmenting them. This is done by giving σ large value such as 1.5, in order to smooth the undesired boundaries (combined circles) and get rid of them. As for the rest of the parameters must be of fair values, as in the Table (4). Figure (9, b, c) shows the success of the MGDM in segmenting those objects. Values of parameters are shown in the Table (4) except evolution iteration number =450, were used in segmenting another image of overlapped objects with simple colorific difference, also note the direct success of the MGDM and this confirms the correctness of the proposed values, especially σ, see Figure (10).

2. Overlapped Objects with Significant Colorific Difference

When applying what we have in previous section, on image of objects with significant colorific difference, the MGDM fails in segmentation process, see Figure (11). Also, there are five circular objects located inside three combined circles with significant colorific difference. The MGDM was able to pass the boundaries of the outer circle and locked on the boundaries of the middle circle and did not access the five desired objects, despite the evolution iteration number = 600. The values of the rest parameters are shown in Table (5).

Clear from the above, when a colorific difference between objects is significant, increasing the capture range is not enough for the success of the MGDM, where the strong boundaries of the outer circles still hamper the work of the MGDM in spite of smoothing them. So, to solve this problem, in addition to increasing the capture range of the MGDM, we should make it able to jump rapidly to overcome those strong boundaries; in other words (the front should be shrank over the strong boundaries of the outer circle only). This is done by giving so very large values for both T and v with attention to increase the value of **λ** (but not so very large value) to avoid the problem of the front vanishing, see Figure (12). The proposed successful values for the parameters are shown in Table (6). Values of parameters are shown in Table (6) except evolution iteration number =220, were used in segmenting another image of overlapped objects with significant colorific difference, also note the direct success of the MGDM and this confirms the correctness of the proposed values, see Figure (13). When comparing the successful values shown in Table (6) with the successful values shown in Table (4), we will find the following:

μ (with overlapped objects with significant colorific difference) < μ (with overlapped objects with simple colorific difference)

σ (with overlapped objects with significant colorific difference) = σ (with overlapped objects with simple colorific difference)

λ (with overlapped objects with significant colorific difference) = λ (with overlapped objects with simple colorific difference) *2

v (with overlapped objects with significant colorific difference) = v (overlapped objects with simple colorific difference) *4 *p* (with overlapped objects with significant colorific difference) = *p* (with overlapped objects with simple colorific difference)

T (with overlapped objects with significant colorific difference) = *T* (with overlapped objects with simple colorific difference) *2

Approach the Objects' Color with the Background Color with Progressing in Illumination

The MGDM is applied on objects with a color similar to the background color in two different cases, the first case, objects with a color similar to the background color to a large extent, with different degrees of illumination (high, medium, low). The second case, objects with a color similar to the background color but not quite, also with different degrees of illumination (high, medium, low), as in the following sections.

1. Objects with a Color Similar to the Background Color to a Large Extent

When applying the MGDM on objects with a color similar to the background color to a large extent, it is normal that the boundaries of the objects are very weak *whatever the difference in the degree of illumination*, and the first thing we think is to reduce the smoothing of the objects boundaries as well as slow down the speed of evolution of the front significantly, to enable the MGDM to detect the boundaries of the desired objects and locked on it instead of rush to within the boundaries of objects. This is done by giving σ, *T*, and *v* small values respectively. This is what has applied on three images of four square objects with different degrees of illumination (high, medium, and low) as depicted in Figure (14), Figure (15), and Figure (16) respectively, where the square objects with *light yellow color* and the background also with *light yellow color*. In Figure (14), we note the MGDM was able to segment the four objects, but the segmentation results were not promising, and the parameters values as in Table (7). In Figure (15) and Figure (16) we note the MGDM failed in segmentation process, where the front rushed to within the boundaries of the objects. When trying to make the values of σ, *T*, and *v* less than in Table (7), the segmentation results become worse.

2. Objects with a Color Similar to the Background Color But Not Quite

When applying the MGDM on objects with a color similar to the background color but not quite, also with different degrees of illumination (high, medium, low), the MGDM succeeded in segmentation process, but the parameters σ, *T*, and *v* have special conditions for each degree of illumination. In Figure (17), the MGDM has succeeded in segmenting three lemons with *dark yellow color* and background with *light yellow color*. because of the intensity of illumination; the boundaries of lemons are somewhat weak. As we mentioned before, the values of the σ, *T*, and *v* must be less than what can be in order to reduce the smoothing of the boundaries and slow the speed of the front evolution, so that the MGDM can detect the boundaries and lock on it. The parameters values are shown in Table (8), where the values of σ and *v* stay as in Table (7), But the value of *T* equals to twice of the value of *T* in Table (7). This is because the boundaries of lemons in Figure (17) are weak, but stronger than the boundaries of the square objects in Figure (14), Figure (15), and Figure (16). So that, when the value of *T* is twice, the front will not rush to within the boundaries of the three lemons. But in the Figure (18) and Figure (19), the degrees of illumination are medium and low, respectively. This means that there are shadows in the images and this makes the work of the MGDM difficult. Because the boundaries are weak, the values of σ, *T*, and *v* should be less than what can be**;** and because of the presence of shadows, the value of *T* should be larger than what can be; so that the MGDM would be able to pass the areas of shadows, instead of suspicion and considered them the desired objects. In this case, in order the MGDM would be able to segment the lemons correctly, the value of σ and *v* stay small as in the case of high illumination, and the value of *T* would be very large as well as *p*. But in return the shape of the initial front must be irregular as in Figure (18, a) and Figure (19, a), where it is close to the boundaries of objects in the areas of shadows and far a little the boundaries of objects in areas that does not have the shadows. So that the speed of the front, which starts to pass the areas of shadows does not lead to a rush of the front to within the boundaries of objects in areas where the boundaries are weak but do not have the shadows. The parameters values of medium and low illumination experiments are shown in Table (9). When comparing the successful values shown in Table (9) with the successful values shown in Table (8), we will find the following:

μ (with medium and low illumination) < μ (with high illumination)
σ (with medium and low illumination) = σ (with high illumination)
λ (with medium and low illumination) = λ (with high illumination)
v (with medium and low illumination) = v (with high illumination)
p (with medium and low illumination) > p (with high illumination)*2
T (with medium and low illumination) > T (with high illumination)*2

Discussion of Experimental Results of MGDM Implementation

When studying the tables of each case of the cases mentioned in previous sections we will find that there are parameters whose values depend on the quality of boundaries and the status or the arrangement of objects in the image. And that there are parameters whose values depend on the status of objects in the image only. Also find that there are some relations linking the parameters with each other and are as follows:

- Values of σ depend on the quality of boundaries and the status of objects in the image. So, when the objects boundaries are strong the optimal value of σ is (1 to 1.5). When the objects boundaries are weak, the optimal value of σ is ($0<\sigma<=0.5$). If the objects status is complex the optimal value of σ is (1.5) to increase the MGDM capture range.
- Values of v depend on the quality of boundaries and the status of objects in the image. So, when the objects boundaries are strong, the optimal value of v is (3). When the objects boundaries are weak, the optimal value of v is ($v<=3$). If the objects status is complex, the optimal value of v is (9 to 12). Also, because of the site of the initial front must be outside the desired objects, the values of v must be always positive.
- Values of T depend on the quality of boundaries and the status of objects in the image. So, when the objects boundaries are strong the optimal value of T is (50 to 60). When the objects boundaries are weak the optimal value of T is ($T<=40$). If the objects status is complex the optimal value of T is (90 to 100).
- Values of μ depend on the quality of boundaries and the status of objects in the image *but indirectly*, where there is an *inverse relationship* links μ with T. Where the cases those require to be a large value of T, in turn the value of μ should be few and vice versa. So, when the objects boundaries are strong, the optimal value of μ is (0.003 to 0.004). When the objects boundaries are very weak the optimal value of μ is (0.01); *this means that, μ differs from the rest parameters, where their values are large when the boundaries are strong and are few values when the boundaries are weak*. If the objects status is complex, the optimal value of μ is (0.002).
- Values of λ depend on the status of objects in the image; being close, far, overlapped, or surrounded by other objects. If the objects status is not complex as if they are spaced from each other, the optimal value of λ is (3). If the objects status is complex as if they are overlapped with other objects or there are long and narrow distances between them, the optimal value of λ is (6 to 9) could also be argued, that the value of λ also depends on the number of objects in the image. If there are many objects in the image, the value of λ must be very large (greater than 9, such as 12), so that the front would be able to

cover the boundaries of these objects. Also, there is a *relationship of equality* that links λ with v; where the length of the front must be equal to its speed of evolution ($\lambda = v$), Otherwise, the front will shrink and then vanish. One case deviated this relationship, a case of overlapped objects with significant colorific difference where ($\lambda = 1/2v$); where the front shrinking was necessary to avoid the strong boundaries of the outer circles.

- Values of evolution iteration No. depend on the status of objects in the image could also be argued, that it also depends on the values of the other parameters, especially σ, v, *and* T. When the values of those parameters are large, the capture range of the MGDM will be large and therefore the optimal values of evolution iteration No. must be few or fair. When the values of those parameters are few, the capture range of the MGDM will be small and therefore the optimal values of evolution iteration No. must be very large in order, the MGDM get a wide range to search for the desired objects boundaries.
- As for p, its values depend on the presence of local minima in the image or not (for example, the presence of shadows). If the image is free from a local minima, the optimal value of p is (6), and the form of the initial front is regular and there is no importance whether the initial front, close or far from the desired objects. But when there is a local minima in the image, the optimal value of p must be very large, where it is (16), and the form of the initial front must be irregular, because the initial front should be close to the boundaries of desired objects in areas where there is a local minima and far from the boundaries of objects in areas that does not have a local minima.

Conclusions

This research focused on studying the effect of changing the eight parameters values on the work of the MGDM and identifies their efficient values, as well as finding out the relations that link these parameters with each other when using it in multiple objects segmentation. It is concluded that, in each case (image) of the cases processed by the MGDM, there are parameters that have a major role in the success of the MGDM and other parameters have a minor role, but this role is different from case to case, for example, in a particular case, a number of parameters have a key role in the success of the MGDM, but in another case have a minor role. It is also concluded, that the image analyzing and understanding by the user is a basic step in determining two things: *first*, the major parameters and the minor parameters that affect the success of the MGDM in segmenting the current image. *Second*, the appropriate values for these eight parameters.

It is also found that the most parameters depend primarily on the quality of the boundaries, what makes the MGDM extremely dependent on the image quality that is lowly noised, fair definition of structures edges and absence of local minima. Therefore, the performance of the MGDM was not promising when segmenting objects with a color similar to the background color to a large extent. But the failure of the MGDM in a part of one case does not degrade its importance because there is no ideal segmentation method. Also, when comparing the MGDM with the known methods used in multiple objects segmentation, we will find that it is very flexible and can adapt and exactly converge to the boundaries of desired objects (no some of the background) regardless the number of objects, the distance between each other, and overlapped or not.

References

1. Srinivasan, G. N. and Shobha, G. (2008) ,Segmentation Techniques for Automatic Targets Detection and Recognition, International Journal of Computers, 2(2).
2. Zuva, T.; Olugbara, O.; Ojo, S.; and Ngwira, S. (2011) ,Image Segmentation, Available Techniques, Developments and Open Issues, Canadian Journal on Image Processing and Computer Vision, 2(3).
3. Jayadevappa, D.; Srinivas, Kumar S.; and Murty, DS. (2011),Medical Image Segmentation Algorithms using Deformable Models: A Review, 28(3): 248-255.
4. Lee, M. (2011) ,Geometric Active Model for Lesion Segmentation on Breast Ultrasound Images, Proceedings of the IEEE 11th International Conference on Computer and Information Technology (CIT).
5. Caselles, V.; Catte, F.; Coll, T.; and Dibos, F. (1993),A Geometric Model for Active Contours in Image Processin,Numer. Math., 66: 1-31.
6. Malladi, R.; Sethian, J. A.; and Vemuri, B. C. (1995) Shape Modeling with Front Propagation: A level Set Approach, IEEE Trans. Patt. Anal. Mach. Intell.,17: 158-175.
7. Derraz, F.; Beladgham, M.; and Khelif, M. (2004) ,Application of Active Contour Models in Medical Image Segmentation, IEEE Trans. On Information Technology: Coding and Computing, Department of Computer Science, Abou Bekr Belkaid University, Tlemcen.
8. Li, C.; Xu, C.; Gui, C.; and Fox, M. D. (2005) , Level Set Evolution without Re-initialization: A New Variational Formulation", Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition.
9. He, L.; Peng, Z.; Everding, B.; Wang, X.; Han, C. Y.; Weiss, K. L.; and Wee, W. G. (2008) ,A Comparative Study of Deformable Contour Methods on Medical Image Segmentation, Image and Vision Computing, 26:141-163.
10. Osher, S. and Sethian, J. A. (1988) " Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations, J. Comp. Phys., 79: 12-49.

Table (1): Parameters values of segmentation process of three weights far from and close to each other

(μ)	(σ)	(λ)	(v)	(p)	(T)	Iterations No.
0.003	1	3	3	6	60	200

Table (2): Parameters values of fail segmentation process of the three weights very close to each other

(μ)	(σ)	(λ)	(v)	(p)	(T)	Iterations No.
0.002	1.5	3	9	6	90	250

Table (3): Successful parameters values of segmentation process of three weights very close to each other.

(μ)	(σ)	(λ)	(v)	(p)	(T)	Iterations No.
0.002	1.5	9	9	6	90	250

Table(4):Parameters values of segmentation process of five circular objects located inside three combined circles with simple colorific difference.

(μ)	(σ)	(λ)	(v)	(p)	(T)	Iterations No.
0.004	1.5	3	3	6	50	300

Table (5) :Parameters values of fail segmentation process of five circular objects located inside three combined circles with significant colorific difference

(μ)	(σ)	(λ)	(v)	(p)	(T)	Iterations No.
0.003	1.5	3	3	6	60	600

Table (6): Successful parameters values of segmentation process of five circular objects located inside three combined circles with significant colorific difference

(μ)	(σ)	(λ)	(v)	(p)	(T)	Iterations No.
0.002	1.5	6	12	6	100	200

Table (7) Parameters values of segmentation process of four square objects with a color similar to the background color to a large extent with high, medium, and low illumination

(μ)	(σ)	(λ)	(v)	(p)	(T)	Iterations No.
0.01	0.5	3	3	6	20	150

Table (8): Parameters values of segmentation process of three lemons with high illumination

(μ)	(σ)	(λ)	(v)	(p)	(T)	Iterations No.
0.005	0.5	3	3	6	40	200

Table (9): Parameters values of Segmentation process of three lemons With medium and low illumination

(μ)	(σ)	(λ)	(v)	(p)	(T)	Iterations No.
0.002	0.5	3	3	16	90	240

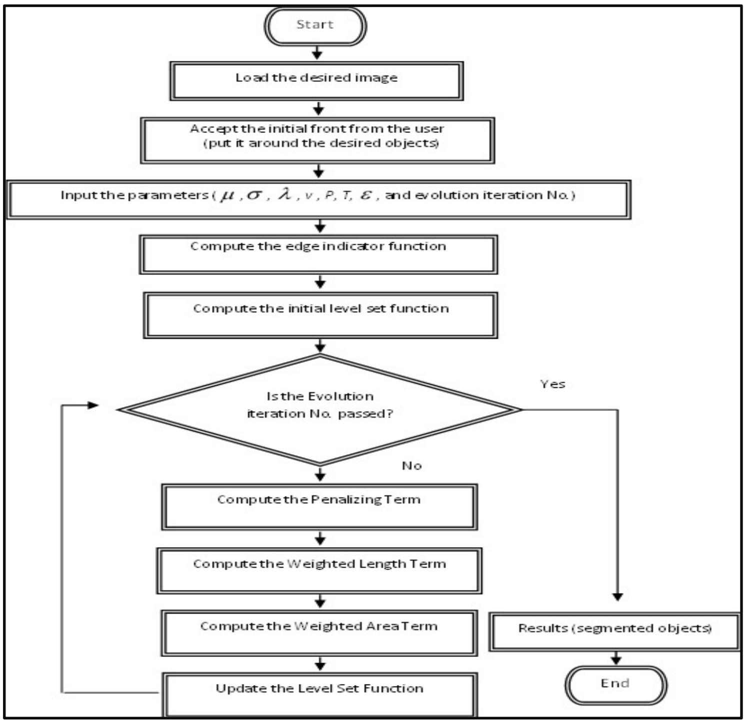
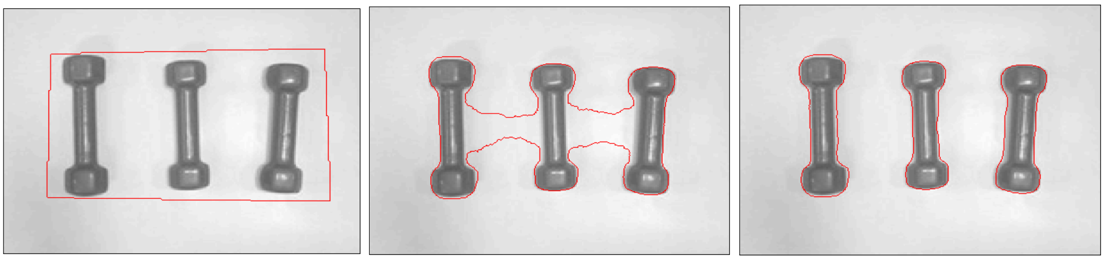
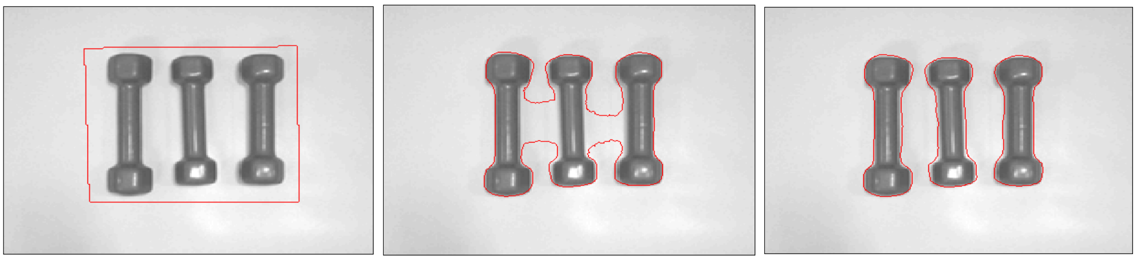


Fig. (1): Overview diagram of the modified geometric deformable model



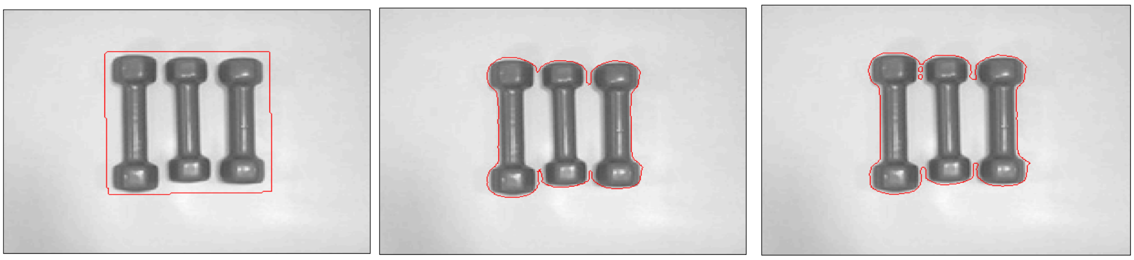
(a) (b) (c)

Fig. (2): Segmenting three weights far from each other, (a) input image with the initial front, (b) initial front evolution, (c) segmentation results.



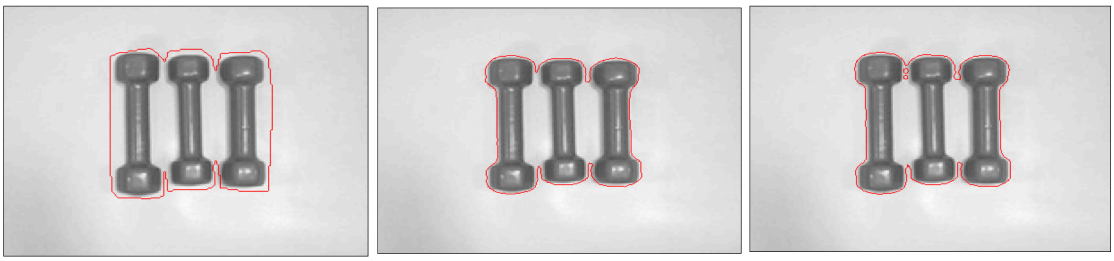
(a) (b) (c)

Fig. (3): Segmenting three weights close to each other, (a) input image with the initial front, (b) initial front evolution, (c) segmentation results.



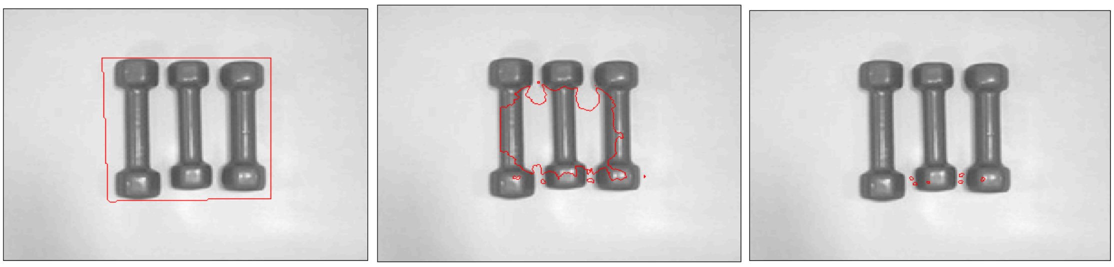
(a) (b) (c)

Fig. (4): Segmenting three weights very close to each other, (a) input image with the initial front, (b) initial front evolution, (c) wrong segmentation results.



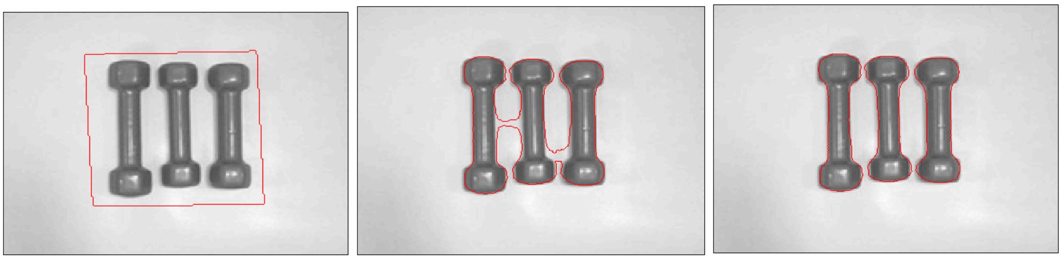
(a) (b) (c)

Fig. (5): Segmenting three weights very close to each other, (a) input image with the initial front permeates the distance between objects, (b) initial front evolution, (c) wrong segmentation results



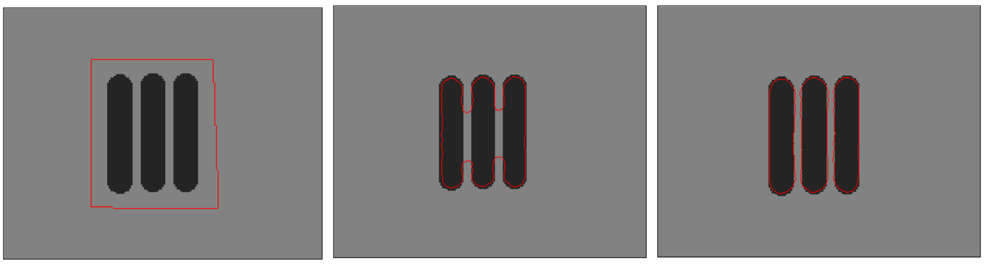
(a) (b) (c)

Fig. (6): Segmenting three weights very close to each other, (a) input image with the initial front, (b) initial front shrank after 120 iterations, (c) initial front vanished after 250 iterations



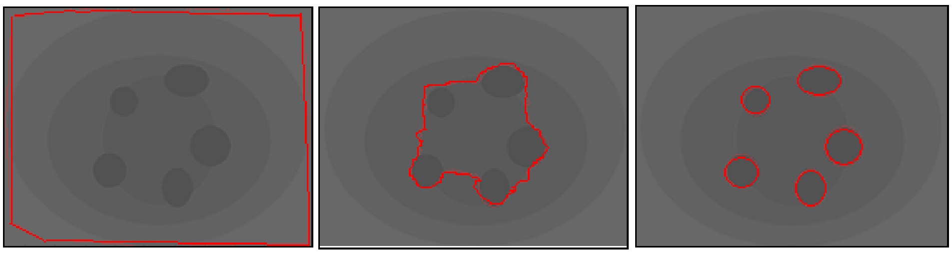
(a) (b) (c)

Fig. (7): Segmenting three weights very close to each other, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result.



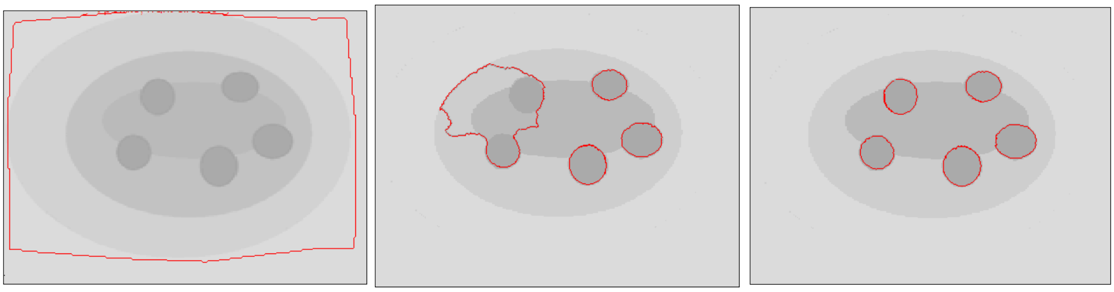
(a) (b) (c)

Fig. (8): Segmenting three objects very close to each other, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result



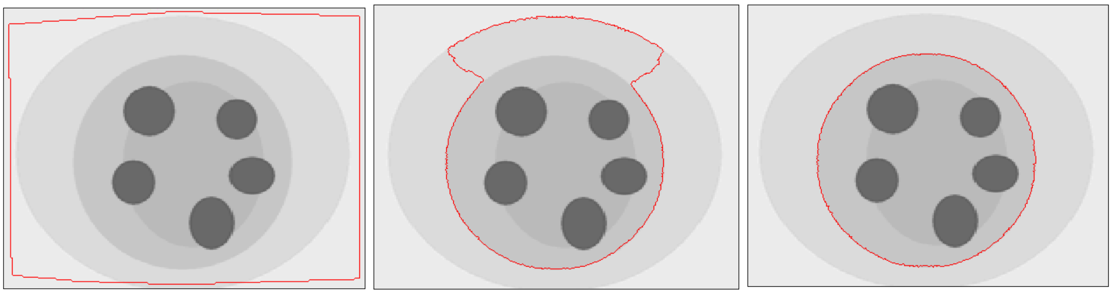
(a) (b) (c)

Fig. (9): Segmenting five circular objects located inside three combined circles with simple colorific difference, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result.



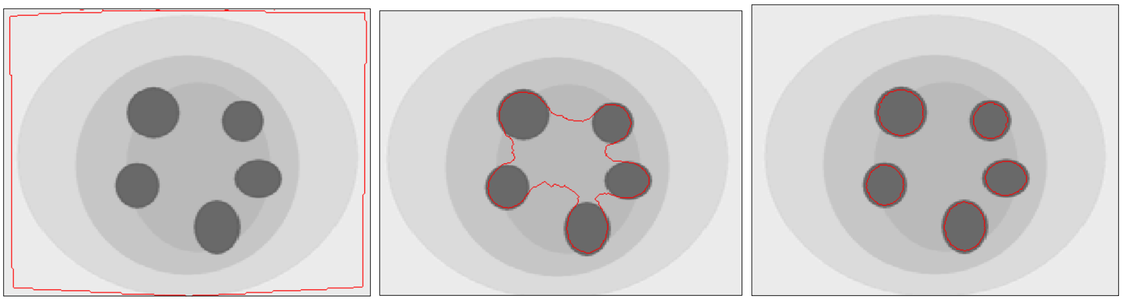
(a) (b) (c)

Fig. (10): Segmenting five circular objects located inside three combined ellipses with simple colorific difference, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result.



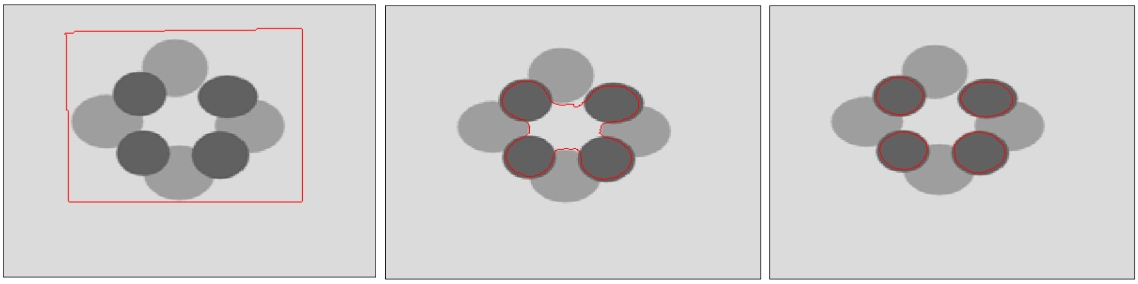
(a) (b) (c)

Fig. (11): Fail segmentation process of five circular objects located inside three combined circles with significant colorific difference, (a) input image with the initial front, (b) initial front evolution after 380 iterations, (c) wrong final segmentation result after 600 iterations.



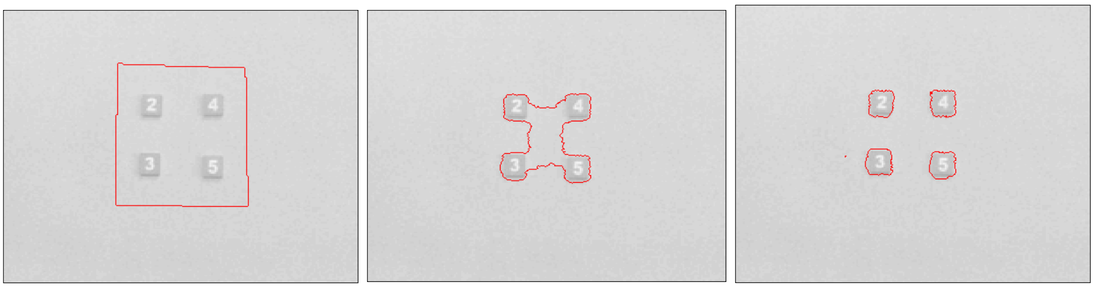
(a) (b) (c)

Fig. (12): Segmenting five circular objects located inside three combined circles with significant colorific difference, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result.



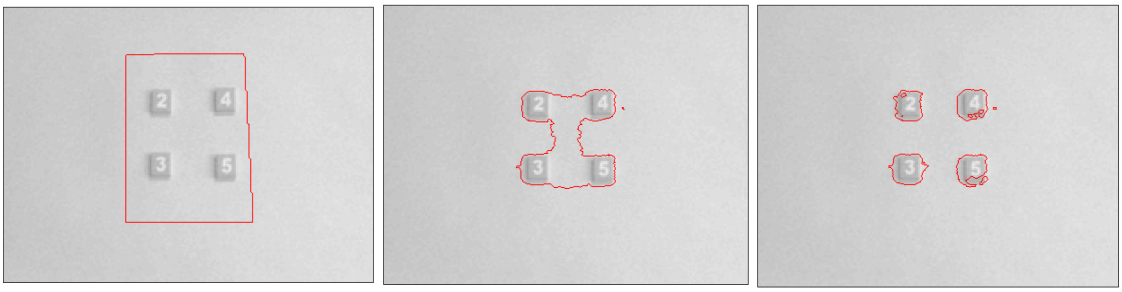
(a) (b) (c)

Fig. (13): Segmentation process of overlapped circles with significant colorific difference, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result.



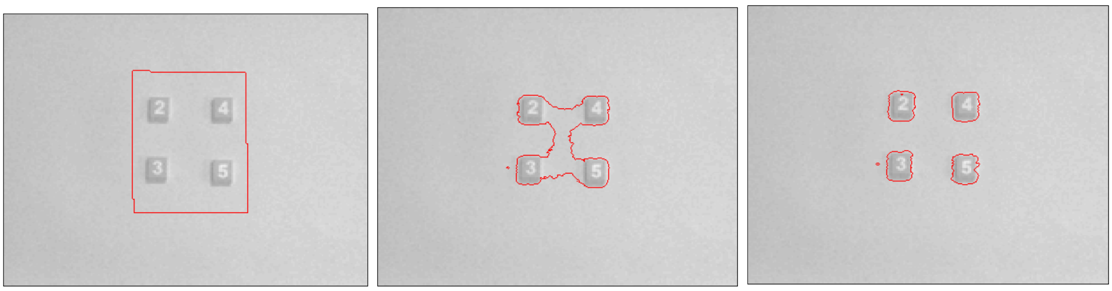
(a) (b) (c)

Fig. (14): Segmentation process of four square objects with a color similar to the background color to a large extent with high illumination, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result



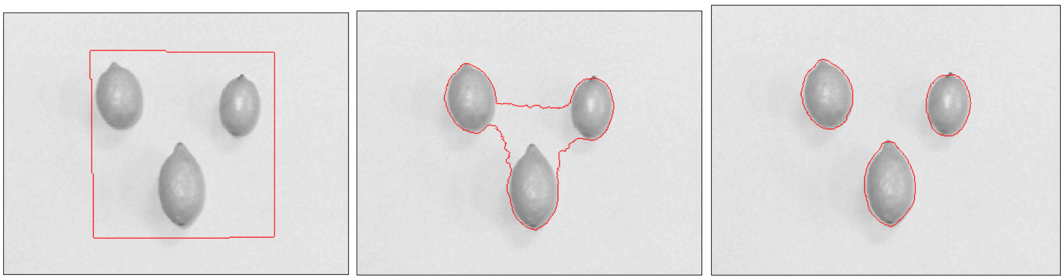
(a) (b) (c)

Fig. (15): Segmentation process of four square objects with a color similar to the background color to a large extent with medium illumination, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result



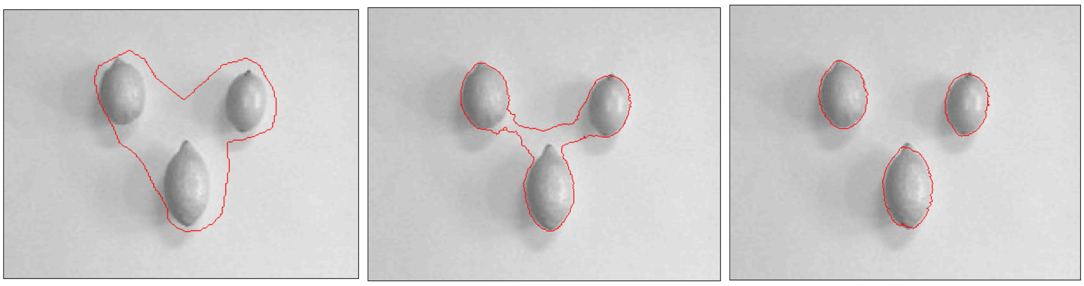
(a) (b) (c)

Fig. (16): Segmentation process of four square objects with a color similar to the background color to a large extent with low illumination, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result



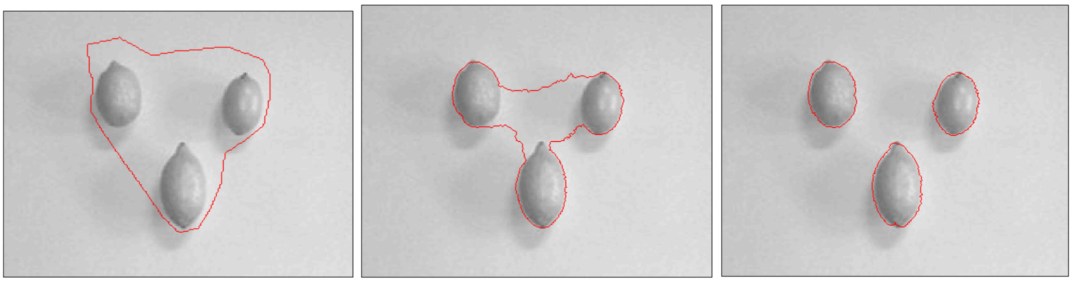
(a) (b) (c)

Fig. (17): Segmentation process of three lemons with high illumination, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result



(a) (b) (c)

Fig. (18): Segmentation process of three lemons with medium illumination, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result



(a) (b) (c)

Fig. (19): Segmentation process of three lemons with low illumination, (a) input image with the initial front, (b) initial front evolution, (c) segmentation result

تحديد القيم المثالية لمعاملات النماذج الهندسية القابلة للتشوه لفصل العناصر المتعددة في الصور الرقمية

عبد المنعم صالح رحمة
قسم علوم الحاسبات / الجامعة التكنولوجية
اسراء عبد الجبار علوان
قسم علوم الحاسبات / كلية التربية للعلوم الصرفة (أبن الهيثم) / جامعة بغداد

استلم البحث في: 5 أيلول 2012 ، قبل البحث في: 20 تشرين الثاني 2012

الخلاصة

الدقة في فصل العناصر المتعددة بأستخدام النماذج الهندسية القابلة للتشوه لا تتحقق احيانا لأسباب تتعلق بعدد من المعاملات الخاصة بالانموذج المستخدم . في هذا البحث درس تأثير تغيير قيم المعاملات في عمل النماذج الهندسية القابلة للتشوه وايجاد القيم الكفوءة لها، كذلك ايجاد العلاقات التي تربط هذه المعاملات مع بعضها وذلك بالاعتماد على حالات دراسية مختلفة تتضمن عناصر متعددة مختلفة بالمسافات, اللون, والاضاءة لنطاقات معينة من قيم المعاملات تم العثور على نتائج فصل جيدة، اذ حصر نجاح عمل النماذج الهندسية القابلة للتشوه ضمن حدود معينة لقيم تلك المعاملات.

الكلمات المفتاحية : فصل العناصر المتعددة، النماذج الهندسية القابلة للتشوه، طريقة مجموعة المستوى.