Handbook of Research on Novel Soft Computing Intelligent Algorithms:

Theory and Practical Applications

Pandian M. Vasant
PETRONAS University of Technology, Maylaysia

Volume I

A volume in the Advances in Computational Intelligence and Robotics (ACIR) Book Series

Information Science REFERENCE
An Imprint of IGI Global

Managing Director:	Lindsay Johnston
Production Manager:	Jennifer Yoder
Publishing Systems Analyst:	Adrienne Freeland
Development Editor:	Allyson Gard
Assistant Acquisitions Editor:	Kayla Wolfe
Typesetter:	Christina Henning
Cover Design:	Jason Mull

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

Library of Congress Cataloging-in-Publication Data

Handbook of research on novel soft computing intelligent algorithms : theory and practical applications / Pandian Vasant, editor.
volumes cm
Includes bibliographical references and index.
Summary: “This book explores emerging technologies and best practices designed to effectively address concerns inherent in properly optimizing advanced systems, demonstrating applications in areas such as bio-engineering, space exploration, industrial informatics, information security, and nuclear and renewable energies”--Provided by publisher.
ISBN 978-1-4666-4450-2 (hardcover) -- ISBN 978-1-4666-4451-9 (ebook) -- ISBN 978-1-4666-4452-6 (print & perpetual access) 1. Soft computing--Industrial applications. 2. Intelligent control systems. 3. Intelligent agents (Computer software). I. Vasant, Pandian.
QA76.9.S63H35 2014
006.3--dc23
2013017606

This book is published in the IGI Global book series Advances in Computational Intelligence and Robotics (ACIR) (ISSN: 2327-0411; eISSN: 2327-042X)

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Chapter 12
Melanocytic Lesions Screening through Particle Swarm Optimization

Rustem Popa
"Dunarea de Jos" University of Galati, Romania

ABSTRACT

Early detection of malignant melanoma, which is the most dangerous skin cancer, significantly improves the chances of curing it. For this reason, dermatologists are looking for new methods for the examination of suspicious lesions that changes their shape over time. The author investigates in this chapter some algorithms which may be used for automated diagnosis of skin lesions. First algorithm performs the image segmentation by edge detection, which plays an important role in identifying borders of the lesion. Next algorithm uses the Particle Swarm Optimization (PSO) paradigm for recognizing the images of the same melanocytic nevus taken at different moments of time. The idea is that a novel view of an object can be recognized by simply matching it to combinations of known views of the same object. The main difficulty in implementing this idea is determining the parameters of the combination of views. The space of parameters is very large and we propose a PSO approach to search this space efficiently. The effectiveness of this approach is shown on a set of real images captured with a camera under different angles of view.

INTRODUCTION

Detection and early diagnosis of skin cancer remains the main concern of dermatologists worldwide. Malignant melanoma is now one of the most common forms of cancer among world's population, especially in fair-skinned individuals. Change of recreational behavior together with the increase in ultraviolet radiation cause a dramatic increase in the number of melanomas diagnosed. The curability of this type of skin cancer (about 70%) depends of early enough recognition and surgically treatment. Many publications report on isolated efforts into the direction of automated

DOI: 10.4018/978-1-4666-4450-2.ch012

melanoma recognition by image processing, but complete integrated dermatological image analysis systems are hardly found in clinical use (Gauster et al., 2001).

Małaczewska and Dabkowski (2004) talk about the contemporary view on the melanocytic nevi and their role in the pathogenesis of skin malignant melanoma. There is a strong relationship between the presence of the melanocytic nevi and the incidence of melanoma. For that reason dermatologists should pay close attention to patients from the risk group, with many common and atypical melanocytic nevi, family history of melanoma, bright fair skin, with history of sun burns. These patients should be meticulously and regularly checked up. Examination should include photographic surveillance and dermatoscopy and every suspected mole should be excised with further histological examination. This kind of procedure intensifies the possibility of early recognition of melanoma malignum of the skin, which is crucial for successful treatment of this dangerous disease.

Clinical features of melanoma are summarized as what's called ABCD rule, promoted by the America Cancer Society: A (Asymmetry), B (Border irregularity), C (Color variegation) and D (Diameter greater than 6mm). Early recognition of changes of lesion in terms of the previous features provides important diagnostic and prognostic information. Other screening guidelines are established by the seven-point checklist, advocated by a group of dermatologists from Glasgow. This checklist emphasizes the progression of the symptoms and consists of three major features (change in size, shape and color) and four minor features (inflammation, crusting or bleeding, sensory change, and diameter greater than 7mm). When any of the major features is detected in a melanocytic lesion, immediate help from health professionals is recommended. The presence of any minor features is advised to be monitored regularly (Lee, 2001; Liu et al., 2011). Schleicher et al. (2003) attached two more letters to the ABCD rule: E (Elevation) and I (Itch). Over time, most melanomas will become raised, and a very early signal that a mole is becoming cancerous is the sensation of itching at the site of the lesion.

Patients with large congenital melanocytic nevi are at increased risk for developing various medical problems, including cutaneous melanoma. In general, most studies reporting on the risk of melanoma in large congenital melanocytic nevi enrolled patients with lesions that were at least 20 cm in diameter, but there are no evidences between the risk and the absolute size of the lesion or its clinical appearance (flat, raised, rugous, speckled, etc). However, some recent data suggest that the risk for melanoma may be highest for lesions greater than 40 cm, located on the torso rather than on the head, neck or extremities (Slutsky et al., 2010). The method described in this chapter may be used also for this kind of lesions.

As a melanocytic lesion ages, both the skin where it is settled as well the nevi cells themselves, undergo alterations in their structure. An estimated percentage of 20 or 30 of lesions disappear in old age. This process of involution and disappearance is hard to prove in individual lesions, but there are proven cases based on old photographs where this had happened (Cintra et al., 1994). So, it's a good idea to follow the evolution of these lesions in time using photographs taken at different moments of time and different angles of view.

At this time dermatologists prefer to examine the suspect moles with the dermatoscope. Dermatoscopy significantly improves the sensitivity for melanoma detection compared to the naked-eye examination. Using dermatoscopy, melanoma may be detected before it displays the classical clinical features summarized in the ABCD rule. Dermatoscopy allows the detection of early melanoma-specific criteria that are visible under the dermatoscope even when the size of the tumor is less than 6mm, leading to a diagnosis at an earlier stage, when melanoma looks like a benign lesion. According with Moscarella et al. (2010), dermatoscopy has to be considered as

a first-level diagnostic tool. This is in contrast with one of the guiding rules of a few years ago, recommending dermatoscopy examination as a second-level screening tool for those lesions considered highly suspicious for melanoma after a first clinical evaluation. Some novel methods used in dermoscopy, helping to bring about a faster, more accurate diagnostics of those lesions which have proven to be more difficult to recognize are discussed in Kaminska-Winciorek and Spiewak (2012). Although the images discussed in this chapter were taken with a camera in visible light, the algorithms discussed here can also be used for dermatoscopic images.

First step in the processing of an image that represents a mole consists in its segmentation. Image segmentation plays an important role in identifying borders of the lesion because accurate description and measurement of image features cannot be achieved without accurate image segmentation. A wide range of algorithms have been used for image segmentation, broadly categorized as pixel-based segmentation (Guo and Aslandogan, 2003; Tzekis et al., 2009), region-based segmentation (van Kaick et al., 2007) and edge detection (Xu and Kasparis, 2003; Taouil et al., 2006). We have proposed an algorithm based on an idea discussed in Tzekis et al. (2009). A complete description of this algorithm is given in the section 3.

Then we built a PSO algorithm to determine whether two different images of the same mole are similar. We have focused on the detection of any modification in the border irregularity of the lesion, by comparing two different images: one of them is an older reference view of the lesion, and the other one is a more recent view of the same lesion. If the novel view of the lesion is able to match the reference view, modified by a 2D affine transformation eventually, then the size and shape of the lesion is not modified. A PSO algorithm searches the parameters of the transformation in an attempt to find as more as matches between some significant points of the first border shape and the contour of the second shape. These kind of algorithms are inspired by the social foraging behavior of some animals such as flocking behavior of birds and the schooling behavior of fish. Particles in the swarm fly through an environment following the fitter members of the swarm and generally biasing their movement toward historically good areas of their environment (Kennedy et al., 2001; Omran, 2004; Poli et al., 2007; Banks et al., 2008).

Other publications report a variety of diverse methods into the direction of automated melanoma recognition by image processing. Korotkov and Garcia (2012) have done a remarkable study on the current state in computer diagnosis of skin lesions. Schmid-Saugeon (2000) investigated the detection of symmetry axes of the lesion through the optimization of a given symmetry measure, computed as a function of the mean-square error between the original and reflected images. His method uses a genetic algorithm and an optimization scheme derived from the self-organizing maps theory. Wallace et al. (2000) used an Artificial Neural Network (ANN) for the classification of optical reflectance spectra (320 to 1100 nm) from malignant melanoma and benign nevi. Also an ANN is used for the classification of melanocytic lesions in Romdhane et al., (2007). A comprehensive classification of melanocytic lesions, with the purpose of differential diagnosis with melanoma, is given in Strungs (2004). Rallan et al. (2006) proposes a combination of high-resolution ultrasound reflex transmission imaging (about 20 MHz) and white light digital photography data for the classification of a variety of pigmented lesions. Finally, Madooei et al. (2012) proposed a method of skin lesion malignancy detection using intrinsic melanin and hemoglobin color components from dermatoscopic images.

The rest of this chapter is structured as follows: next section presents an overview of the theory of Algebraic Functions of Views. This concept provides a powerful foundation for tackling variations in the appearance of an object's shape due to viewpoint changes. Image segmentation

and the methodology used to estimate the best values for the parameters of the combination scheme, including the search based on PSO, are described in Section 3. Our experimental results are presented in Section 4, and finally, future research directions, discussion, and concluding remarks are presented in the next three sections.

BACKGROUND ON ALGEBRAIC FUNCTIONS OF VIEWS

A short introduction in the theory of Algebraic Function of Views is presented in Bebis et al. (1998) and Bebis et al. (2002). According to this theory, the variety of two-dimensional views depicting an 2D or 3D object can be expressed as a combination of a small number of 2D views of the object. In the case of 2D objects, one aspect per object is enough, while in the case of general 3D objects, more aspects are necessary to represent the object from different viewing directions (Bebis et al., 1998; Bebis and Bourbakis, 2004). In this case of 2D objects, the combination scheme is equivalent to an affine transformation of the image coordinates from the known view.

We assume that each aspect is represented by *V* different views which we call reference views. For each aspect, we assume a number of "interest" points *N* (e.g., corners, junctions, etc.), which are common in all the views associated with the aspect. We also assume that the point correspondences across the views have been established (Bebis et al., 1998).

It was shown in Ullman and Basri (1991) that if we let an object undergo 3D rigid transformations, (i.e., rotations and translations in space), and we assume that the images of an object are obtained by orthographic projection followed by a uniform scaling, then any novel view of an object can be expressed as a linear combination of three other views of the same object. If we have three corresponding points, with coordinates $\left(x',y'\right)$, $\left(x'',y''\right)$ and $\left(x''',y'''\right)$, one from each reference view of the object, then the coordinates $\left(x,y\right)$ of the corresponding point from a novel view of the same object, obtained by applying a different rigid transformation is as the following:

$$x = a_1x' + a_2x'' + a_3x''' + a_4 \quad (1)$$

$$y = b_1y' + b_2y'' + b_3y''' + b_4 \quad (2)$$

where the parameters $a_i, b_i,\ i = \overline{1,4}$ are the same for all the points that are in correspondence across the four views. These parameters can be recovered by solving a linear system of equations, given that we know at least four point correspondences across the views. Although it was not explicitly discussed in the literature, Bebis et al. (2002) shows that algebraic functions of views also exist in the case of 2D objects, and is sufficient a single reference view. This is because in the case of planar objects, scaled orthographic projection is equivalent to a 2D affine transformation. Given a novel view and a point with coordinates $\left(x,y\right)$ that is in correspondence with the point with coordinates $\left(x',y'\right)$ in reference view, then the coordinates of the corresponding point from the novel view are:

$$x = a_1x' + a_2y' + a_3 \quad (3)$$

$$y = b_1x' + b_2y' + b_3 \quad (4)$$

In this chapter, we have used Equations (3) and (4) for a number of *N* selected points from the border of the first lesion. Given the point correspondences across the reference and new view, the following system of equations should be satisfied:

$$\begin{bmatrix} x_1' & y_1' & 1 \\ x_2' & y_2' & 1 \\ \dots & \dots & \dots \\ x_N' & y_N' & 1 \end{bmatrix} \cdot \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix} = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \dots & \dots \\ x_N & y_N \end{bmatrix} \quad (5)$$

where (x_1', y_1'), (x_2', y_2'), … (x_N', y_N') are the coordinates of the points in the reference image, and (x_1, y_1), (x_2, y_2), … (x_N, y_N) are the coordinates of the points in the new view.

A very important problem consists in determining the range of values for $a_i, b_i,\ i = 1,2,3$. Splitting the prior system of equations into two subsystems, one involving the a_i parameters and one involving the b_i parameters into two subsystems, we have:

$$P \cdot c_1 = p_x \quad (6)$$

$$P \cdot c_2 = p_y \quad (7)$$

where the columns of the *P* matrix are shown in (5), c_1 and c_2 are vectors corresponding to the parameters of the combination scheme a_i and b_i, and p_x, p_y are vectors corresponding to the *x* and *y* coordinates of the new view.

According with Bebis et al. (1998) and Bebis et al. (2002), both Equations (6) and (7) can be solved using a least-squares approach such as Singular Value Decomposition (SVD). The main steps of this method are presented next. Using SVD, the *P* matrix can be factorized as $P = U_P \cdot W_P \cdot V_P^T$, where both U_P and V_P are orthonormal matrices, while W_P is a diagonal matrix whose elements w_{ii}^P are always nonnegative. The solutions of the previous two systems are $c_1 = P^+ \cdot p_x$ and $c_2 = P^+ \cdot p_y$, where P^+ is the pseudoinverse of *P*. Assuming that *P* has been factorized, its pseudoinverse is $P^+ = V_P \cdot W_P^+ \cdot U_P^T$, where W_P^+ is also a diagonal matrix with elements $1/w_{ii}^P$ if w_{ii}^P is greater than zero and zero otherwise. The solutions of (6) and (7) are given by the following equations:

$$c_1 = \sum_{i=1}^{k} \left(\frac{u_i^P \cdot p_x}{w_{ii}^P} \right) \cdot v_i^P \quad (8)$$

$$c_2 = \sum_{i=1}^{k} \left(\frac{u_i^P \cdot p_y}{w_{ii}^P} \right) \cdot v_i^P \quad (9)$$

where u_i^P is the *i*-th column of matrix U_P, v_i^P is the *i*-th column of matrix V_P, and *k* = 3. To determine the range of values for c_1 and c_2, we assume first that the novel views has been scaled such that the *x* and *y* coordinates belong within a specific interval. This scaling can be done, e.g., by mapping the views to the unit square. In this way, its *x* and *y* image coordinates will be mapped to the interval [0,1]. To determine the range of values for c_1 and c_2, we need to consider all possible solutions of (6) and (7), assuming that p_x and p_y belong to [0,1]. This problem is solved by Interval Analysis in Bebis et al. (1998). It should be mentioned that since the matrix *P* and the intervals for p_x, p_y are all the same, the interval solutions for c_1^I and c_2^I will be the same (the superscript *I* denotes an interval vector).

Significant research has been performed in the area of interval linear systems. In general, the matrix of a system of interval equations is also an interval matrix, that is, a matrix whose components are interval variables. In our case, things are simpler since the elements of *P* are the *x* and *y* coordinates of the reference view of the object which are always fixed. When interval solutions are computed, not every solution in c_1^I and c_2^I corresponds to p_x and p_y that belong to p_x^I and p_y^I. If new views are generated by choosing the values for the parameters of the algebraic functions from the interval solutions obtained, then some of the generated views might not lie completely within the unit square. In this case these

views are regarded as "invalid views" and they can be rejected easily by testing whether their coordinates are within the unit square.

An interval solution is called "sharp" if it does not contain many invalid solutions. Within our context, it is important to compute sharp interval solutions since this will save time and space. However, if we apply the interval arithmetic operators on Equations (8) and (9), it is very likely that we will obtain solutions that will not be very sharp. Bebis et al. (1998) discusses the main factors that affect the sharpness of interval solutions. Equations (8) and (9) are slightly modified and each component of c_1^I and c_2^I may be obtained by applying the interval arithmetic operations on the new rewritten equations. Since both (8) and (9) involve the same matrix *P* and p_x^I and p_y^I assume values from the same interval, the interval solutions c_1^I and c_2^I will be the same.

Bebis et al. (1998) describes also a method for "preconditioning" the reference views in order to consider ways to compute narrower ranges. Preconditioning implies a transformation of the original reference views to new reference views, yielding very narrow ranges. Different experiments have demonstrated the useful of this transformation and the computed ranges using the preconditioned views. A model used in Bebis et al. (1998) delivered the values between [-0.454, 0.454] for a_1, b_1, [-0.392, 0.392] for a_2, b_2, and [0.0, 1.0] for a_3, b_3. Other model used in Bebis et al. (2002) established other values, very closely from the previous range: [-0.408, 0.408] for a_1, b_1, [-0.391, 0.391] for a_2, b_2, and [0.0, 1.0] for a_3, b_3. Our experiments in this chapter have been designed without any preconditioning of the parameters of affine transformation, and this aspect will be discussed in the next sections.

COMPARISON OF THE TWO IMAGES OF THE MOLE

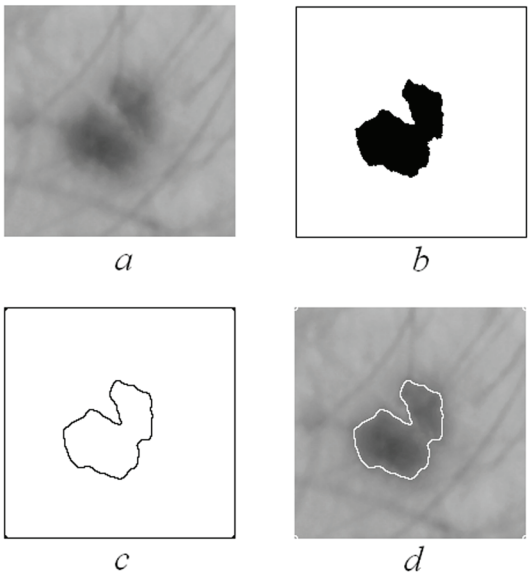
This section discusses two algorithms that establish if the investigated melanocytic lesion has modified its shape in time. Two different views of the lesion, acquired at different moments of time, are compared between them. First algorithm detects the border of the lesion, using an image segmentation by edge detection, and the other estimates the best values for the parameters of the combination scheme between two images, using the search based on a PSO algorithm.

Automatic Detection of Skin Lesion Border

We have used a simple algorithm of border detection for the explored mole, using an idea from Tzekis et al. (2009). First we must determine whether a point belongs to the lesion, or not. Different images have different colors and contrasts. For this reason, we should find a threshold color between the color of the skin and the color of the melanocytic lesion. This color can be defined by estimating the mean color of the image, that is taking into account all pixels in the image. In order to minimize the complexity of this process, the Monte Carlo method is applied with *k* points, resulting in a good approximation of the base color. The algorithm is briefly described as follows:

1. Convert RGB color image into a gray level image, *I* (See Figure 1.a).
2. Select *k* random pixels in the image.
3. Calculate *T*, the value of threshold, as a weighted average between th*e* values of selected pixels.
4. Generate a binary image, *B*, by comparing each pixel value in *I* with *T* (See Figure 1.b):

Figure 1. Border detection of a melanocytic lesion



$$B(n)=\begin{cases}0 & \text{if } I(n)\le T\\ 1 & \text{if } I(n)>T\end{cases}$$

5. Apply a *dilate* filter, *D*, which expands the size of the object (0-valued pixels).
6. Calculate the difference between the binary image *B*, and the dilated version *D*: *Boundary = NOT{XOR[B,D]}* (See Figure 1.c).
7. Combine *the original gray level* image *I*, with the *Boundary* (See Figure 1.d):

$$Final(n)=\begin{cases}I(n) & \text{if } Boundary(n)=1\\ 255 & \text{if } Boundary(n)=0\end{cases}$$

In the case from Figure 1.a, we have chosen an image of a melanocytic lesion represented on 200 x 200 pixels, and $k = 100$. The values of the gray pixels in the image are between 0 and 255, and threshold was established by calculating the average value of the selected pixels, multiplied by a scale factor of 0.8, that is $T = 126$. This value is of critical importance, since it controls the particular abstraction of information that is obtained. Indeed, different thresholds can produce different valuable abstractions of the image. In our case, we are interested in image segmentation, and the lesion on the skin is a dark object on a bright background, like in Figure 1.b. Our image contain an object and background with significantly different average brightness, that is, it has associated a bimodal histogram, and thresholding is a very effective technique. We apply then to the image in Figure 1.b a morphological filter of dilation, with a square window of 3 pixels, which expands the size of the dark object. The process of dilation also smoothes the boundaries of objects, removing gaps or bays of too narrow width. The "difference" between the binary image in Figure 1.b and the dilated version of it is illustrated in Figure 1.c. If window is small, then the difference between the image and its dilated version is not too large, and the object border is effectively detected. Finally, Figure 1.d represents a combination between the original image of the lesion and its border (Bovik, 2009).

The presence of hair may result in a different lesion border than the actual one. In Tzekis et al. (2009), a simple algorithm is employed to remove the hair that appears to cut the borderline. For every *n*-th point of the defined border, the next 15 points on the border are cheked, one by one. If the Euclidian distance value between any of these *m*-th points and the *n*-th point is less than 3, then all points from the (*n*+1)-th point to the *m*-th one are removed.

Other good software algorithm, called Dull-Razor, is used in Lee (2001). This program consists of three basic steps: identifying the dark hair locations, replacing the hair pixels with the nearby non-hair pixels, and smoothing the final result. In this chapter, we have not dealt with this problem.

Finding the Transformation Parameters using PSO Search

The goal of the PSO algorithm is to have all the particles locate the optima in the search space. This is achieved by assigning initially random positions to all particles in the space and small initial random velocities. The algorithm is executed like a simulation, advancing the position of each particle in turn based on its velocity, the best known global position in the problem space and the best position known to a particle. The objective function is sampled after each position update. Over time, through a combination of exploration and exploitation of known good positions in the search space, the particles cluster converge together around an optima.

The PSO algorithm is comprised of a collection of particles that move around the search space influenced by their own best past location and the best past location of the whole swarm or a close neighbor. Each iteration a particle's velocity is updated using:

$$v_i\left(t+1\right) = v_i\left(t\right) + coef_1 \cdot \left(pbest_i - p_i\left(t\right)\right) + coef_2 \cdot \left(p_g best - p_i\left(t\right)\right) \quad (10)$$

where $v_i\ (t+1)$ is the new velocity for the i-th particle, $coef_1$ and $coef_2$ are the weighting coefficients for the personal best and global best positions respectively, $p_i(t)$ is the i-th particle position at time t, $pbest_i$ is the i-th particle best known position, and $p_g best$ is the best position known to the swarm. Each iteration a particle's position is updated using:

$$p_i\left(t+1\right) = p_i\left(t\right) + v_i\left(t\right) \quad (11)$$

where $p_i\ (t+1)$ is the new position for the i-th particle at time $t+1$, $p_i(t)$ is the i-th particle position at time t, and $v_i\ (t)$ is the velocity for the i-th particle at time t (Kennedy et al., 2001). Some authors, like Brownlee (2011), recommend multiplication of the coefficients $coef_1$ and $coef_2$ with a random number between 0 and 1. In this way, the share between exploration and exploitation of the search space changes randomly with each iteration.

Each particle in the swarm represents a possible solution for our problem. It contains six numbers which are the six parameters a_i, b_i, (i=1, 2, 3) of the affine transformation (See Equation 5). Our goal is to find a transformation which would bring a large number of model points into alignment with the scene. We consider the reference view as a model, and the novel view as a scene. The fitness of individuals is estimated by computing the back-projection error between the model and the scene. To compute this error, for every model point, we find the closest scene point and then compute the distance between these two points. The overall back-projection error, that is the total distance in pixels, is the sum of all these partial distances. The algorithm is briefly described as follows:

1. Generate randomly initial population of particles, $p_i(t=0)$, $i = 1,\ldots,n$.
2. Set $v_i(t=0) = 0$, $i = 1,\ldots,n$.
3. **Repeat**
4. For each particle, evaluate the fitness, that is a distance function between points.
5. Compare particle's fitness evaluation with it's $pbest_i$. If current value is better than $pbest_i$, then set $pbest_i$ equal to the current value.
6. Identify the particle in the neighborhood with the best success so far, and assign its index to the variable g, $p_g best$.
7. Change the velocity and position of the particle according with equations (10) and (11).
8. **Until** termination criterion is met.

The termination criterion may be sufficiently good fitness or a maximum number of iterations. We tried different metrics for distance function

between the two sets of points. One of them was the Hausdorff distance, defined as the maximum distance of a set to the nearest point in the other set. It seems that best results were obtained with this metric. Then we considered an average distance between each two points from the model and the scene. We also took into account the difference between the centroids of the two sets of points, but it depends largely on the shapes of the two contours. Finally, we considered the total amount of distance between each point of a set and all other points in the other set. The modified Hausdorff distance was also taken into account in our experiments, but this metric was not used in further comparisons (Spyridonos et al., 2013).

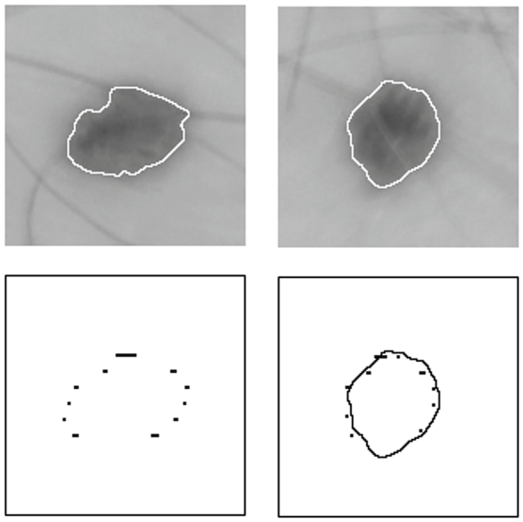
As we can see in the previous description of the algorithm, the velocity for the *i*-th particle at time 0, that is $v_i(0)$, is equal to zero. This value is modified in time using Equation (10), and it contributes to the new position of particle using Equation (11). This algorithm was implemented in Matlab, and its pseudocode with complete Matlab code are given in the appendix. Other PSO algorithms implemented in Matlab are cited in Khan et al. (2012), Ebbesen et al. (2012), and Vis (2009).

Figure 2 presents two different views of the same mole, and an example of matching between some points on these two contours, using the PSO algorithm previously described. The selection of the parameters of the algorithm will be discussed in the next section. In Popa and Aiordachioaie (2004) we solved the same problem with a standard Genetic Algorithm (GA). It seems that PSO performes usually better than GA, and it is also faster and more robust.

EXPERIMENTS AND RESULTS

We have selected in the model from Figure 2 a number of $N = 25$ representative points. These points are matched with the second contour, and, as we can see in Figure 3, the maximum back-

Figure 2. An example of matching between the model and the scene of the same melanocytic lesion using a PSO algorithm for searching of the parameters of transformation.

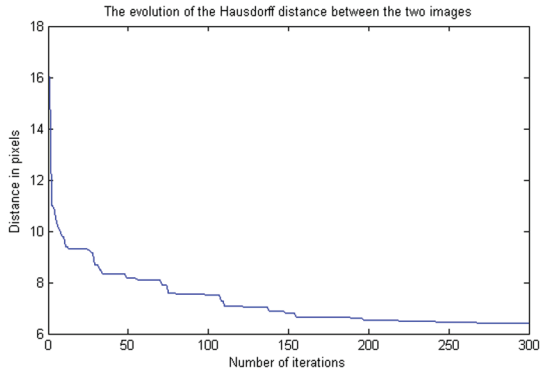


projection error between the model and the scene, after an evolution of 300 iterations, is less than 6 pixels, if we consider the Hausdorff distance. Evolution shown in Figure 4 is an average acquired after 10 successive runs of the PSO algorithm. Each of the two images representing the model and the scene has a resolution of 150×150 pixels. The size of the population is often set empirically on the basis of the dimensionality and perceived difficulty of a problem. Values in the range 20–50 are quite common, and we chosed 20 particles in population. Our previous experiments with GAs show that the best results were obtained with small populations (between 50 and 100 chromosomes) and sufficiently large number of generations (Popa and Aiordachioaie, 2004).

The parameters $coef_1$ and $coef_2$ determine the magnitude of the random forces in the direction of best positions at current iteration. These are often called acceleration coefficients, and their values are usually 2. Each time when a particle's velocity is updated, the new value of velocity mostly depends either by its own best current

Figure 3. Evolution of the best particle in the swarm, that is the minimum distance between the two views of the same lesion, given in Figure 2

Figure 4. Evolution of the best particle in the swarm, that is the minimum distance between the two views of the same lesion, averaged in 10 successive runs of the algorithm



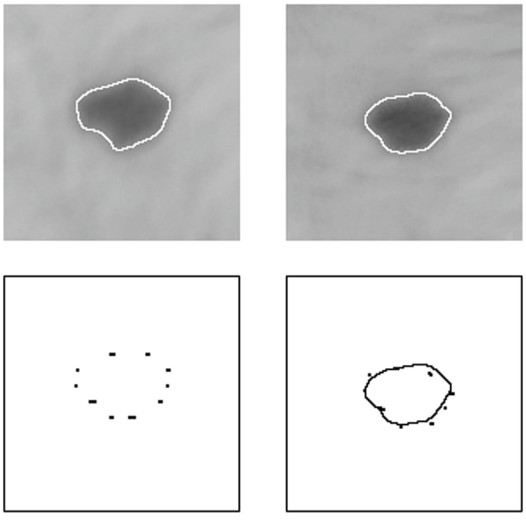
location, either by the best current location of the whole swarm, because of multiplication of these coefficients with different random numbers between 0 and 1. Velocity v_i (t) is kept within the range [$-V_{max}$, $+V_{max}$], and the value of V_{max} is 0.5 in our case. The choice of the parameter V_{max} required some care since it appeared to influence the balance between exploration and exploitation (Kennedy et al., 2001; Poli et al., 2007).

Like GAs, PSO is a stochastic algorithm, so two consecutive runs of the same algorithm with the same parameters could provide slight different results. Conclusions can be drawn only after several successive runs of the algorithm in the same conditions. Evolutions represented in Figures 3 and 4 show a similar behavior of the algorithm for several successive runs. PSO algorithm is fast and robust, the change of distance in time retains its shape, even if we change the algorithm parameters.

In Figure 5, we have presented another two different images of the same lesion as in Figure 2. The angles of views are completely different than in the first case. This fact explains maybe some errors in the process of matching between the model and the scene. However, it's easy to see that the approximate matching is a precious indication that all these images are obtained from the same lesion.

The main parameters of the PSO algorithm used in our experiments were previously mentioned: 20 particles in population, $coef_1$ and $coef_2$ are both equal to 2, V_{max} is 0.5, and the number of iterations is 300. Hausdorff distance was the preferred metric for the comparison of different

Figure 5. Another example of matching between other two different images of the same lesion



solutions in population. Table 1 illustrates some results obtained by comparing the distance between the two sets of points for the same lesion. As we see from the table, the number of particles in population and the number of iterations do not seem to be so important. The value of V_{max} would be desirable to be relatively small, and the values of the two coefficients do not seem to influence much the distance. PSO algorithm appears to be more robust than equivalent GA.

In Table 2 we presented different metrics for various lesions. Although the centroids of sets of points seem to provide minimum distance, it depends largely on the shapes of the two contours. We prefered to use Hausdorff distance in our experiments. Other considered metrics were the average distance between each two points from the model and the scene, and the total amount of distance between each point of a set and all other points in the other set.

Images from the Figure 6 represent a new lesion, with more irregulated borders, taken under different angles of views. These images are represented on 500 x 500 pixels. As we can see, the matching between the model and the scene seems to be more difficult, even if we extended the number of iterations to 1000. The points are smaller than in other figures because of the

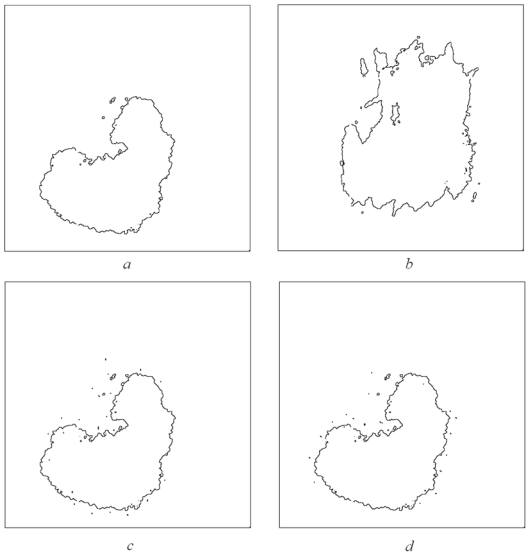
Table 1. Different Hausdorff distances between two scenes of the same lesion, for different parameters of the PSO algorithm

Parameters of the PSO algorithm					Hausdorff distance for two scenes of a mole
population	**coef 1**	**coef 2**	**V max**	**iterations**	
20	2	2	1	100	7.9893
40	2	2	1	100	8.5351
20	2	2	0.5	300	6.6914
20	2	2	0.5	1000	6.1193
20	1	10	0.5	1000	5.7458
20	10	1	0.5	1000	5.4951
20	2	2	2	1000	8.2364
20	2	2	10	1000	9.4049
20	2	2	0.1	1000	4.6964
40	2	2	0.5	1000	5.9267

Table 2. Different definitions for distances between two scenes of different lesions

Comparison between two scenes of the moles	Distance between two scenes of the moles			
	Hausdorff	**centroids**	**two points**	**all points**
same mole (Figure 2)	3.3792	0.9534	23.2186	280.6208
different moles_1	7.4174	0.5948	32.0553	577.8672
different moles_2	4.8806	0.7098	26.1206	427.8653
different moles_3	8.3674	0.6698	35.5425	842.4174
different moles_4	10.4628	0.7220	38.6206	777.9434
different moles_5	14.7869	0.8871	39.5031	954.2186

Figure 6. Another example of matching between other two different images of the same lesion



higher resolution of the images, but we see as they try to follow the outline of the image. They follow pretty well the border of the lesion, even if they are not evenly distributed throughout its length.

In Figure 7, we showed the matching of the model in modified scenes, corresponding to segmented images of the different lesions. The left image presents a modified scene by increasing the surface of the lesion with about 10% of the total area. Although some points are placed on the contour, we are not able to find transformation parameters to match exactly the set of points with the lesion border. The right image represents the contour of a different lesion, which can not be matched with the original model. In both cases the matching is not possible, and so, the decision is that these two lesions are completely different from the model discussed. Obviously, in a melanoma screening program, we are interested on the evolution of suspicious lesions in time and the results seem to be helpful for clinical diagnostic.

Color variegation of the lesion, that is the C component from the ABCD rule, may introduce difficulties in detecting lesion borders. Melanocytic lesion shown in Figure 8, represented by two distinct images, has some areas colored in different shades of brown. Randomly choosing a number of k points in the image, and then calculating the average of their values for determining the threshold necessary for generating binary image, can lead to the case from Figure 8, where lighter areas are not considered. Moreover, the variation of brightness or camera position can generate very different borders and applying of the PSO matching algorithm is no longer possible. To avoid this, we can choose a different threshold, possibly by choosing k points only from the background, which has a lighter color. However, errors are still possible, and this aspect remains to be investigated in future studies.

Figure 7. The scenes differ from the model given in Figure 5. The surface of the lesion in the left is greater than in the original model, and the right image represents a completely different lesion.

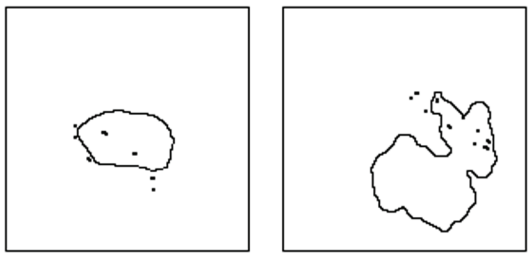
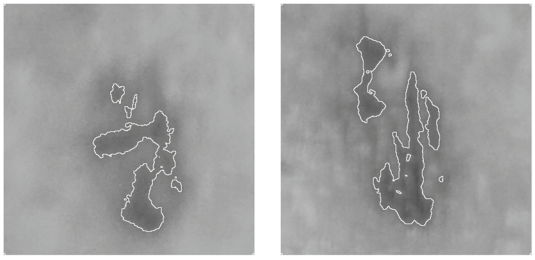


Figure 8. Two different images of the same melanocytic lesion with different colors. Border detection by image segmentation is not relevant and further comparison is not possible.



FUTURE RESEARCH DIRECTIONS

We have observed in our experiments that, even if the back-projection error is not so small in some cases, the shape of the model looks like the shape of the scene, if we discuss about the same lesion. If the borders of the lesion are modified, we must easy observe this modification. Near-exact matches are useful in the sense that can actually reduce the search space of the parameters to a limited domain. Then, a local optimization technique can be used for finding an exact match. The preliminary stage of rough alignment may help preventing such methods from reaching a local minimum instead of the global one.

The most important step in this process of recognition is automatic segmentation of the images. This task is not trivial, because the software approach to hair removal affects some pixel values underneath the hairs, and these values cannot be reconstructed accurately by a single view. On the other hand, some lesions have fuzzy borders, and these borders may depend on angle of view, light of the scene, the distance between the camera and the skin, and so on. Our segmentation algorithm introduces some errors, but we think that they are not very important for our results. For future research, we plan to solve better the problem of segmentation, to increase the number of the points on the contour and to establish some useful rules for the construction of a robust algorithm. A robust method to design properly the parameters of PSO algorithm should be found. Also, the ranges of values of affine transform parameters are very important for a successful transformation.

Some other ideas may be used in the future research. A better distance function may be defined, the number of the points on the contour may be increased, although more points may introduce new difficulties in matching with the contour, or other properties of the lesion may be taken into account, like analyzing surface morphology of the lesion using a three dimensional (3D) imaging technique (Tosca et al., 2013).

In Popa and Aiordachioaie (2004) we solved a similar problem with a standard GA, and experiments showed that usually PSO performed better. Hybrid approaches combining PSO and GA were proposed by some researchers. Their idea was to take the population of one algorithm when it has made no fitness improvement and using it as the starting population for the other algorithm. Two versions were proposed, GA-PSO and PSO-GA, and both of them performed slightly better than PSO (Poli et al., 2007).

DISCUSSION

There are two major problems in researching done so far. On the one hand, there is no connection between the selected points from the two contours. In Bebis et al. (2002) the authors have assumed the existence of the correspondence between the points of the two contours. Therefore it is necessary to introduce a preprocessing stage that would make the correspondence between the two sets of points. Because the image of the scene may become deformed or rotate, finding the correspondence between points is not a trivial problem.

On the other hand, also in Bebis et al. (2002) shows that the affine transformation parameters must lie within a certain range of values. We observed in our experiments that limitation of the search space does not offer any advantage, so we did not use recommended limitations. It is still possible that taking into account the recommended ranges for the variation of the parameters will produce better results in the future.

Particle Swarm Optimization (PSO) is a relatively recent heuristic search method that is based on the idea of collaborative behavior and swarming in biological populations. PSO is similar to the Genetic Algorithm (GA) in the sense that they are both population-based search approaches and that they both depend on information sharing among their population members to enhance their search processes using a combination of deterministic

and probabilistic rules. Conversely, the GA is a well established algorithm with many versions and many applications.

We have shown in this research the hypothesis that states that although PSO and the GA on average yield the same effectiveness (solution quality), PSO is more computationally efficient (uses less number of function evaluations) than the GA. Further analysis shows that the difference in computational effort between PSO and the GA is problem dependent. It appears that "PSO outperforms the GA with a larger differential in computational efficiency when used to solve unconstrained nonlinear problems with continuous design variables and less efficiency differential when applied to constrained nonlinear problems with continuous or discrete design variables" (Hassan et al., 2004; Somasundaram and Muthuselvan, 2010). A good comparison between Evolution Strategies (ESs) and PSO may be found in Vis (2009). An interesting hybrid approach which combines a PSO algorithm with a Gravitational Search Algorithm (GSA) is proposed in Mirjalili and Mohd Hashim, (2010).

In their paper from 2007 published in the journal Swarm Intelligence, Poli et al. (2007) cited from the book of Kennedy et al. (2001) and their final conclusion remain essentially the same: "… particle swarm optimization is exponentially growing. So, clearly, *we are still looking at a paradigm in its youth, full of potential and fertile with new ideas and new perspectives*. Researchers in many countries are experimenting with particle swarms and applying them in real-world applications. Many more questions have been asked, although still too few have been satisfactorily answered, so the quest goes on." (Poli et al., 2007).

CONCLUSION

In this paper, we considered using a PSO algorithm to recognize some contours of the skin lesions. This idea is useful in the screening of the suspicious skin lesions. Two different images of the same lesion are taken at different moments of time, under different angles of view, and we must decide if the borders of the lesion are modified or not. The recognition strategy used was based on the theory of Algebraic Functions of Views. Our goal was only the matching between the model and the scene. It was discussed also an algorithm for image segmentation, in order to detect the contour of the lesion, and the problem of removing the hair in the image have been solved in another works.

Provided experiments on different lesions, with different parameters of PSO algorithm, show that this idea is sound in order to compare the shapes of different lesions taken at different moments of time. The definitions of distances or their values are not very important. Much more important seems to be the degree of matching between the points of the model and the contour of the scene. If the points are placed near the contour of the lesion, with the same shape, then is plausible that we have two different images of the same lesion. In opposing case, we have different lesions, or the old lesion has now a modified contour, and it should be studied with more attention.

REFERENCES

Banks, A., Vincent, J., & Anyakoha, C. (2008). A review of particle swarm optimization. Part I: Background and development. *Natural Computing*, *6*(4), 467–484. doi:10.1007/s11047-007-9049-5.

Bebis, G., Georgiopoulos, M., Shah, M., & da Vitoria Lobo, N. (1998). indexing based on algebraic functions of views. *Computer Vision and Image Understanding*, *72*(3), 360–378. doi:10.1006/cviu.1998.0679.

Bebis, G., Louis, S., Varol, Y., & Yfantis, A. (2002). Genetic object recognition using combina-tions of views. *IEEE Transactions on Evolutionary Computation*, *6*(2), 132–146. doi:10.1109/4235.996013.

Bebis, G., & Bourbakis, N. G. (2004). Integrating algebraic functions of views with indexing and learning for 3D object recognition. In *Proceedings of Computer Vision and Pattern Recognition Workshop* (CVPRW'04). IEEE Press. doi: 10.1109/CVPR.2004.95.

Bovik, A. (2009). *The essential guide to image processing*. Burlington, MA: Academic Press Publisher (an imprint of Elsevier).

Brownlee, J. (2011). *Clever algorithms. Nature-inspired programming recipes*. Australia: LuLu.

Cintra, M. L., de Souza, E. M., & Fernandez, F. M. (1994). Complete regression of melanocytic nevi: Clues for proper diagnosis. *Sao Paulo Medical Journal/RPM, 112*(3), 597-601.

Ebbesen, S., Kiwitz, P., & Guzzella, L. (2012). A generic particle swarm optimization matlab function. *2012 American Control Conference*. Montréal: INSPEC.

Ganster, H., Pinz, A., Rohrer, R., Wilding, E., Binder, M., & Kittler, H. (2001). Automated melanoma recognition. *IEEE Transactions on Medical Images*, *20*(3), 233–239. doi:10.1109/42.918473 PMID:11341712.

Guo, W., & Aslandogan, Y. (2003). Mining skin lesion images with spatial data mining methods. *Technical Report CSE-2003-20.* Arlington, TX: Department of Computer Science and Engineering, University of Texas at Arlington. Retrieved from http://cse.uta.edu/research/Publications/Downloads/CSE-2003-20.pdf

Hassan, R., Cohanim, B., de Weck, O., & Venter, G. (2004). *A comparison of particle swarm optimization and the genetic algorithm.* Reston, VA: AIAA. Retrieved from http://web.mit.edu/deweck/www/PDF_archive/3 Refereed Conference/3_50_AIAA-2005-1897.pdf.

Van Kaick, O., Hamameh, G., Zhang, H., & Wighton, P. (2007). Contour correspondence via ant colony optimization. In *Proceedings of 15th Pacific Conference on Computer Graphics and Applications*. Maui, HA: IEEE Press. doi: 10.1109/PG.2007.56.

Kaminska-Winciorek, G., & Spiewak, R. (2012). Tips and tricks in the dermoscopy of pigmented lesions. *BMC Dermatology*, *12*(14). Retrieved from http:// www.biomedcentral.com/1471-5945/12/14. doi: doi:10.1186/1471-5945-12-14 PMID:22916721.

Kennedy, J., Eberhart, R. C., & Shi, Y. (2001). *Swarm intelligence*. San Francisco: Morgan Kaufmann Publishers.

Khan, T. A., Talha, A. T., Talha, A., Asif, M. K., & Ijaz, I. (2012). Modeling of a standard particle swarm optimization algorithm in matlab by different benchmarks. In *Proceedings of Second International Conference on Innovative Computing Technology.* Casablanca, Morocco: InTech. doi: 10.1109/INTECH.2012.6457817.

Korotkov, K., & Garcia, R. (2012). Computerized analysis of pigmented skin lesions: A review. *Artificial Intelligence in Medicine*, *56*(2), 69–90. doi:10.1016/j.artmed.2012.08.002 PMID:23063256.

Lee, T. (2001). *Measuring Border Irregularity and Shape of Cutaneous Melanocytic Lesions.* (Doctoral dissertation). Burnaby, BC, Canada, Simon Fraser University. Retrieved from http://fas.sfu.ca/pub/cs/theses/2001/TimKamLeePhD.pdf.

Liu, Z., Smith, L., Sun, J., Smith, M., & Warr, R. (2011). Biological indexes based reflectional asymmetry for classifying cutaneous lesions. *Medical Image Compututing and Computer Assisted Intervention*, *14*(3), 124–132. PMID:22003692.

Madooei, A., Drew, M. S., Sadeghi, M., & Atkins, M. S. (2012). Intrinsic melanin and hemoglobin colour components for skin lesion malignancy detection. *Medical Image Computing and Computer Assisted Interventon*, *15*(1), 315–322. PMID:23285566.

Małaczewska, J., & Dabkowski, J. (2004). Melanocytic nevi as one of the risk factors of melanoma malignum. *Polski Merkuriusz Lekarski*, *16*(93), 298–302. PMID:15190614.

Mirjalili, S., & Mohd Hashim, S. Z. (2010). A new hybrid particle swarm optimization and gravitational search algorithm (PSOGSA) for function optimization. In *Proceedings of IEEE International Conference on Computer and Information Application.* China: IEEE Explore.

Moscarella, E., Catricala, C., Zalaudek, I., & Argenziano, G. (2010). The dermatoscope as the dermatologist's stethoscope. *Practical Dermathology*, 34-38. Retrieved from http://bmctoday.net/practicaldermatology/pdfs/PD0710_Dermoscopyfea.pdf.

Omran, M. G. H. (2004). *Particle Swarm Optimization Methods for Pattern Recognition and Image Processing.* (Doctoral dissertation). Pretoria, South Africa, University of Pretoria. Retrieved from http://upetd.up.ac.za/thesis/available/etd-02172005-110834/.

Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. An overview. *Swarm Intelligence*, *1*(1), 33–57. doi:10.1007/s11721-007-0002-0.

Popa, R., & Aiordachioaie, D. (2004). Genetic recognition of changes in melanocytic lesions. In *Proceedings of the 8th International Symposium on Automatic Control and Computer Science, SACCS 2004*. Iasi, Romania: IEEE Press. Retrieved from www.etc.ugal.ro/rpopa/papers/saccs2004.pdf.

Rallan, D., Dickson, M., Bush, N. L., Harland, C. C., Mortimer, P., & Bamber, J. C. (2006). High-resolution ultrasound reflex transmission imaging and digital photography: Potential tools for the quantitative assessment of pigmented lesions. *Skin Research and Technology*, *12*(1), 50–59. doi:10.1111/j.0909-725X.2006.00136.x PMID:16420539.

Romdhane, N. B., Taouil, K., Boudaya, S., Turki, H., & Bouhlel, M. S. (2007). Selection des variables et classification par reseaux de neurones des lesions dermatologiques. In *Proceedings of 4th International Conference on Sciences of Electronic, Technologies of Information and Telecommunications, SETIT 2007*. Tunisia: IEEE Explore. Retrieved from http://www.setit.rnu.tn/last_edition/setit2007/IV/18.pdf.

Schleicher, S. M., Schiffman, L. A., & Stairs, B. J. (2003). Malignant melanoma: Learn to recognize a killer. *Massage Today, 3*(9). Retrieved from http://www.massagetoday.com/.

Schmid-Saugeon, P. (2000). Symmetry axis computation for almost-symmetrical and asymmetrical objects: application to pigmented skin lesions. *Medical Image Analysis*, *4*(3), 269–282. doi:10.1016/S1361-8415(00)00019-0 PMID:11145313.

Slutsky, J. B., Barr, J. M., Femia, A. N., & Marghoob, A. A. (2010). Large congenital melanocytic nevi: Associated risks and management considerations. *Seminars in Cutaneous Medicine and Surgery*, *29*, 79–84. doi:10.1016/j.sder.2010.04.007 PMID:20579596.

Somasundaram, P., & Muthuselvan, N. B. (2010). A modified particle swarm optimization technique for solving transient stability constrained optimal flow. *Journal of Theoretical and Applied Information Technology*, 154-164. Retrieved from: http://www.jatit.org.

Spyridonos, P., Gaitanis, G., Bassukas, I. D., & Tzaphlidou, M. (2013). Gray Hausdorff distance measure for medical image comparison in dermatology: Evaluation of treatment effectiveness by image similarity. *Skin Research and Technology*, *19*(1), 498–506. doi:10.1111/srt.12001 PMID:23020792.

Strungs, I. (2004). Common and uncommon variants of melanocytic naevi. *Pathology*, *36*(5), 396–403. doi:10.1080/00313020412331283888 PMID:15370108.

Taouil, K., Romdhane, N. B., & Bouhlel, M. S. (2006). A new automatic approach for edge detection of skin lesion images. In *Proceedings of Second International Conference on Information and Communication Technologies from Theory to Applications.* Damascus, Syria: IEEE Explore. doi: 10.1109/ICTTA.2006.1684373.

Tosca, A., Kokolakis, A., Lasithiotakis, K., Zacharopoulos, A., Zabulis, X., & Marnelakis, I. et al. (2013). Development of a three-dimensional surface imaging system for melanocytic skin lesion evaluation. *Journal of Biomedical Optics*, *18*(1). doi:10.1117/1.JBO.18.1.016009 PMID:23296085.

Tzekis, P., Papastergiou, A., Hatzigaidas, A., Zaharis, Z., Kampitaki, D., Lazaridis, P., & Goula, M. (2009). A simple algorithm for automated skin lesion border detection. *WSEAS Transactions on Signal Processing, 5*(6), 229-240. Retrieved from http://www.wseas.us/e-library/transactions/signal/2009/32-369.pdf.

Ullman, S., & Basri, R. (1991). Recognition by linear combination of models. *IEEE Pattern Analysis and Machine Intelligence*, *13*(10), 992–1006. doi:10.1109/34.99234.

Vis, J. K. (2009). *Particle Swarm Optimizer for Finding Robust Optima*. (Unpublished dissertation). Leiden, Netherlands, Leiden University. Retrieved from http://ir.liacs.nl/assets/Bachelorscripties/2009-12JonathanVis.pdf.

Wallace, V. P., Bamber, J. C., Crawford, D. C., Ott, R. J., & Mortimer, P. S. (2000). Classification of reflectance spectra from pigmented skin lesions, a comparison of multivariate discriminant analysis and artificial neural networks. *Physics in Medicine and Biology*, *45*(10), 2859–2871. doi:10.1088/0031-9155/45/10/309 PMID:11049176.

Xu, D., & Kasparis, T. (2003). Detection and localization of edge contours. In Faust, N. L., & Roper, W. E. (Eds.), *Geo-Spatial and Temporal Image and Data Exploitation III*. SPIE Press. doi:10.1117/12.486351.

ADDITIONAL READING

Angeline, P. J. (1998). Using selection to improve particle swarm optimization. In *Proceedings of IEEE Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press. doi: 10.1109/ICEC.1998.699327.

Bachelder, I. A., & Ullman, S. (1992). Contour matching using local affine transformations. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition.* Champaign, IL: IEEE Press. doi: 10.1109/CVPR.1992.223169.

Banks, A., Vincent, J., & Anyakoha, C. (2008). A review of particle swarm optimization. Part II. *Hybridisation, Combinatorial, Multicriteria, and Constrained Optimization and Indicative Applications of Natural Computing*, *7*(1), 109–124. doi: doi:10.1007/s11047-007-9050-z.

Berenguer, V. J., Ruiz, D., & Soriano, A. (2009). Application of hidden markov models to melanoma diagnosis. In *Proceedings of International Symposium on Distributed Computing and Artificial Intelligence 2008.* Berlin: Springer-Verlag. doi: 10.1007/978-3-540-85863-8_42.

Birge, B. (2003). PSOT-A particle swarm optimization toolbox for use with matlab. In *Proceedings of the Swarm Intelligence Symposium*. Indianapolis, IN: IEEE Press. Retrieved from http://www4.ncsu.edu/~bkbirge/PSO/PSOt.exe.

Blackwell, T. M., & Bentley, P. J. (2002). Don't push me! Collision-avoiding swarms. In *Proceedings of IEEE Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press. doi: 10.1109/CEC.2002.1004497.

Blondin, J. (2009). Particle swarm optimization: A tutorial. Retrieved from http://cs.armstrong.edu/saad/csci8100/pso_tutorial.pdf.

Blum, A., Hofmann-Wellenhof, R., Luedtke, H., Ellwanger, U., Steins, A., & Roehm, S. et al. (2004). Value of the clinical history for different users of dermoscopy compared with results of digital image analysis. *Journal of the European Academy of Dermatology and Venereology*, *18*, 665–669. doi:10.1111/j.1468-3083.2004.01044.x PMID:15482291.

Blum, A., Luedtke, H., Ellwanger, U., Schwabe, R., Rassner, G., & Garbe, C. (2004). Digital image analysis for diagnosis of cutaneous melanoma. Development of a highly effective computer algorithm based on analysis of 837 melanocytic lesions. *The British Journal of Dermatology*, *151*, 1029–1038. doi:10.1111/j.1365-2133.2004.06210.x PMID:15541081.

Brenn, T. (2012). Pitfalls in the evaluation of melanocytic lesions. *Histopathology*, *60*(5), 690–705. doi:10.1111/j.1365-2559.2011.04042.x PMID:22176022.

Chakravarthy, C. K., & Reddy, P. (2010). Particle swarm optimization based approach for resource allocation and scheduling in OFDMA systems. *International Journal of Communications, Networks, and System Sciences*, *3*, 466–471. doi:10.4236/ijcns.2010.35062.

Clerc, M., & Kennedy, J. (2002). The particle swarm. Explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, *6*(1), 58–73. doi:10.1109/4235.985692.

Coelho, L. S., & Sierakowski, C. A. (2008). A software tool for teaching of particle swarm optimization fundamentals. *Advances in Engineering Software*, *39*, 877–887. Retrieved from http://www.elsevier.com/locate/advengsoft doi:10.1016/j.advengsoft.2008.01.005.

Culpepper, K. S., Granter, S. R., & McKee, P. H. (2004). My approach to atypical melanocytic lesions. *Journal of Clinical Pathology*, *57*(11), 1121–1131. doi:10.1136/jcp.2003.008516 PMID:15509670.

Donoser, M., Riemenschneider, H., & Bischof, H. (2010). Efficient partial shape matching of outer contours. Lecture Notes in Computer Science, Computer Vision–ACCV 2009, 5994, 281-292. Berlin: Springer-Verlag. doi: doi:10.1007/978-3-642-12307-8_26.

Dubuisson, M. P., & Jain, A. K. (1994). A modified Hausdorff distance for object matching. In *Proceedings of International Conference on Pattern Recognition.* Jerusalem, Israel: IEEE Press. doi: 10.1109/ICPR.1994.576361.

Fabbrocini, G., Betta, G., Di Leo, G., Liguori, C., Paolillo, A., & Pietrosanto, A. et al. (2010). epiluminescence image processing for melanocytic skin lesion diagnosis based on 7-point check-list: A preliminary discussion on three parameters. *The Open Dermatology Journal*, *4*, 110–115.

Gulia, A., Brunasso, A. M. G., & Massone, C. (2012). Dermoscopy: Distinguishing malignant tumors from benign. *Expert Review of Dermatology*, *7*(5), 439–458. doi:10.1586/edm.12.47.

Haupt, R. L., & Haupt, S. E. (2004). *Practical Genetic Algorithms* (2nd ed.). Hoboken, NJ: John Wiley & Sons.

Hoos, H. H., & Stützle, T. (2004). *Stochastic local search: Foundations and applications*. San Francisco: Morgan Kaufmann.

Iqbal, M., & Montes de Oca, M. A. (2006). An estimation of distribution particle swarm optimization algorithm. In M. Dorigo et al. (Eds.), *Ant Colony Optimization and Swarm Intelligence. 5th International Workshop, ANTS 2006, Springer LNCS 4150*. Berlin: Springer-Verlag. doi: 10.1007/11839088_7.

Iyatomi, H., Norton, K., Celebi, M. E., Schaefer, G., Tanaka, M., & Ogawa, K. (2010). Classification of Melanocytic Skin Lesions from Non-melanocytic Lesions. In *Proceedings of 2010 IEEE Annual International Conference on Engineering in Medicine and Biology Society.* Buenos Aires: IEEE Press. doi: 10.1109/IEMBS.2010.5626500.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks*. Piscataway, NJ: IEEE Press. doi: 10.1109/ICNN.1995.488968.

Kennedy, J., & Eberhart, R. (1997). A discrete binary version of the particle swarm algorithm. In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*. Piscataway, NJ: IEEE Press.

Kwasnicka, H., & Paradowski, M. (2005). Melanocytic lesion images segmentation enforcing by spatial relations based declarative knowledge. In *Proceedings of 5th International Conference on Intelligent Systems Design and Applications.* IEEE Press.

Larsson, F., Felsberg, M., & Forssen, P. E. (2009). Patch contour matching by correlating fourier descriptors. In *Proceedings of Digital Image Computing: Techniques and Applications*. Melbourne: IEEE Press. doi:10.1109/DICTA.2009.17.

Lee, T., McLean, D., Coldman, A., Gallagher, R., & Sale, J. (1995). A multi-stage segmentation method for images of skin lesions. *IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing*. Victoria, British Columbia: IEEE Press. doi: 10.1109/PACRIM.1995.520437.

Lee, Y. K., & Chen, L. H. (2002). Object-based image steganography using affine transformation. *International Journal of Pattern Recognition and Artificial Intelligence*, *16*(6), 681–696. doi:10.1142/S0218001402001952.

Li, X., Aldridge, B., Ballerini, L., Fisher, B., & Rees, J. (2009). Depth Data Improves Skin Lesion Segmentation. In Yang, G. Z. et al. (Eds.), *Medical Image Computing and Computer Assisted Intervention*. Berlin: Springer-Verlag.

Liu, J., Xu, W., & Sun, J. (2005). Quantum-behaved particle swarm optimization with mutation operator. In *Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence.* Piscataway, NJ: IEEE Press. doi: 10.1109/ICTAI.2005.104.

Lu, C., Latecki, L. J., Adluru, N., Yang, X., & Ling, H. (2009). Shape guided contour grouping with particle filters. In *Proceedings of IEEE 12th International Conference on Computer Vision.* Kyoto: IEEE Press. doi: 10.1109/ICCV.2009.5459446.

McKee, P. H. (2010). Clues to the diagnosis of atypical melanocytic lesions. [Bethesda, MD: PubMed.]. *Histopathology*, *56*(1), 100–111. doi:10.1111/j.1365-2559.2009.03451.x PMID:20055908.

Mendes, R., Kennedy, J., & Neves, J. (2004). The fully informed particle swarm: Simpler, maybe better. *IEEE Transactions on Evolutionary Computation*, *8*(3), 204–210. doi:10.1109/TEVC.2004.826074.

Mohais, A., Mendes, R., Ward, C., & Postoff, C. (2005). Neighborhood re-structuring in particle swarm optimization. In S. Zhang and R. Jarvis (Eds.), *Proceedings of the 18th Australian Joint Conference on Artificial Intelligence.* Berlin: Springer-Verlag.

Norton, K. A., Iyatomi, H., Celebi, M., Schaefer, G., Tanaka, M., & Ogawa, K. (2010). Development of a novel border detection method for melanocytic and non-melanocytic dermoscopy images. In *Proceedings of IEEE Conference on Engineering in Medicine and Biology Society.* Buenos Aires: IEEE Press. doi: 10.1109/IEMBS.2010.5626499.

O'Connor, K. M. & Chien, A. J. (2008). Management of Melanocytic Lesions in the Primary Care Setting. *Mayo Clinical Proceedings, 83*(2), 208-214. doi: http://dx.doi.org/10.4065/83.2.208.

Oliveira, F. P. M., & Tavares, J. M. R. S. (2009). Contours Matching using Curvature Information and Optimization Based on Dynamic Programming. *IEEE Latin America Transactions*, *7*(6), 703–712. doi:10.1109/TLA.2009.5419369.

Oliveira, F. P. M., & Tavares, J. M. R. S. (2008). *Optimization of the global matching between two contours defined by ordered points using an algorithm based on dynamic programming*. Retrieved from http://repositorio-aberto.up.pt/bitstream/10216/6643/2/14995.pdf.

Parsopoulos, K. E., & Vrahatis, M. N. (2002). Recent approaches to global optimization problems through particle swarm optimization. *Natural Computing*, *1*(2-3), 235–306. doi:10.1023/A:1016568309421.

Pla, F., & Marchant, J. A. (1997). Matching feature points in image sequences through a region-based method. *Computer Vision and Image Understanding*, *66*(3), 271–285. doi:10.1006/cviu.1996.0512.

Poli, R., Di Chio, C., & Langdon, W. B. (2005). Exploring extended particle swarms: A genetic programming approach. In *Proceedings of the 2005 Conference on Genetic and Evolutionary Computation*. New York: ACM Press. doi: 10.1145/1068009.1068036.

Popa, R. (2012). Genetic algorithms: An overview with applications in evolvable hardware. In Gao, S. (Ed.), *Bio-inspired Computational Algorithms and their Applications*. Rijeka, Croatia: InTech.

Ravariu, C. (2011). Contributions to novel methods in electrophysiology aided by electronic devices and circuits. In Gargiulo, G. (Ed.), *Applied Biomedical Engineering*. Rijeka, Croatia: InTech. doi:10.5772/21443.

Reynolds, C. W. (1987). Flocks, herds, and schools: A distributed behavioral model. *ACM Computer Graphics, 21*(4), 25-34. Retrieved from: http://www.cs.toronto.edu/~dt/siggraph97-course/cwr87/.

Riget, J., & Vesterstroem, J. (2002). A diversity-guided particle swarm optimizer-the ARPSO. *Technical Report 2002-02*. Aarhus, Denmark: University of Aarhus Press. Retrieved from http://www.evalife.dk.

Rosenhahn, B., Brox, T., Cremers, D., & Seidel, H. P. (2006). A comparison of shape matching methods for contour based pose estimation. In Reulke, R. et al. (Eds.), *Combinatorial Image Analysis*. Berlin: Springer-Verlag. doi:10.1007/11774938_21.

Setayesh, M., Zhang, M., & Johnston, M. (2011). A novel local thresholding technique in PSO for detecting continuous edges in noisy images. In *Proceedings of 26th International Conference on Image and Vision Computing*. Auckland, New Zealand: ACM Press. Retrieved from www.ivs.auckland.ac.nz/ivcnz2011_temp/uploads/print_papers/1058.pdf

Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer. *IEEE International Conference on Evolutionary Computation*. Piscataway, NJ: IEEE Press. doi: 10.1109/ICEC.1998.699146.

Shi, Y., & Eberhart, R. (1999). Empirical study of particle swarm optimization. *IEEE Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press. doi: 10.1109/CEC.1999.785511.

Suganthan, P. N. (1999). Particle swarm optimiser with neighbourhood operator. *IEEE Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press. doi: 10.1109/CEC.1999.785514.

Trelea, I. C. (2003). The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, *85*(6), 317–325. doi:10.1016/S0020-0190(02)00447-7.

Ushie, J. O., Obu, J. A. E., & Iniobong, P. (2012). Optimising digital combinational circuit using particle swarm optimisation technique. *Latin American Journal of Physics Education, 6*(1), 72-77. Retrieved from from http://www.lajpe.org

Wang, F., Xing-shi, S., Luo, L., & Wang, Y. (2011). Hybrid optimization algorithm of PSO and Cuckoo Search. In *Proceedings of Second International Conference on Artificial Intelligence, Management Science, and Electronic Commerce*. Deng Leng, China: IEEE Press. doi: 10.1109/AIMSEC.2011.6010750.

Wang, K. P., Huang, L., Zhou, C. G., & Pang, W. (2003). Particle swarm optimization for traveling salesman problem. *IEEE International Conference on Machine Learning and Cybernetics*. Piscataway, NJ: IEEE Press. doi: 10.1109/ICMLC.2003.1259748.

Wang, L., Kang, Q., Xiao, H., & Wu, O. (2005). A modified adaptive particle swarm optimization algorithm. In *Proceedings of IEEE International Conference on Industrial Technology*. China: IEEE Press. doi: 10.1109/ICIT.2005.1600637.

Wurm, E., Curchin, C., & Soyer, H. P. (2010). Recent advances in diagnosing cutaneous melanomas. *F1000. Medica-Report*, *2*(46). doi: doi:10.3410/M2-46.

Xiaohui, H., & Russell, E. (2002). Multiobjective optimization using dynamic neighborhood particle swarm optimization. *IEEE Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press. doi: 10.1109/CEC.2002.1004494.

Yadav, R., & Mandal, D. (2011). Optimization of artificial neural network for speaker recognition using particle swarm optimization. *International Journal of Soft Computing and Engineering*, *1*(3), 80–84.

Zavala, A. E. M., Aguirre, A. H., & Diharce, E. R. V. (2005). Particle evolutionary swarm optimization algorithm (PESO). In *Proceedings of Sixth Mexican International Conference on Computer Science.* Puebla, Mexico: IEEE Press. doi: 10.1109/ENC.2005.32.

KEY TERMS AND DEFINITIONS

Affine Transformation: A transformation which preserves straight lines and ratios of distances between points lying on a straight line. It is equivalent to a linear transformation followed by a translation.

Combination of Views: In the theory of algebraic functions of view, the variety of 2-D views depicting an object can be expressed as a combination of a small number of 2-D views of the object. Results are also available in 3-D.

Dermatoscope: An instrument used in dermatology, which lets you look at the upper 2 mm of the skin by the use of polarized light.

Dermatoscopy: (Also known as dermoscopy) Is the examination of skin lesions with a dermatoscope.

Edge Detection: A set of mathematical methods which aim at identifying points in a digital image at which the image brightness changes sharply. These points generates an edge.

Genetic Algorithm: A search heuristic that generate solutions to optimization problems using techniques inspired by natural evolution, such as selection, crossover and mutation.

Image Matching: The act of checking a distance function between two sets of points which belong to some images taken from the same content.

Image Segmentation: The process of partitioning a digital image into multiple segments (sets of pixels) for the purpose to simplify or change the representation of an image into something that is more meaningful and easier to analyze.

Melanocytes: Cells that produce the dark pigment, melanin, which is responsible for the color of skin. These cells are located in the bottom layer of the skin's epidermis and in other parts of the body.

Melanocytic Nevus: A type of lesion that contains nevus cells (a type of melanocyte), called by some sources with the term "mole".

Melanoma: A malignant tumor of melanocytes and the most dangerous skin cancer. Early detection of melanoma, while it is still small and thin, and completely removal of the tumor, significantly improves the chances of cure.

Particle Swarm Optimization: A computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. The population of particles are moving in the search space, according with an heuristic based on each particle's position and velocity. Each particle's movement is influenced by its local best known position and by the best position of the swarm, which are updated as better positions are found by other particles.

APPENDIX

AUTOMATIC DETECTION OF SKIN LESION BORDER PSEUDOCODE USING NOTATION FROM BROWNLEE, (2011)

Algorithm 1. Pseudocode for Border Detection

```
Input: RGB color image in matrix M, k
Output: Gray level image in matrix new_I
1  convert  I ← M;
2  foreach  i ∈ k  do
          vector (i) ← I(random_1(i), random_2(i));
   end
3  threshold ← average (vector);
4  foreach  pixel n ∈ I  do
            if  I(n) ≤ threshold
                      B(n) ← 0;
            else
                      B(n) ← 1;
            end
   end
5  D ← dilate (B);
6  boundary  ← NOT{XOR[B,D]};
7  foreach  pixel n ∈ I  do
          if  boundary(n) = 1
                   new_I(n) ← I(n);
          else
                   new_I(n) ← 255;
          end
   end
```

Automatic Detection of Skin Lesion Border

Matlab Code

```
clear all;

M=imread('image_1.jpg');              % original RGB color image;
Figure(1)
imshow(M);

I=rgb2gray(M);                        % conversion in gray scale;
Figure(2)
```

```
imshow(I);

[m,n] = size(I);
nr_pixeli = 100;                    % choose k, a random number of pixels;
A(nr_pixeli) = 0;                   % vector which contains selected pixels;
B(m,n) = 0;                         % new image matrix;

for k = 1:nr_pixeli
    c1 = fix(m * rand) + 1; c2 = fix(n * rand) + 1;
    A(k) = I(c1, c2);
end

threshold = 0.8*sum(A)/nr_pixeli;

for i = 1:m
    for j = 1:n
        if I(i,j)<threshold;
            B(i,j) = 0;
        else
            B(i,j) = 255;
        end
    end
end

Figure(3)
temp = B;
imshow(B);                          % a binary image of the lesion;

filt = medfilt2(temp,[5 5]);        % optional filtering;

se = strel('square',3);
interm = 255 - filt;
D = imdilate(interm,se);            % dilated version of binary image;

boundary = filt - D;
for i = 1:m
    for j = 1:n
        if boundary(i,j)<0;
            boundary(i,j) = 255;
        end
    end
end
```

```
Figure(4)
temp = boundary;
imshow(boundary);                    % the border of the lesion;

new_I = I;

for i = 1:m
    for j = 1:n
        if temp(i,j)==0;
           new_I(i,j,:) = 255;
        end
    end
end

Figure(5)
imshow(new_I);                       % original image and lesion's border;
```

PSO SEARCH OF THE TRANSFORMATION PARAMETERS PSEUDOCODE USING NOTATION FROM BROWNLEE, (2011)

Algorithm 2. Pseudocode for PSO Search.

Input: *two binary images of contours in matrices* I *and* M, population_size, nr_iterations
Output: P_{g_best}

establish the 2 sets of points from the 2 contours I *and* M;
$P_{g\_best} \leftarrow 0$;
for $i = 1$ **to** *population_size* **do**
  $P_{position} \leftarrow$ RandomPosition(*population_size*);
  $P_{velocity} \leftarrow 0$;
  $P_{p\_best} \leftarrow P_{position}$;
while StopCondition **do**
  foreach $P \in$ *Population*
    $P_{cost} \leftarrow$ Cost($P_{position}$);
    if $P_{cost} \leq P_{p\_best}$ **then**
      $P_{p\_best} \leftarrow P_{position}$;
      if $P_{cost} \leq P_{g\_best}$ **then**
        $P_{g\_best} \leftarrow P_{p\_best}$;
      end
    end
    $P_{velocity} \leftarrow$ UpdateVelocity($P_{velocity}$, P_{p_best}, P_{g_best});
    $P_{position} \leftarrow$ UpdatePosition($P_{position}$, $P_{velocity}$);
  end
end

PSO Search of the Transformation Parameters

Matlab Code

```
clear all;

nr_agenti = 20;                     % the number of particles;
nr_iteratii = 100;                  % the number of iterations;
factor_scala = 1;                   % a scale factor;

I=imread('image_1.bmp');            % images have 150 x 150 pixels;
Figure(1)
imshow(I);

M=imread('image_2.bmp');
Figure(2)
imshow(M);

k = 1;

for i = 1:10:150                    % generate the first set of points;
    for j = 1:150
        if I(i,j) == 0
            xc(k) = i;
            yc(k) = j;
            k = k + 1;
        end
    end
end

I_puncte = ones(150,150);

for p = 1:k-1,
    I_puncte(xc(p),yc(p)) = 0;
end

I_puncte = lupa(I_puncte);          % magnify the points 4 times;

Figure(3)
imshow(I_puncte);

k = 1;
```

```
for i = 1:10:150                    % generate the second set of points;
    for j = 1:150
        if M(i,j) == 0
            xf(k) = i;
            yf(k) = j;
            k = k + 1;
        end
    end
end

M_puncte = ones(150,150);

for p = 1:k-1,
    M_puncte(xf(p),yf(p)) = 0;
end

M_puncte = lupa(M_puncte);          % magnify the points 4 times;

Figure(4)
imshow(M_puncte);

lungime_tinta = length(xf);         % calculate the same number of points;
lungime = length(xc);
if lungime_tinta>=lungime
    xf = xf(1:lungime);
    yf = yf(1:lungime);
else
    xc = xc(1:lungime_tinta);
    yc = yc(1:lungime_tinta);
end

                                    % PSO algorithm

populatie = rand(nr_agenti,6)*factor_scala;
viteza (nr_agenti,6) = 0;

for q = 1:nr_iteratii
    eval = evaluez(populatie,xc,yc,xf,yf);
    [minima indice] = min(eval);
    best = populatie(indice,:);

    a1 = best(1); a2 = best(2); a3 = best(3);
    b1 = best(4); b2 = best(5); b3 = best(6);
```

```
    lungime = length(xc);

    V=[xc' yc' ones(lungime,1)];     %affine transformation
    S = [a1 b1; a2 b2; a3 b3];
    R = V * S;
    x_nou = R(:,1);
    y_nou = R(:,2);

    x_nou = fix(x_nou);
    y_nou = fix(y_nou);

    for k = 1:4,
            pop(1:5,:) = populatie(5*k-4:5*k,:);
            eval_local = evaluez(pop,xc,yc,xf,yf);
            [minime(k) pozitii(k)] = min(eval_local);
    end
    pozitii(5) = indice;

    [new_populatie new_viteza] = actualizez(populatie, pozitii, viteza);

    solutie(q) = minima;
    populatie = new_populatie;
    viteza = new_viteza;
end

M_puncte(1:150,1:150)=255; M_modif(1:150,1:150)=255;
for k=1:lungime,
       M_puncte(x_nou(k),y_nou(k))= 0;
end
M_puncte = lupa(M_puncte);
    for i=1:150,
        for j=1:150,
            if (M_puncte(i,j) == 0)||(M(i,j) == 0)
                M_modif(i,j) = 0;
            end
        end
    end

Figure(5)
imshow(M_modif);

Figure(6)
plot(solutie);
```

```
title('The evolution of the distance between the two images');
xlabel('Number of iterations');
ylabel('Distance in pixels');

%   UPDATE  FUNCTION
function [new_pop new_speed] = actualizez(pop,pozitii,speed)

coeficient_1 = 2;
coeficient_2 = 2;
v_max = 0.5;

[linii coloane] = size(pop);

best = pop(pozitii(5),:);

for v=1:5,
    for s = 1:4,
        best_local(v+5*(s-1),:) = pop(pozitii(s),:);
    end
end

for c=1:linii,
    for r=1:coloane,
        new_speed(c,r) = speed(c,r) + coeficient_1 * rand * … (best_local(c,r)
- pop(c,r)) + coeficient_2 * rand * (best(r) - pop(c,r));
        if new_speed(c,r) > v_max
            new_speed(c,r) = v_max;
        elseif new_speed(c,r) < -v_max
            new_speed(c,r) = -v_max;
        end
        new_pop(c,r) = pop(c,r) + new_speed(c,r);
    end
end

new_pop(pozitii(5),:) = best;


%   EVALUATION FUNCTION
function eval  = evaluez(pop,x_vector,y_vector,x_tinta,y_tinta)

[linii coloane] = size(pop);

for c=1:linii,
```

```
    a1 = pop(c,1);a2 = pop(c,2);a3 = pop(c,3);
    b1 = pop(c,4);b2 = pop(c,5);b3 = pop(c,6);

    lungime_tinta = length(x_tinta);
    lungime = length(x_vector);
    if lungime_tinta>=lungime
        x_tinta = x_tinta(1:lungime);
        y_tinta = y_tinta(1:lungime);
    else
        x_vector = x_vector(1:lungime_tinta);
        y_vector = y_vector(1:lungime_tinta);
    end

    new_lungime = length(x_vector);
    V=[x_vector' y_vector' ones(new_lungime,1)];
    S = [a1 b1; a2 b2; a3 b3];
    R = V * S;
    x_new = R(:,1);
    y_new = R(:,2);

    % Hausdorff distance

    eval(c) = 0;
    for r = 1:new_lungime,
        shortest = 1000000;
        for s = 1:new_lungime,
            d(r,s) = sqrt((x_new(r)-x_tinta(s))^2 + (y_new(r)- … y_tinta(s))^2);
            if d(r,s) < shortest
                shortest = d(r,s);
            end
        end
        if shortest > eval(c)
            eval(c) = shortest;
        end
    end
end
```