Research Article

An Improved Computer Vision Method for White Blood Cells Detection

**Erik Cuevas, Margarita Díaz, Miguel Manzanares,
Daniel Zaldivar, and Marco Perez-Cisneros**

Departamento de Electrónica, Universidad de Guadalajara, CUCEI, Avenida Revolución 1500, 44430 Guadalajara, JAL, Mexico

Correspondence should be addressed to Erik Cuevas; erik.cuevas@cucei.udg.mx

The automatic detection of white blood cells (WBCs) still remains as an unsolved issue in medical imaging. The analysis of WBC images has engaged researchers from fields of medicine and computer vision alike. Since WBC can be approximated by an ellipsoid form, an ellipse detector algorithm may be successfully applied in order to recognize such elements. This paper presents an algorithm for the automatic detection of WBC embedded in complicated and cluttered smear images that considers the complete process as a multiellipse detection problem. The approach, which is based on the differential evolution (DE) algorithm, transforms the detection task into an optimization problem whose individuals represent candidate ellipses. An objective function evaluates if such candidate ellipses are actually present in the edge map of the smear image. Guided by the values of such function, the set of encoded candidate ellipses (individuals) are evolved using the DE algorithm so that they can fit into the WBCs which are enclosed within the edge map of the smear image. Experimental results from white blood cell images with a varying range of complexity are included to validate the efficiency of the proposed technique in terms of its accuracy and robustness.

1. Introduction

Medical image processing has become more and more important in diagnosis with the development of medical imaging and computer technique. Huge amounts of medical images are obtained by X-ray radiography, CT, and MRI. They provide essential information for efficient and accurate diagnosis based on advanced computer vision techniques [1, 2].

On the other hand, white blood cells (WBCs) also known as leukocytes play a significant role in the diagnosis of different diseases. Although computer vision techniques have successfully contributed to generating new methods for cell analysis, which, in turn, have led to more accurate and reliable systems for disease diagnosis, high variability on cell shape, size, edge, and localization complicates the data extraction process. Moreover, the contrast between cell boundaries and the image's background may vary due to unstable lighting conditions during the capturing process.

Many works have been conducted in the area of blood cell detection. In [3] a method based on boundary support vectors is proposed to identify WBC. In such approach, the intensity of each pixel is used to construct feature vectors whereas a support vector machine (SVM) is used for classification and segmentation. By using a different approach, in [4], Wu et al. developed an iterative Otsu method based on the circular histogram for leukocyte segmentation. According to such technique, the smear images are processed in the hue-saturation-intensity (HSI) space by considering that the hue component contains most of the WBC information. One of the latest advances in white blood cell detection research is the algorithm proposed by Wang [5] that is based on the fuzzy cellular neural network (FCNN). Although such method has proved successful in detecting only one leukocyte in the image, it has not been tested over images containing several white cells. Moreover, its performance commonly decays when the iteration number is not properly defined, yielding a challenging problem itself with no clear clues on how to make the best choice.

Since white blood cells can be approximated with an ellipsoid form, computer vision techniques for detecting ellipses may be used in order to recognize them. Ellipse

detection in real images is an open research problem long time ago. Several approaches have been proposed which traditionally fall under three categories: symmetry-based, Hough transform-based (HT) and random sampling.

In symmetry-based detection [6, 7], the ellipse geometry is taken into account. The most common elements used in ellipse geometry are the ellipse center and axis. Using these elements and edges in the image, the ellipse parameters can be found. Ellipse detection in digital images is commonly solved through the Hough transform [8]. It works by representing the geometric shape by its set of parameters and then accumulating bins in the quantized parameter space. Peaks in the bins provide the indication of where ellipses may be. Obviously, since the parameters are quantized into discrete bins, the intervals of the bins directly affect the accuracy of the results and the computational effort. Therefore, for fine quantization of the space, the algorithm returns more accurate results, while suffering from large memory loads and expensive computation. In order to overcome such a problem, some other researchers have proposed other ellipse detectors following the Hough transform principles by using random sampling. In random sampling-based approaches [9, 10], a bin represents a candidate shape rather than a set of quantized parameters, as in the HT. However, like the HT, random sampling approaches go through an accumulation process for the bins. The bin with the highest score represents the best approximation of an actual ellipse in the target image. McLaughlin's work [11] shows that a random sampling-based approach produces improvements in accuracy and computational complexity, as well as a reduction in the number of false positives (nonexistent ellipses), when compared to the original HT and the number of its improved variants.

As an alternative to traditional techniques, the problem of ellipse detection has also been handled through optimization methods. In general, they have demonstrated to give better results than those based on the HT and random sampling with respect to accuracy and robustness [13]. Such approaches have produced several robust ellipse detectors using different optimization algorithms such as genetic algorithms (GAs) [14, 15] and particle swarm optimization (PSO) [16].

Although detection algorithms based on optimization approaches present several advantages in comparison to traditional approaches, they have been scarcely applied to WBC detection. One exception is the work presented by Karkavitsas and Rangoussi [12] that solves the WBC detection problem through the use of GA. However, since the evaluation function, which assesses the quality of each solution, considers the number of pixels contained inside of a circle with fixed radius, the method is prone to produce misdetections particularly for images that contained overlapped or irregular WBC.

In this paper, the WBC detection task is approached as an optimization problem and the differential evolution algorithm is used to build the ellipsoidal approximation. Differential evolution (DE), introduced by Storn and Price [17], is a novel evolutionary algorithm which is used to optimize complex continuous nonlinear functions. As a population-based algorithm, DE uses simple mutation and crossover operators to generate new candidate solutions and

applies one-to-one competition scheme to greedily decide whether the new candidate or its parent will survive in the next generation. Due to its simplicity, ease of implementation, fast convergence, and robustness, the DE algorithm has gained much attention, reporting a wide range of successful applications in the literature [18–22].

This paper presents an algorithm for the automatic detection of blood cell images based on the DE algorithm. The proposed method uses the encoding of five edge points as candidate ellipses in the edge map of the smear. An objective function allows to accurately measure the resemblance of a candidate ellipse with an actual WBC on the image. Guided by the values of such objective function, the set of encoded candidate ellipses are evolved using the DE algorithm so that they can fit into actual WBC on the image. The approach generates a subpixel detector which can effectively identify leukocytes in real images. Experimental evidence shows the effectiveness of such method in detecting leukocytes despite complex conditions. Comparison to the state-of-the-art WBC detectors on multiple images demonstrates a better performance of the proposed method.

The main contribution of this study is the proposal of a new WBC detector algorithm that efficiently recognizes WBC under different complex conditions while considering the whole process as an ellipse detection problem. Although ellipse detectors based on optimization present several interesting properties, to the best of our knowledge, they have not yet been applied to any medical image processing up to date.

This paper is organized as follows: Section 2 provides a description of the DE algorithm while in Section 3 the ellipse detection task is fully explained from an optimization perspective within the context of the DE approach. The complete WBC detector is presented in Section 4. Section 5 reports the obtained experimental results whereas Section 6 conducts a comparison between state-of-the-art WBC detectors and the proposed approach. Finally, in Section 7, some conclusions are drawn.

2. Differential Evolution Algorithm

The DE algorithm is a simple and direct search algorithm which is based on population and aims for optimizing global multimodal functions. DE employs the mutation operator to provide the exchange of information among several solutions.

There are various mutation base generators to define the algorithm type. The version of DE algorithm used in this work is known as rand-to-best/1/bin or "DE1" [23]. DE algorithms begin by initializing a population of N_p and D-dimensional vectors considering parameter values that are randomly distributed between the prespecified lower initial parameter bound $x_{j,\text{low}}$ and the upper initial parameter bound $x_{j,\text{high}}$ as follows:

$$x_{j,i,t} = x_{j,\text{low}} + \text{rand}(0,1) \cdot \left(x_{j,\text{high}} - x_{j,\text{low}}\right);$$
$$j = 1, 2, \ldots, D; \ i = 1, 2, \ldots, N_p; \ t = 0. \tag{1}$$

The subscript t is the generation index, while j and i are the parameter and particle indexes, respectively. Hence, $x_{j,i,t}$

is the jth parameter of the ith particle in generation t. In order to generate a trial solution, DE algorithm first mutates the best solution vector $\mathbf{x}_{\text{best},t}$ from the current population by adding the scaled difference of two vectors from the current population:

$$\mathbf{v}_{i,t} = \mathbf{x}_{\text{best},t} + F \cdot \left(\mathbf{x}_{r_1,t} - \mathbf{x}_{r_2,t}\right); \quad r_1, r_2 \in \left\{1, 2, \ldots, N_p\right\} \tag{2}$$

with $\mathbf{v}_{i,t}$ being the mutant vector. Indices r_1 and r_2 are randomly selected with the condition that they are different and have no relation to the particle index i whatsoever (i.e., $r_1 \neq r_2 \neq i$). The mutation scale factor F is a positive real number, typically less than one. Figure 1 illustrates the vector-generation process defined by (2).

In order to increase the diversity of the parameter vector, the crossover operation is applied between the mutant vector $\mathbf{v}_{i,t}$ and the original individuals $\mathbf{x}_{i,t}$. The result is the trial vector $\mathbf{u}_{i,t}$ which is computed by considering element to element as follows:

$$u_{j,i,t} = \begin{cases} v_{j,i,t}, & \text{if } \text{rand}\,(0,1) \le \text{CR or } j = j_{\text{rand}}, \\ x_{j,i,t}, & \text{otherwise}, \end{cases} \tag{3}$$

with $j_{\text{rand}} \in \{1, 2, \ldots, D\}$. The crossover parameter ($0.0 \le \text{CR} \le 1.0$) controls the fraction of parameters that the mutant vector is contributing to the final trial vector. In addition, the trial vector always inherits the mutant vector parameter according to the randomly chosen index j_{rand}, assuring that the trial vector differs by at least one parameter from the vector to which it is compared ($\mathbf{x}_{i,t}$).

Finally, a greedy selection is used to find better solutions. Thus, if the computed cost function value of the trial vector $\mathbf{u}_{i,t}$ is less than or equal to the cost of the vector $\mathbf{x}_{i,t}$, then such trial vector replaces $\mathbf{x}_{i,t}$ in the next generation. Otherwise, $\mathbf{x}_{i,t}$ remains in the population for at least one more generation:

$$\mathbf{x}_{i,t+1} = \begin{cases} \mathbf{u}_{i,t}, & \text{if } f\left(\mathbf{u}_{i,t}\right) \le f\left(\mathbf{x}_{i,t}\right), \\ \mathbf{x}_{i,t}, & \text{otherwise}. \end{cases} \tag{4}$$

Here, $f(\ )$ represents the objective function. These processes are repeated until a termination criterion is attained or a predetermined generation number is reached.

3. Ellipse Detection Using DE

3.1. Data Preprocessing.
In order to detect ellipse shapes, candidate images must be preprocessed first by an edge detection algorithm which yields an edge map image. Then, the (x_i, y_i) coordinates for each edge pixel p_i are stored inside the edge vector $P = \{p_1, p_2, \ldots, p_{N_p}\}$, with N_p being the total number of edge pixels.

3.2. Individual Representation.
Just as a line requires two points to completely define its characteristics, an ellipse is defined by five points. Therefore, each candidate solution E (ellipse candidate) considers five edge points to represent an individual. Under such representation, edge points are selected following a random positional index within the edge array P. This procedure will encode a candidate solution as the ellipse that passes through five points p_1, p_2, p_3, p_4, and p_5 ($E = \{p_1, p_2, p_3, p_4, p_5\}$). Thus, by substituting the coordinates of each point of E into (5), we gather a set of five simultaneous equations which are linear in the five unknown parameters a, b, f, g, and h:

$$ax^2 + 2hxy + by^2 + 2gx + 2fy + 1 = 0. \tag{5}$$

Considering the configuration of the edge points shown by Figure 2, the ellipse center (x_0, y_0), the radius maximum ($r_{\max}$), the radius minimum ($r_{\min}$), and the ellipse orientation (θ) can be calculated as follows:

$$\begin{aligned} x_0 &= \frac{hf - bg}{C}, \\ y_0 &= \frac{gh - af}{C}, \\ r_{\max} &= \sqrt{\frac{-2\Delta}{C(a + b - R)}}, \\ r_{\min} &= \sqrt{\frac{-2\Delta}{C(a + b + R)}}, \\ \theta &= \frac{1}{2}\arctan\left(\frac{2h}{a - b}\right), \end{aligned} \tag{6}$$

where

$$R^2 = (a - b)^2 + 4h^2, \qquad C = ab - h^2,$$

$$\Delta = \det\left(\begin{vmatrix} a & h & g \\ h & b & f \\ g & f & 1 \end{vmatrix}\right). \tag{7}$$

3.3. Objective Function.
Optimization refers to choosing the best element from one set of available alternatives. In the simplest case, it means to minimize an objective function or error by systematically choosing the values of variables from their valid ranges. In order to calculate the error produced by a candidate solution E, the ellipse coordinates are calculated as a virtual shape which, in turn, must also be validated, if it really exists in the edge image. The test set is represented by $S = \{s_1, s_2, \ldots, s_{N_s}\}$, where N_s are the number of points over which the existence of an edge point, corresponding to E, should be tested.

The set S is generated by the Midpoint Ellipse Algorithm (MEA) [24] which is a searching method that seeks required points for drawing an ellipse. For any point (x, y) lying on the boundary of the ellipse with a, h, b, g, and f, it does satisfy the equation $f_{\text{ellipse}}(x, y) \cong r_{\max} x^2 + r_{\min} y^2 - r_{\max}^2 r_{\min}^2$, where $r_{\max}$ and $r_{\min}$ represent the major and minor axes, respectively. However, MEA avoids computing square-root calculations by comparing the pixel separation distances. A method for direct distance comparison is to test the halfway position between two pixels (subpixel distance) to determine if this midpoint is inside or outside the ellipse boundary. If the point is in the
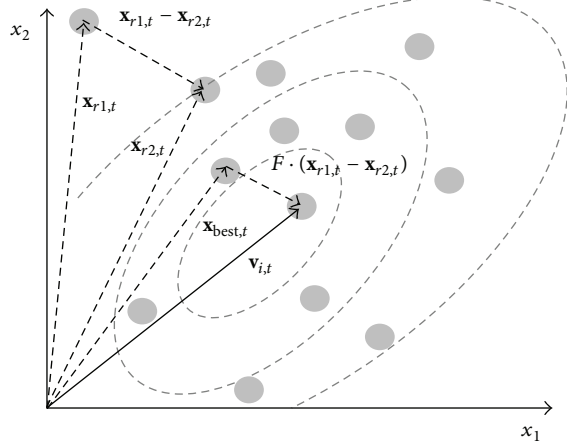
FIGURE 1: Two-dimensional example of an objective function showing its contour lines and the process for generating **v** in scheme DE/best/l/exp from vectors of the current generation.
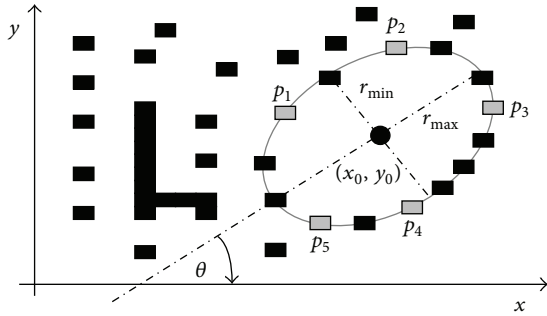

FIGURE 2: Ellipse candidate (individual) built from the combination of points p_1, p_2, p_3, p_4, and p_5.

interior of the ellipse, the ellipse function is negative. Thus, if the point is outside the ellipse, the ellipse function is positive. Therefore, the error involved in locating pixel positions using the midpoint test is limited to one-half the pixel separation (subpixel precision). To summarize, the relative position of any point (x, y) can be determined by checking the sign of the ellipse function:

$$f_{\text{ellipse}}(x, y) \begin{cases} < 0 & \text{if } (x, y) \text{ is inside the ellipse boundary} \\ = 0 & \text{if } (x, y) \text{ is on the ellipse boundary} \\ > 0 & \text{if } (x, y) \text{ is outside the ellipse boundary.} \end{cases} \tag{8}$$

The ellipse-function test in (8) is applied to midpositions between pixels nearby the ellipse path at each sampling step. Figures 3(a) and 3(b) show the midpoint between the two candidate pixels at sampling position. The ellipse is used to divide the quadrants into two regions; the limit of the two regions is the point at which the curve has a slope of −1 as shown in Figure 4.

In MEA the computation time is reduced by considering the symmetry of ellipses. Ellipses sections in adjacent octants within one quadrant are symmetric with respect to the $dy/dy = -1$ line dividing the two octants. These symmetry

conditions are illustrated in Figure 4. The algorithm can be considered as the quickest providing a subpixel precision [25]. However, in order to protect the MEA operation, it is important to assure that points lying outside the image plane must not be considered in S.

The objective function $J(E)$ represents the matching error produced between the pixels S of the ellipse candidate E and the pixels that actually exist in the edge image, yielding

$$J(E) = 1 - \frac{\sum_{v=1}^{N_s} G(x_v, y_v)}{N_s}, \tag{9}$$

where $G(x_i, y_i)$ is a function that verifies the pixel existence in (x_v, y_v), with $(x_v, y_v) \in S$ and N_s being the number of pixels lying on the perimeter corresponding to E currently under testing. Hence, function $G(x_v, y_v)$ is defined as

$$G(x_v, y_v) = \begin{cases} 1 & \text{if the pixel } (x_v, y_v) \text{ is an edge point} \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

A value of $J(E)$ near to zero implies a better response from the "ellipsoid" operator. Figure 5 shows the procedure to evaluate a candidate action E with its representation as a virtual shape S. Figure 5(a) shows the original edge map, while Figure 5(b) presents the virtual shape S representing the individual $E = \{p_1, p_2, p_3, p_4, p_5\}$. In Figure 5(c), the virtual shape S is compared to the original image, point by point, in order to find coincidences between virtual and edge points. The individual has been built from points p_1, p_2, p_3, p_4, and p_5 which are shown by Figure 5(a). The virtual shape S, obtained by MEA, gathers 52 points ($N_s = 52$) with only 35 of them existing in both images (shown as darker points in Figure 5(c)) and yielding $\sum_{v=1}^{N_s} G(x_v, y_v) = 35$; therefore $J(E) = 0.327$.

3.4. Implementation of DE for Ellipse Detection. The ellipse detector algorithm based on DE can be summarized in Algorithm 1.

4. The White Blood Cell Detector

In order to detect WBC, the proposed detector combines a segmentation strategy with the ellipse detection approach presented in Section 3.

4.1. Image Preprocessing. To employ the proposed detector, smear images must be preprocessed to obtain two new images: the segmented image and its corresponding edge map. The segmented image is produced by using a segmentation strategy whereas the edge map is generated by a border extractor algorithm. Such edge map is considered by the objective function to measure the resemblance of a candidate ellipse with an actual WBC.

The goal of the segmentation strategy is to isolate the white blood cells (WBC's) from other structures such as red blood cells and background pixels. Information of color, brightness, and gradients is commonly used within a thresholding scheme to generate the labels to classify each pixel.
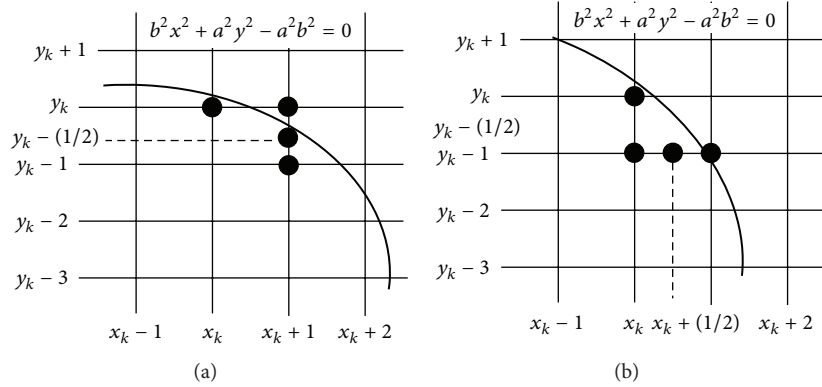
FIGURE 3: (a) Symmetry of the ellipse: an estimated one octant which belongs to the first region where the slope is greater than −1. (b) In this region the slope will be less than −1 to complete the octant and continue to calculate the same as the remaining octants.
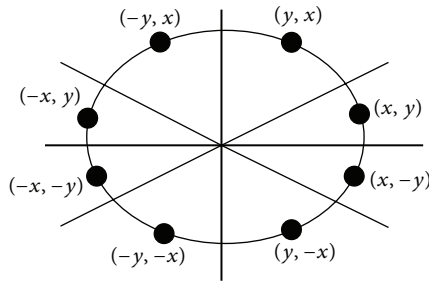

FIGURE 4: Midpoint between candidate pixels at sampling position x_k along an elliptical path.

Although a simple histogram thresholding can be used to segment the WBCs, at this work the diffused expectation-maximization (DEM) has been used to assure better results [26].

DEM is an expectation-maximization- (EM-) based algorithm which has been used to segment complex medical images [27]. In contrast to classical EM algorithms, DEM considers the spatial correlations among pixels as a part of the minimization criteria. Such adaptation allows to segment objects in spite of noisy and complex conditions. The method models an image as a finite mixture, where each mixture component corresponds to a region class and uses a maximum likelihood approach to estimate the parameters for each class, via the expectation maximization (EM) algorithm, which is coupled to anisotropic diffusion over classes in order to account for the spatial dependencies among pixels.

For the WBC's segmentation, the implementation of DEM provided in [17] has been used. Since the implementation allows to segment gray-level images and color images, it can be used for operating over all smear images with no regard about how each image has been acquired. The DEM has been configured considering three different classes ($K = 3$), $g(\nabla h_{ik}) = |\nabla h_{ik}|^{-9/5}$, $\lambda = 0.1$, and $m = 10$ iterations. These values have been found as the best configuration set according to [26].

As a final result of the DEM operation, three different thresholding points are obtained: the first corresponds to the WBCs and the second to the red blood cells whereas the third represents the pixels classified as background. Figure 6(b) presents the segmentation results obtained by the DEM approach employed at this work considering Figure 6(a) as the original image.

Once the segmented image has been produced, the edge map is computed. The purpose of the edge map is to obtain a simple image representation that preserves object structures. The DE-based detector operates directly over the edge map in order to recognize ellipsoidal shapes. Several algorithms can be used to extract the edge map; however, at this work, the morphological edge detection procedure [28] has been used to accomplish such a task. Morphological edge detection is a traditional method to extract borders from binary images in which original images (I_B) are eroded by a simple structure element (I_E) composed by a matrix template of 3×3 with all its values equal to one. Then, the eroded image is inverted ($\bar{I}_E$) and compared with the original image ($\bar{I}_E \wedge I_B$) in order to detect pixels which are present in both images. Such pixels compose the computed edge map from I_B. Figure 6(c) shows the edge map obtained by using the morphological edge detection procedure.

4.2. Ellipse Detection Approach. The edge map is used as input image for the ellipse detector presented in Section 3. Table 1 presents the parameter set that has been used in this work for the DE algorithm after several calibration examples have been conducted. The final configuration matches the best possible calibration proposed in [29], where the effect of modifying the DE-parameters for several generic optimization problems has been analyzed. The population-size parameter ($m = 20$) has been selected considering the best possible balance between convergence and computational overload. Once it has been set, such configuration has been kept for all test images employed in the experimental study.

Under such assumptions, the complete process to detect WBCs is implemented as in Algorithm 2.

4.3. Numerical Example. In order to present the algorithm's step-by-step operation, a numerical example has been set by applying the proposed method to detect a single leukocyte
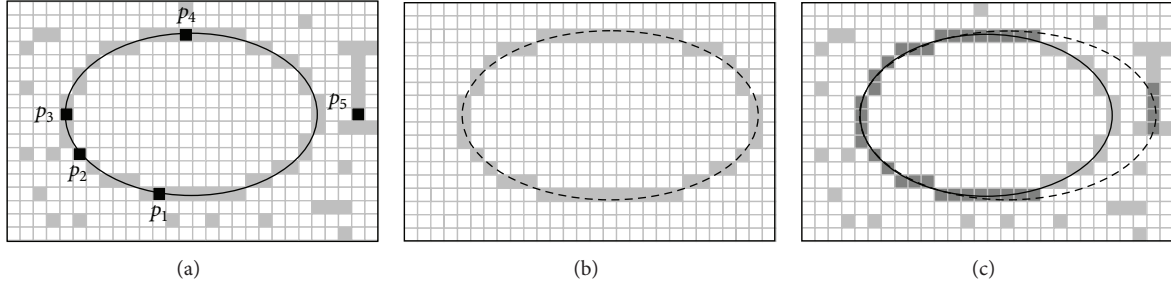
FIGURE 5: Evaluation of a candidate solution E: the image in (a) shows the original image while (b) presents the generated virtual shape drawn from points p_1, p_2, p_3, p_4, and p_5. The image in (c) shows coincidences between both images which have been marked by darker pixels while the virtual shape is also depicted through a dashed line.

Step 1: Set the DE parameters $F = 0.25$ and CR $= 0.8$.
Step 2: Initialize the population of m individuals $\mathbf{E}^k = \{E_1^k, E_2^k, \ldots, E_m^k\}$ where each decision
        variable p_1, p_2, p_3, p_4 and p_5 of E_a^k is set randomly within the interval $[1, N_p]$. All
        values must be integers. Considering that $k = 0$ and $a \in (1, 2, \ldots, m)$.
Step 3: Evaluate the objective value $J(E_a^k)$ for all m individuals, and determining the $E^{\text{best},k}$ showing
        the best fitness value, such that $E^{\text{best},k} \in \{\mathbf{E}^k\} \big| J(E^{\text{best},k}) = \min\{J(E_1^k), J(E_2^k), \ldots, J(E_m^k)\}$.
Step 4: Generate the trial population $\mathbf{T} = \{T_1, T_2, \ldots, T_m\}$:
        for $(i = 1; i < m + 1; i{+}{+})$
        do $r_1 = \text{floor}(\text{rand}(0, 1) \cdot m)$; while $(r_1 = i)$;
        do $r_2 = \text{floor}(\text{rand}(0, 1) \cdot m)$; while $((r_2 = i)$ or $(r_2 = r_2))$;
        jrand $= \text{floor}(5 \cdot \text{rand}(0, 1))$;
                for $(j = 1; j < 6; j{+}{+})$ // generate a trial vector
                    if $(\text{rand}(0,1){<}{=}\text{CR}$ or $j = \text{jrand})$
                    $T_{j,i} = E_j^{\text{best},k} + F \cdot (E_{j,r_1}^k - E_{j,r_2}^k)$;
                        else
                            $T_{j,i} = E_{j,i}^k$;
                    end if
                end for
        end for
Step 5: Evaluate the fitness values $J(T_i)$ $(i \in \{1, 2, \ldots, m\})$ of all trial individuals. Check all
        individuals. If a candidate parameter set is not physically plausible, i.e. out of the
        range $[1, N_p]$, then an exaggerated cost function value is returned. This aims to
        eliminate "unstable" individuals.
Step 6: Select the next population $\mathbf{E}^{k+1} = \{E_1^{k+1}, E_2^{k+1}, \ldots, E_m^{k+1}\}$:
        for $(i = 1; i < m + 1; i{+}{+})$
            if $(J(T_i) < J(E_i^k))$
            $E_i^{k+1} = T_i$
                else
                    $E_i^{k+1} = E_i^k$
            end if
        end for
Step 7: If the iteration number (N_I) is met, then the output $E^{\text{best},k}$ is the solution (an actual
        ellipse contained in the image), otherwise go back to Step 3.

ALGORITHM 1

Step 1: Segment the WBC's using the DEM algorithm (described in Section 4.1)
Step 2: Get the edge map from the segmented image.
Step 3: Start the ellipse detector based in DE over the edge map while saving best ellipses (Section 3).
Step 4: Define parameter values for each ellipse that identify the WBC's.
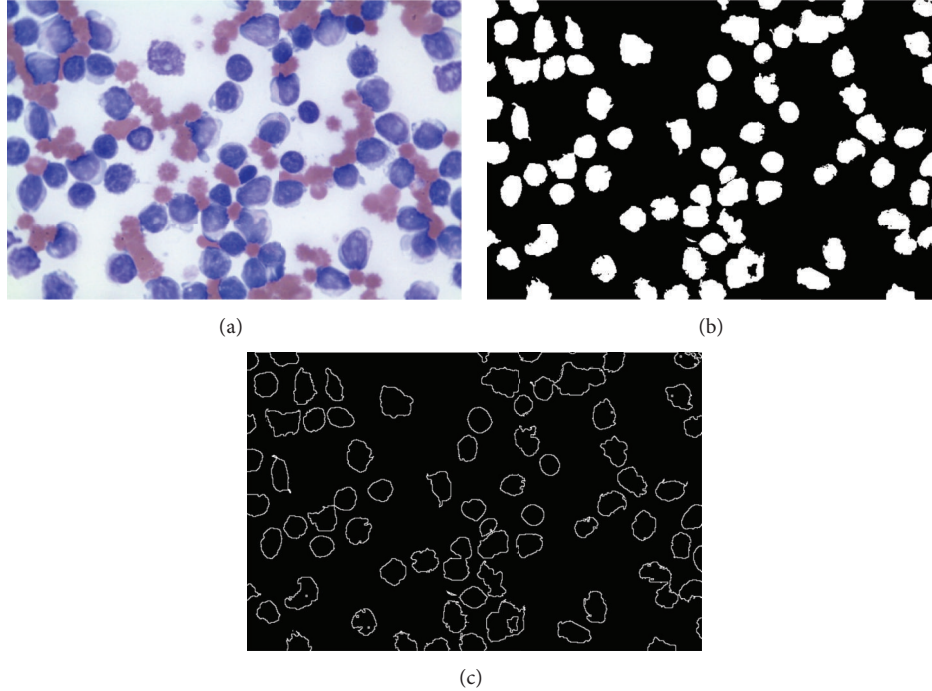
ALGORITHM 2

(a)

(b)

(c)

FIGURE 6: Preprocessing process: (a) original smear image, (b) segmented image obtained by DEM, and (c) the edge map obtained by using the morphological edge detection procedure.

TABLE 1: DE parameters used for leukocytes detection in medical images.

m	F	CR	N_I
20	0.25	0.80	200

lying inside of a simple image. Figure 7(a) shows the image used in the example. After applying the threshold operation, the WBC is located beside few other pixels which are merely noise (see Figure 7(b)). Then, the edge map is subsequently computed and stored pixel by pixel inside the vector P. Figure 7(c) shows the resulting image after such procedure.

The DE-based ellipse detector is executed using information of the edge map (for the sake of easiness, it only considers a population of four particles). Like all evolutionary approaches, DE is a population-based optimizer that attacks the starting point problem by sampling the search space at multiple, randomly chosen, and initial particles. By taking five random pixels from vector P, four different particles are constructed. Figure 7(d) depicts the initial particle distribution $\mathbf{E}^0 = \{E_1^0, E_2^0, E_3^0, E_4^0\}$. By using the DE operators, four different trial particles $\mathbf{T} = \{T_1, T_2, T_3, T_4\}$ (ellipses) are generated and their locations are shown in Figure 7(e). Then, the new population $\mathbf{E}^1$ is selected considering the best elements obtained among the trial elements $\mathbf{T}$ and the initial particles $\mathbf{E}^0$. The final distribution of the new population is depicted in Figure 7(f). Since the particles E_2^0 and E_2^0 hold (in Figure 7(f)) a better fitness value ($J(E_2^0)$ and $J(E_3^0)$) than the trial elements T_2 and T_3, they are considered as particles of the final population $\mathbf{E}^1$. Figures 7(g) and 7(h) present the second

iteration produced by the algorithm whereas Figure 7(i) shows the population configuration after 25 iterations. From Figure 7(i), it is clear that all particles have converged to a final position which is able to accurately cover the WBC.

5. Experimental Results

Experimental tests have been developed in order to evaluate the performance of the WBC detector. It was tested over microscope images from blood smears holding a 960×720 pixel resolution. They correspond to supporting images on the leukemia diagnosis. The images show several complex conditions such as deformed cells and overlapping with partial occlusions. The robustness of the algorithm has been tested under such demanding conditions. All the experiments have been developed using an Intel Core i7-2600 PC, with 8 GB in RAM.

Figure 8(a) shows an example image employed in the test. It was used as input image for the WBC detector. Figure 8(b) presents the segmented WBCs obtained by the DEM algorithm. Figures 8(c) and 8(d) present the edge map and the white blood cells after detection, respectively. The results show that the proposed algorithm can effectively detect and mark blood cells despite cell occlusion, deformation, or overlapping. Other parameters may also be calculated through the algorithm: the total area covered by white blood cells and relationships between several cell sizes.

Another example is presented in Figure 9. It represents a complex example with an image showing seriously deformed cells. Despite such imperfections, the proposed approach can effectively detect the cells as it is shown in Figure 9(d).
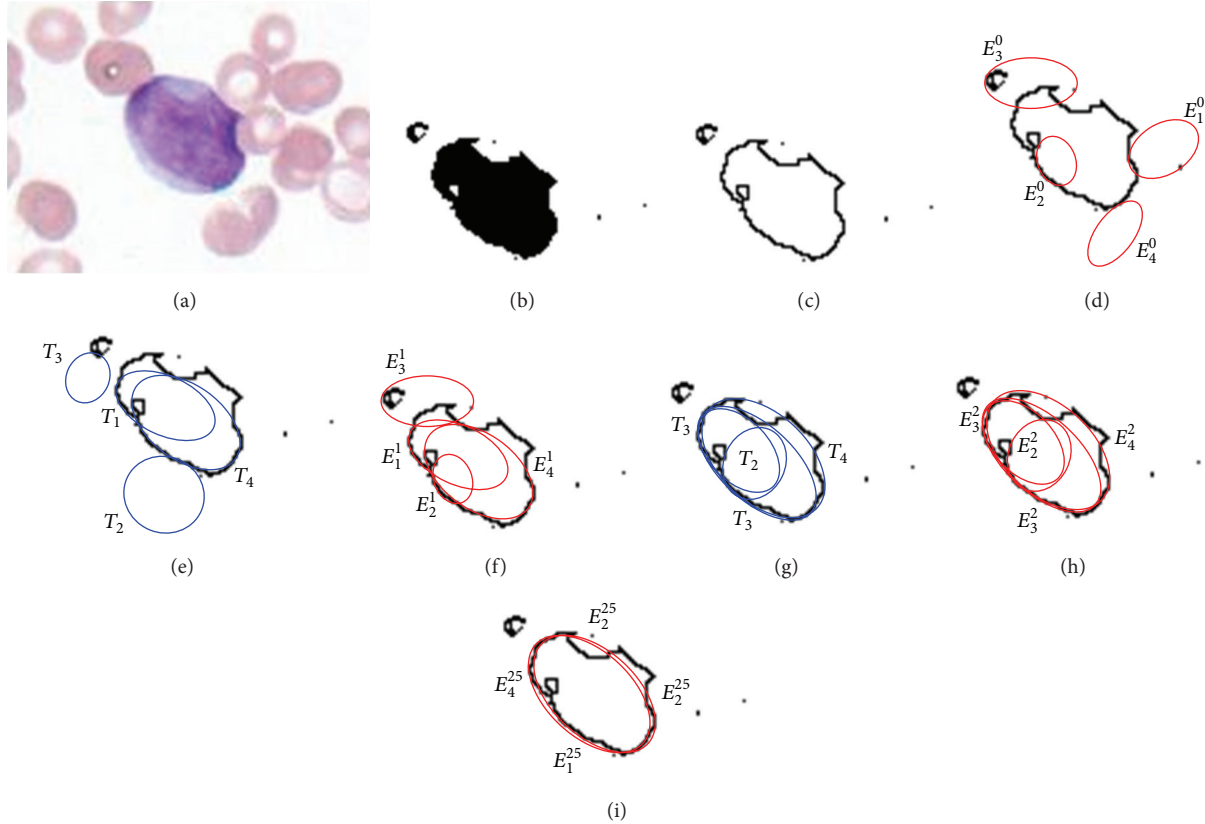
FIGURE 7: Detection numerical example: (a) the image used as an example, (b) segmented image, (c) edge map, (d) initial particles $\mathbf{E}^0$, (e) trial elements $\mathbf{T}$ produced by the DE operators, (f) new population $\mathbf{E}^1$, (g) trial elements produced considering $\mathbf{E}^1$ as input population, (h) new population $\mathbf{E}^2$, and (i) final particle configuration after 25 iterations.

6. Comparisons to Other Methods

A comprehensive set of smear-blood test images is used to test the performance of the proposed approach. We have applied the proposed DE-based detector to test images in order to compare its performance to other WBC detection algorithms such as the boundary support vectors (BSVs) approach [3], the iterative Otsu (IO) method [4], the Wang algorithm [5], and the genetic algorithm-based (GAB) detector [12]. In all cases, the algorithms are tuned according to the value set which is originally proposed by their own references.

6.1. Detection Comparison. To evaluate the detection performance of the proposed detection method, Table 2 tabulates the comparative leukocyte detection performance of the BSV approach, the IO method, the Wang algorithm, the BGA detector, and the proposed method, in terms of detection rates and false alarms. The experimental data set includes 50 images which are collected from the ASH Image Bank (http://imagebank.hematology.org/). Such images contain 517 leukocytes (287 bright leukocytes and 230 dark leukocytes according to smear conditions) which have been detected and counted by a human expert. Such values act as ground truth for all the experiments. For the comparison, the detection rate (DR) is defined as the ratio between the number of leukocytes

correctly detected and the number leukocytes determined by the expert. The false alarm rate (FAR) is defined as the ratio between the number of nonleukocyte objects that have been wrongly identified as leukocytes and the number leukocytes which have been actually determined by the expert.

Experimental results show that the proposed DE method, which achieves 98.26% leukocyte detection accuracy with 2.71% false alarm rate, is compared favorably with other WBC detection algorithms, such as the BSV approach, the IO method, the Wang algorithm, and the BGA detector.

6.2. Robustness Comparison. Images of blood smear are often deteriorated by noise due to various sources of interference and other phenomena that affect the measurement processes in imaging and data acquisition systems. Therefore, the detection results depend on the algorithm's ability to cope with different kinds of noises. In order to demonstrate the robustness in the WBC detection, the proposed DE approach is compared to the BSV approach, the IO method, the Wang algorithm, and the BGA detector under noisy environments. In the test, two different experiments have been studied. The first inquest explores the performance of each algorithm when the detection task is accomplished over images corrupted by salt and pepper noise. The second experiment considers images polluted by Gaussian noise.
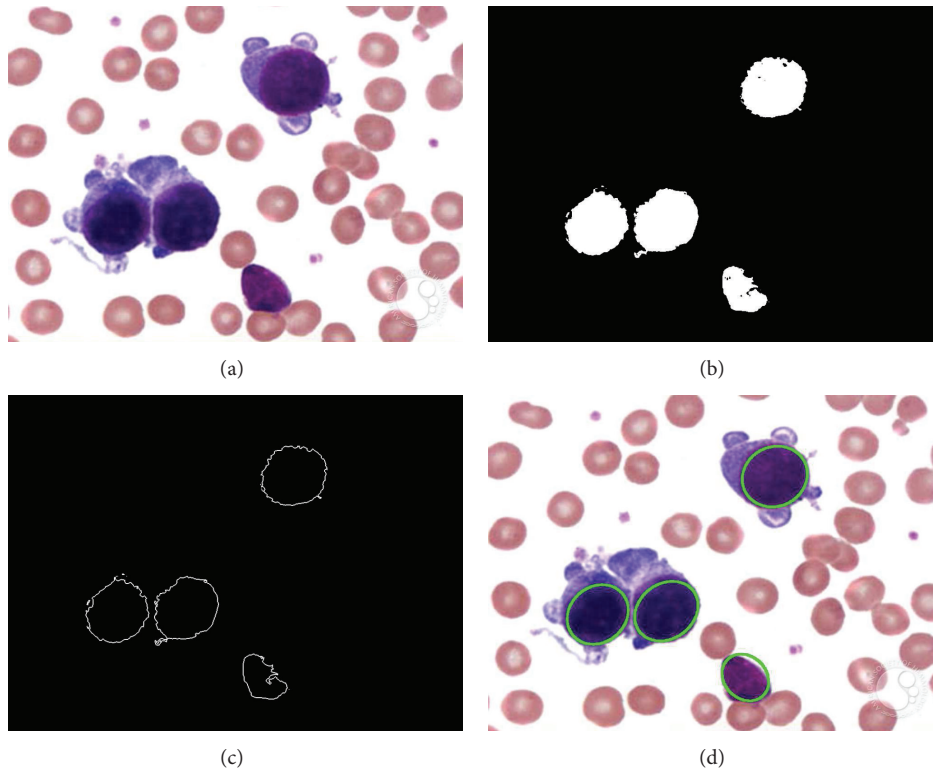
(a)

(b)

(c)

(d)

Figure 8: Resulting images of the first test after applying the WBC detector: (a) original image, (b) image segmented by the DEM algorithm, (c) edge map, and (d) the white detected blood cells.
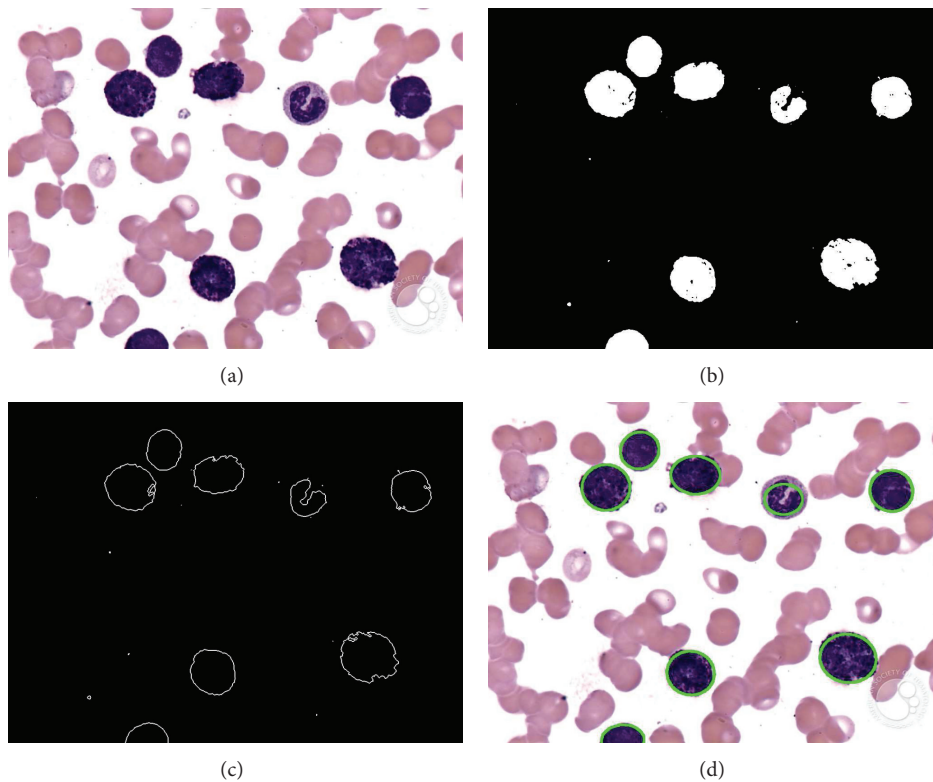

(a)

(b)

(c)

(d)

Figure 9: Resulting images of the second test after applying the WBC detector: (a) original image, (b) image segmented by the DEM algorithm, (c) edge map, and (d) the white detected blood cells.

TABLE 2: Comparative leukocyte detection performance of the BSV approach, the IO method, the Wang algorithm, the BGA detector, and the proposed DE method over the data set which contains 30 images and 426 leukocytes.

Leukocyte type	Method	Leukocytes detected	Missing	False alarms	DR	FAR
	BSV [3]	130	157	84	45.30%	29.27%
	IO [4]	227	60	73	79.09%	25.43%
Bright leukocytes (287)	Wang [5]	231	56	60	80.49%	20.90%
	GAB [12]	220	67	22	76.65%	7.66%
	DE based	281	6	11	97.91%	3.83%
	BSV [3]	105	125	59	46.65%	25.65%
	IO [4]	183	47	61	79.56%	26.52%
Dark leukocytes (230)	Wang [5]	196	34	47	85.22%	20.43%
	GAB [12]	179	51	23	77.83%	10.00%
	DE based	227	3	3	98.70%	1.30%
	BSV [3]	235	282	143	45.45%	27.66%
	IO [4]	410	107	134	79.30%	25.92%
Overall (517)	Wang [5]	427	90	107	82.59%	20.70%
	GAB [12]	399	118	45	77.18%	8.70%
	DE based	508	9	14	98.26%	2.71%

Salt and pepper and Gaussian noises are selected for the robustness analysis because they represent the most compatible noise types commonly found in images of blood smear [30]. The comparison considers the complete set of 50 images presented in Section 6.1 containing 517 leukocytes which have been detected and counted by a human expert. The added noise is produced by MATLAB, considering two noise levels of 5% and 10% for salt and pepper noise, whereas $\sigma = 5$ and $\sigma = 10$ are used for the case of Gaussian noise. Such noise levels, according to [31], correspond to the best tradeoff between detection difficulty and the real existence in medical imaging. If higher noise levels are used, then the detection process would be unnecessarily complicated without representing a feasible image condition.

Figure 10 shows two examples of the experimental set. The outcomes in terms of the detection rate (DR) and the false alarm rate (FAR) are reported for each noise type in Tables 3 and 4. The results show that the proposed DE algorithm presents the best detection performance, achieving in the worst case a DR of 89.55% and 91.10%, under contaminated conditions of salt and pepper and Gaussian noises, respectively. On the other hand, the DE detector possesses the least degradation performance presenting a FAR value of 5.99% and 6.77%.

6.3. Stability Comparison. In order to compare the stability performance of the proposed method, its results are compared to those reported by Wang et al. in [5] which is considered as an accurate technique for the detection of WBC.

The Wang algorithm is an energy-minimizing method which is guided by internal constraint elements and influenced by external image forces, producing the segmentation of WBC's at a closed contour. As external forces, the Wang approach uses edge information which is usually represented by the gradient magnitude of the image. Therefore, the contour is attracted to pixels with large image gradients, that is, strong edges. At each iteration, the Wang method finds a new contour configuration which minimizes the energy that corresponds to external forces and constraint elements.

In the comparison, the net structure and its operational parameters, corresponding to the Wang algorithm, follow the configuration suggested in [5] while the parameters for the DE-based algorithm are taken from Table 1.

Figure 11 shows the performance of both methods considering a test image with only two white blood cells. Since the Wang method uses gradient information in order to appropriately find a new contour configuration, it needs to be executed iteratively in order to detect each structure (WBC). Figure 11(b) shows the results after the Wang approach has been applied considering only 200 iterations. Furthermore, Figure 11(c) shows results after applying the DE-based method which has been proposed in this paper.

The Wang algorithm uses the fuzzy cellular neural network (FCNN) as optimization approach. It employs gradient information and internal states in order to find a better contour configuration. In each iteration, the FCNN tries, as contour points, different new pixel positions which must be located nearby the original contour position. Such fact might cause the contour solution to remain trapped into a local minimum. In order to avoid such a problem, the Wang method applies a considerable number of iterations so that a near optimal contour configuration can be found. However, when the number of iterations increases, the possibility to cover other structures increases too. Thus, if the image has a complex background (just as smear images do) or the WBC's are too close, the method gets confused so that finding the correct contour configuration from the gradient magnitude is not easy. Therefore, a drawback of Wang's method is related to its optimal iteration number (instability). Such number must be determined experimentally as it depends on the image context and its complexity. Figure 12(a) shows the result of
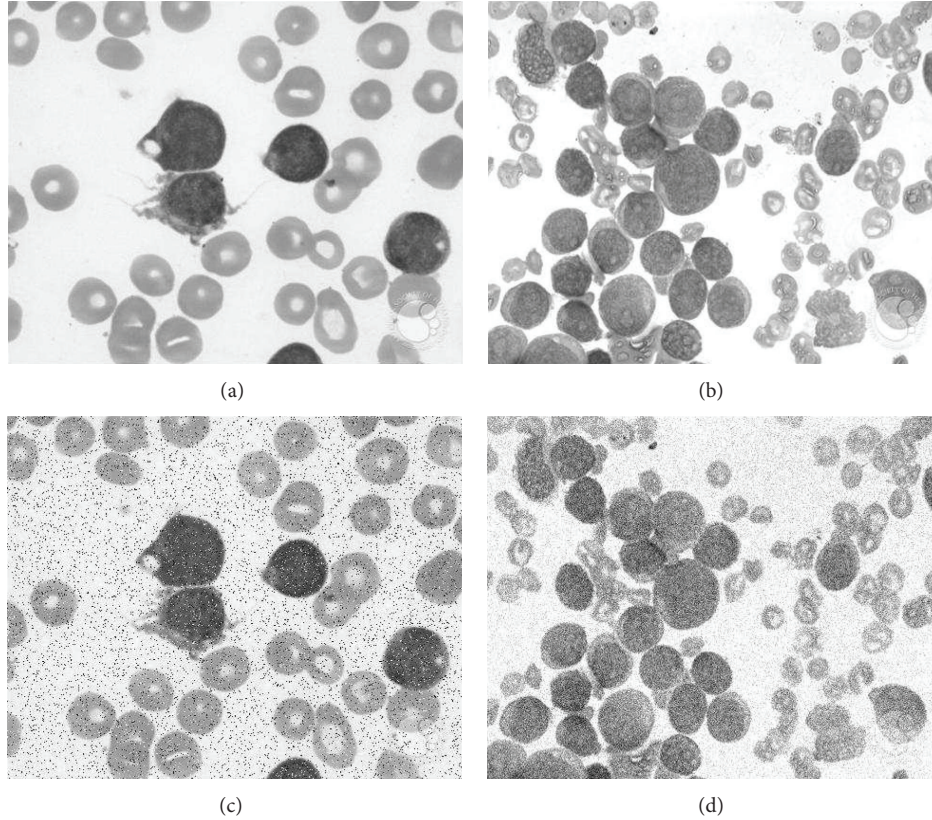
FIGURE 10: Examples of images included in the experimental set for robustness comparison: (a) and (b) original images, (c) image contaminated with 10% of salt and pepper noise, and (d) image polluted with $\sigma = 10$ of Gaussian noise.

TABLE 3: Comparative WBC detection among methods that considers the complete data set of 30 images corrupted by different levels of salt and pepper noise.

Noise level	Method	Leukocytes detected	Missing	False alarms	DR	FAR
5% salt and pepper noise 517 leukocytes	BSV [3]	185	332	133	34.74%	26.76%
	IO [4]	311	206	106	63.38%	24.88%
	Wang [5]	250	176	121	58.68%	27.70%
	GAB [12]	298	219	135	71.83%	24.18%
	DE based	482	35	32	91.55%	7.04%
10% salt and pepper noise 517 leukocytes	BSV [3]	105	412	157	20.31%	30.37%
	IO [4]	276	241	110	53.38%	21.28%
	Wang [5]	214	303	168	41.39%	32.49%
	GAB [12]	337	180	98	65.18%	18.95%
	DE based	463	54	31	89.55%	5.99%

applying 400 cycles of Wang's algorithm while Figure 12(b) presents the detection of the same cell shapes after 1000 iterations using the proposed algorithm. From Figure 12(a), it can be seen that the contour produced by Wang's algorithm degenerates as the iteration process continues, wrongly covering other shapes lying nearby.

In order to compare the accuracy of both methods, the estimated WBC area, which has been approximated by both approaches, is compared to the actual WBC size considering different degrees of evolution, that is, the cycle number for each algorithm. The comparison considers only one WBC because it is the only detected shape in Wang's method.

Table 5 shows the averaged results over twenty repetitions for each experiment. In order to enhance the analysis, Figure 13 illustrates the error percentage versus iterations evolution from an extended data set which has been compiled from Table 5.

7. Conclusions

In this paper, an algorithm for the automatic detection of blood cell images based on the DE algorithm has been presented. The approach considers the complete process as

TABLE 4: Comparative WBC detection among methods which considers the complete data set of 30 images corrupted by different levels of Gaussian noise.

Noise level	Method	Leukocytes detected	Missing	False alarms	DR	FAR
$\sigma = 5$ Gaussian noise 517 leukocytes	BSV [3]	214	303	98	41.39%	18.95%
	IO [4]	366	151	87	70.79%	16.83%
	Wang [5]	358	159	84	69.25%	16.25%
	GAB [12]	407	110	76	78.72%	14.70%
	DE-based	487	30	21	94.20%	4.06%
$\sigma = 10$ Gaussian noise 517 leukocytes	BSV [3]	162	355	129	31.33%	24.95%
	IO [4]	331	186	112	64.02%	21.66%
	Wang [5]	315	202	124	60.93%	23.98%
	GAB [12]	363	154	113	70.21%	21.86%
	DE-based	471	46	35	91.10%	6.77%

(a)   (b)
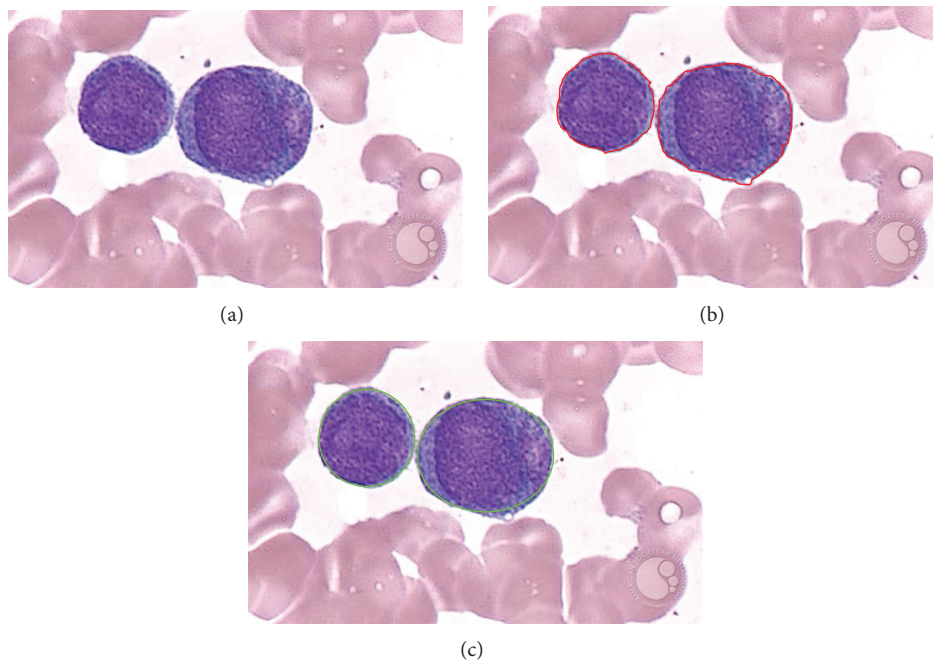
(c)

FIGURE 11: Comparison of the DE and Wang's method for white blood cell detection in medical images: (a) original image, (b) detection using the Wang's method, and (c) detection after applying the DE method.

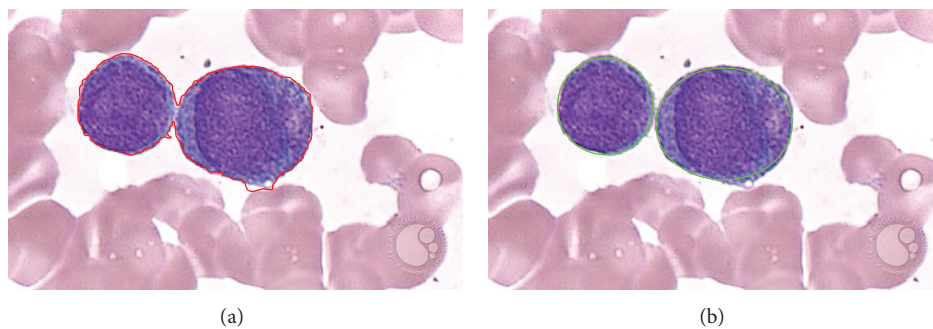(a)   (b)

FIGURE 12: Result comparison for the white blood cells detection showing (a) Wang's algorithm after 400 cycles and (b) DE detector method considering 1000 cycles.
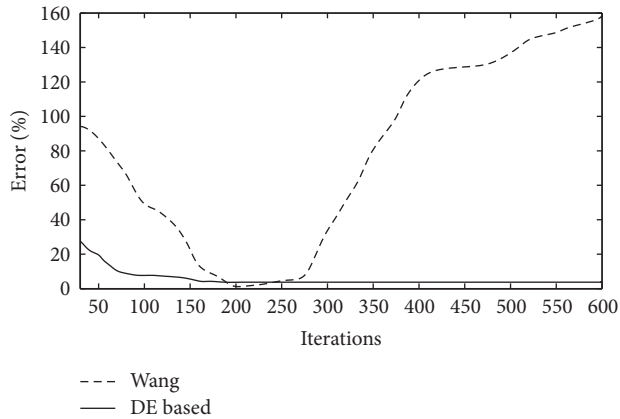
FIGURE 13: Error percentage versus iterations evolution from an extended data set from Table 5.

TABLE 5: Error in cell's size estimation after applying the DE algorithm and Wang's method to detect one leukocyte embedded in a blood-smear image. The error is averaged over twenty experiments.

Algorithm	Iterations	Error%
Wang	30	88%
	60	70%
	200	1%
	400	121%
	600	157%
DE-based	30	24.30%
	60	7.17%
	200	2.25%
	400	2.25%
	600	2.25%

a multiple ellipse detection problem. The proposed method uses the encoding of five edge points as candidate ellipses in the edge map of the smear. An objective function allows to accurately measure the resemblance of a candidate ellipse with an actual WBC on the image. Guided by the values of such objective function, the set of encoded candidate ellipses are evolved using the DE algorithm so that they can fit into actual WBC on the image. The approach generates a subpixel detector which can effectively identify leukocytes in real images.

The performance of the DE method has been compared to other existing WBC detectors (the boundary support vectors (BSV) approach [3], the iterative Otsu (IO) method [4], the Wang algorithm [5], and the genetic algorithm-based (GAB) detector [12]) considering several images which exhibit different complexity levels. Experimental results demonstrate the high performance of the proposed method in terms of detection accuracy, robustness and stability.

References

[1] X. Zhuang and Q. Meng, "Local fuzzy fractal dimension and its application in medical image processing," *Artificial Intelligence in Medicine*, vol. 32, no. 1, pp. 29–36, 2004.

[2] I. Scholl, T. Aach, T. M. Deserno, and T. Kuhlen, "Challenges of medical image processing," *Computer Science*, vol. 26, no. 1-2, pp. 5–13, 2011.

[3] M. Wang and R. Chu, "A novel white blood cell detection method based on boundary support vectors," in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC '09)*, pp. 2595–2598, Texas, Tex, USA, October 2009.

[4] J. Wu, P. Zeng, Y. Zhou, and C. Olivier, "A novel color image segmentation method and its application to white blood cell image analysis," in *Proceedings of the 8th International Conference on Signal Processing (ICSP '06)*, November 2006.

[5] W. Shitong, K. F. L. Chung, and F. Duan, "Applying the improved fuzzy cellular neural network IFCNN to white blood cell detection," *Neurocomputing*, vol. 70, no. 7–9, pp. 1348–1359, 2007.

[6] H. Muammar and M. Nixon, "Approaches to extending the Hough transform," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP '89)*, vol. 3, pp. 1556–1559, May 1989.

[7] T. Atherton and D. Kerbyson, "Using phase to represent radius in the coherent circle Hough transform," in *Proceedings of IEE Colloquium on Hough Transforms*, pp. 1–4, IEEE, 1993.

[8] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[9] D. Shaked, O. Yaron, and N. Kiryati, "Deriving stopping rules for the probabilistic Hough transform by sequential analysis," *Computer Vision and Image Understanding*, vol. 63, no. 3, pp. 512–526, 1996.

[10] L. Xu, E. Oja, and P. Kultanen, "A new curve detection method: randomized Hough transform (RHT)," *Pattern Recognition Letters*, vol. 11, no. 5, pp. 331–338, 1990.

[11] J. Han and L. Koczy, "Fuzzy Hough transform," in *Proceedings of the 2nd IEEE International Conference on Fuzzy Systems*, vol. 2, pp. 803–808, 1993.

[12] G. Karkavitsas and M. Rangoussi, "Object localization in medical images using genetic algorithms," *World Academy of Science, Engineering and Technology*, vol. 2, pp. 6–9, 2005.

[13] V. Ayala-Ramirez, C. H. Garcia-Capulin, A. Perez-Garcia, and R. E. Sanchez-Yanez, "Circle detection on images using genetic algorithms," *Pattern Recognition Letters*, vol. 27, no. 6, pp. 652–657, 2006.

[14] E. Lutton and P. Martinez, "A genetic algorithm for the detection of 2D geometric primitives in images," in *Proceedings of the 12th International Conference on Pattern Recognition*, vol. 1, pp. 526–528, Jerusalem, Israel, October 1994.

[15] J. Yao, N. Kharma, and P. Grogono, "A multi-population genetic algorithm for robust and fast ellipse detection," *Pattern Analysis and Applications*, vol. 8, no. 1-2, pp. 149–162, 2005.

[16] H. D. Cheng, Y. Guo, and Y. Zhang, "A novel Hough transform based on eliminating particle swarm optimization and its applications," *Pattern Recognition*, vol. 42, no. 9, pp. 1959–1969, 2009.

[17] DEM: Diffused expectation maximization function for image segmentation, Version 1.0, 2012, http://www.mathworks.com/matlabcentral/fileexchange/37197-dem-diffused-expectation-maximisation-for-image-segmentation.

[18] B. V. Babu and S. A. Munawar, "Differential evolution strategies for optimal design of shell-and-tube heat exchangers," *Chemical Engineering Science*, vol. 62, no. 14, pp. 3720–3739, 2007.

[19] D. G. Mayer, B. P. Kinghorn, and A. A. Archer, "Differential evolution—an easy and efficient evolutionary algorithm for model optimisation," *Agricultural Systems*, vol. 83, no. 3, pp. 315–328, 2005.

[20] S. Kannan, S. Mary Raja Slochanal, and N. Padhy, "Application and comparison of metaheuristic techniques to generation expansion planning problem," *IEEE Transactions on Power Systems*, vol. 20, no. 1, pp. 466–475, February 2005.

[21] J. P. Chiou, C. F. Chang, and C. T. Su, "Variable scaling hybrid differential evolution for solving network reconfiguration of distribution systems," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 668–674, 2005.

[22] E. Cuevas, D. Zaldivar, and M. Pérez-Cisneros, "A novel multi-threshold segmentation approach based on differential evolution optimization," *Expert Systems with Applications*, vol. 37, no. 7, pp. 5265–5271, 2010.

[23] R. Storn and K. Price, "Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces," Tech. Rep. TR-95-012, International Computer Science Institute, Berkeley, Calif, USA, 1995.

[24] J. E. Bresenham, "A linear algorithm for incremental digital display of circular arcs," *Communications of the ACM*, vol. 20, no. 2, pp. 100–106, 1977.

[25] J. R. Van Aken, "An efficient ellipse-drawing algorithm," *IEEE Computer Graphics and Applications*, vol. 4, no. 9, pp. 24–35, 1984.

[26] G. Boccignone, M. Ferraro, and P. Napoletano, "Diffused expectation maximisation for image segmentation," *Electronics Letters*, vol. 40, no. 18, pp. 1107–1108, 2004.

[27] G. Boccignone, P. Napoletano, V. Caggiano, and M. Ferraro, "A multi-resolution diffused expectation-maximization algorithm for medical image segmentation," *Computers in Biology and Medicine*, vol. 37, no. 1, pp. 83–96, 2007.

[28] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Addison-Wesley, Reading, Mass, USA, 1992.

[29] L. Wang and F. Z. Huang, "Parameter analysis based on stochastic model for differential evolution algorithm," *Applied Mathematics and Computation*, vol. 217, no. 7, pp. 3263–3273, 2010.

[30] G. Landi and E. Loli Piccolomini, "An efficient method for nonnegatively constrained Total Variation-based denoising of medical images corrupted by Poisson noise," *Computerized Medical Imaging and Graphics*, vol. 36, no. 1, pp. 38–46, 2012.

[31] M. J. Tapiovaara and R. F. Wagner, "SNR and noise measurements for medical imaging: I. A practical approach based on statistical decision theory," *Physics in Medicine and Biology*, vol. 38, no. 1, pp. 71–92, 1993.