

Computational Intelligence CSA-Based Approach for Machine-Driven Calculation of Outline Curves of Cutaneous Melanoma

Akemi Gálvez^{1,2}, Andrés Iglesias^{1,2,†}

¹Department of Information Science, Faculty of Sciences, Toho University, Funabashi, JAPAN

²Dpt. of Applied Mathematics & Computational Sciences, Universidad de Cantabria, Santander, SPAIN

[†]Corresponding author. Email: iglesias@unican.es

Abstract—This paper addresses the problem of obtaining automatically a good approximation of the outline curve of skin lesions from dermoscopy images. This problem appears as a critical step in machine-driven segmentation of dermoscopy images for semi-automatic early diagnosis of cutaneous melanoma. Given a set of feature points selected by a specialist, the method applies a powerful nature-inspired metaheuristic optimization method called cuckoo search algorithm (CSA) to obtain the free-form parametric Bézier curve that fits the points better in the least-squares sense. Two illustrative examples of a benign and a malignant skin lesions (a naevus and a melanoma, respectively) are used to evaluate the performance of the method. Our experimental results show that the method performs very well and can be applied as an intermediate step of semi-automatic image segmentation for early diagnosis of cutaneous melanoma.

Index Terms—computational intelligence, artificial intelligence for cognitive health, cuckoo search algorithm, machine-driven segmentation, outline curves, cutaneous melanoma

I. INTRODUCTION

Malignant melanoma is the most frequent and most dangerous type of skin cancer. It is also one of the most rapidly increasing types of cancer in the world. As a result, melanoma has become one of the biggest health concerns all over the world. An important feature of melanoma is that early detection is critical for survival. Early diagnosis is also important because early-stage melanoma can be cured by a simple clinical procedure (e.g., a simple excision).

Visual inspection by a specialist is the most common diagnostic procedure. However, it is difficult to distinguish visually the melanoma from other skin features. Other diagnosis procedures include the ABCDE method, the Menzies scale, the 7-point checklist, and different types of biopsy. These procedures rely heavily on human intervention, leading to varying diagnostic results. Other procedures involve imaging tests. Among them, dermoscopy is the most precise and reduces screening errors as it enhances discrimination between real melanoma and other skin lesions [3]. However, dermoscopy is also prone to errors due to the subjectivity of visual interpretation of images. Nowadays, there is a strong demand for automated procedures to reduce the human part in some critical steps. This is the main motivation of this paper.

First step in melanoma diagnosis is to classify the skin lesion as benign or malignant. To this purpose, analysis of dermoscopy images is carried out involving image segmentation,

to distinguish the foreground (the lesion) from the background (the skin). Several methods have been applied to this problem, including: thresholding methods [6], edge-based methods [2], clustering methods [37], gradient vector flow snakes [9], level set methods [24] and active contours [23].

A key task in this segmentation step is the border detection, i.e., the determination of the outline curve of the skin lesion from the image. Until recently, the border detection was handled manually by dermatologists, by clicking with the mouse on different points of the picture to get an initial collection of data points joined with segments to cover the region of the lesion. This manual procedure is time consuming and prone to errors. Also, the polyline is not the best graphical primitive for this process, as the border of skin lesions rarely happens to be piecewise linear, but smooth. Since the input data is a collection of 2D points, automated parametric approximation schemes are better suited for this task.

This paper addresses the problem of obtaining automatically a good approximation of the outline curve of skin lesions from dermoscopy images. Given a set of feature points selected by a specialist, the method applies a powerful nature-inspired metaheuristic optimization method called cuckoo search algorithm to obtain an accurate approximation by using free-form parametric curves (in particular, polynomial Bézier curves).

The structure of this paper is as follows: Section II describes the previous work on free-form parametric curve approximation. Section III describes the cuckoo search algorithm. Our proposed method is described in detail in Section IV. Our results and conclusions are discussed in Sections V and VI.

II. PREVIOUS WORK

Data approximation with free-form parametric curves has been studied for decades. First approaches relied on numerical procedures [7], [10]. Subsequent attempts provided only partial solutions [5]. Mathematically, this issue can be formulated as an optimization problem [21]. Some approaches in this line use error bounds [26], curvature-based squared distance minimization [30], or dominant points [27]. In general, they perform well but require some particular constraints (high differentiability, noiseless data) which are not common in real-world applications. Recently, artificial intelligence has been applied to this problem, such as standard neural networks [17]

or self-organizing maps nets [18]. This neural approach is combined with partial differential equations in [4]. Generalization of these methods to functional networks is analyzed in [19], and in combination with genetic algorithms in [16]. Application of support vector machines is reported in [20]. Other approaches are based on the application of nature-inspired metaheuristic techniques [8], [22], [31], [32]. Genetic algorithms have been applied to this problem in [35], [36]. Other approaches include the use of the popular particle swarm optimization [11], artificial immune systems [15], [29], firefly algorithm [12], [13] and memetic approaches [14]. However, these methods are designed for explicit curves and are not applicable to the parametric case.

III. THE CUCKOO SEARCH ALGORITHM

The *Cuckoo search algorithm* (CSA) is a nature-inspired population-based metaheuristic algorithm for optimization proposed in [33]. It is inspired by the obligate interspecific brood-parasitism of some cuckoo species that lay their eggs in the nests of host birds of other species for raising their offspring. This interesting and surprising breeding behavioral pattern is the metaphor of the CSA, where the eggs in the nest are interpreted as a pool of candidate solutions of an optimization problem while the cuckoo egg represents a new coming solution. The cuckoo search algorithm is based on three idealized rules [33], [34]:

- 1) Each cuckoo lays one egg at a time, and dumps it in a randomly chosen nest;
- 2) The best nests with high quality of eggs (solutions) will be carried over to the next generations;
- 3) The number of available host nests is fixed, and a host can discover an alien egg with a probability $p_a \in [0, 1]$.

For simplicity, the third assumption is approximated by a fraction p_a of the \mathcal{P} nests being replaced by new nests (with new random solutions at new locations). For a minimization problem, the quality or fitness of a solution can simply be proportional to the objective function. Based on these rules, the CSA can be summarized in the pseudo-code in Table I. Basically, it starts with an initial population of \mathcal{P} host nests and it is performed iteratively. The initial values of the j th component of the i th nest are determined by the expression $x_i^j(0) = rand.(up_i^j - low_i^j) + low_i^j$, where up_i^j and low_i^j represent the upper and lower bounds of that j th component, respectively, and $rand$ represents a standard uniform random number on the interval $(0, 1)$. This choice ensures that the variables are within the search space domain. These boundary conditions are also controlled in each iteration step.

For each iteration g , a cuckoo egg i is selected randomly and new solutions $\mathbf{x}_i(g+1)$ are generated by using the Lévy flight, a random walk in which the steps are defined in terms of the step-lengths, with the directions of the steps being isotropic and random. The use of Lévy flights is preferred over other random walks because it leads to better overall performance of the CSA. The general equation for the Lévy flight is:

$$\mathbf{x}_i(g+1) = \mathbf{x}_i(g) + \alpha \oplus levy(\lambda) \quad (1)$$

TABLE I: Cuckoo Search Algorithm (CSA) via Lévy flights.

Algorithm: Cuckoo Search Algorithm (CSA) via Lévy Flights	
begin	Objective function $f(\mathbf{x})$, $\mathbf{x} = (x_1, \dots, x_D)^T$
	Generate initial population of p host nests \mathbf{x}_i ($i = 1, 2, \dots, \mathcal{P}$)
	while ($t < MaxGeneration$) or (stop criterion)
	Get a cuckoo (say, i) randomly by Lévy flights
	Evaluate its fitness F_i
	Choose a nest among \mathcal{P} (say, j) randomly
	if ($F_i > F_j$)
	Replace j by the new solution
	end
	A fraction (p_a) of worse nests are abandoned and new ones are built via Lévy flights
	Keep the best solutions (or nests with quality solutions)
	Rank the solutions and find the current best
	end while
	Postprocess results and visualization
end	

where g indicates the number of the current generation, and $\alpha > 0$ indicates the step size, related to the scale of the problem under study. The symbol \oplus indicates the entry-wise multiplication. Eq. (1) is essentially a Markov chain, since next location at generation $g+1$ only depends on the current location at generation g and a transition probability, given by the first and second terms of Eq. (1), respectively. This transition probability is modulated by the Lévy distribution:

$$levy(\lambda) \sim g^{-\lambda}, \quad (1 < \lambda \leq 3) \quad (2)$$

which has an infinite variance and infinite mean. Computationally, the generation of random numbers with Lévy flights is comprised of two steps: firstly, a random direction according to a uniform distribution is chosen; then, the generation of steps following the chosen Lévy distribution is carried out. We use the Mantegna's algorithm for symmetric distributions:

$$\hat{\phi} = \left(\frac{\Gamma(1 + \hat{\beta}) \cdot \sin\left(\frac{\pi \cdot \hat{\beta}}{2}\right)}{\Gamma\left(\frac{1 + \hat{\beta}}{2}\right) \cdot \hat{\beta} \cdot 2^{\frac{\hat{\beta}-1}{2}}}\right)^{\frac{1}{\hat{\beta}}} \quad (3)$$

where Γ denotes the Gamma function and $\hat{\beta} = 3/2$ as in [34]. This factor is used to compute the step length $\zeta = \frac{u}{|v|^{\frac{1}{\hat{\beta}}}}$, where u and v follow the normal distribution of zero mean and deviation σ_u^2 and σ_v^2 , respectively, where σ_u obeys the Lévy distribution given by Eq. (3) and $\sigma_v = 1$. Then, the stepsize ζ is computed as $\zeta = 0.01 \zeta(\mathbf{x} - \mathbf{x}_{best})$. Finally, \mathbf{x} is modified as: $\mathbf{x} \leftarrow \mathbf{x} + \zeta \cdot \Psi$ where Ψ is a random vector following the normal distribution $N(0, 1)$. The CSA then evaluates the fitness of the new solution. Also, a fraction of the worse nests (according to the fitness) are abandoned and replaced by new solutions to increase exploration of the search space looking for more promising solutions. The rate of replacement is given by p_a , a parameter of the model that has to be tuned for better performance. For each iteration, all current solutions are ranked according to their fitness and best solution so far is

stored as the vector \mathbf{x}_{best} . This algorithm is applied iteratively until a fixed number of generations has been reached.

IV. THE PROPOSED METHOD

A. Problem to be solved

A free-form parametric Bézier curve $\mathbf{C}(t)$ of degree n is defined as:

$$\mathbf{C}(t) = \sum_{j=0}^n \mathbf{P}_j B_j^n(t) \quad (4)$$

where \mathbf{P}_j are vector coefficients called *control points*, $B_j^n(t)$ are the *Bernstein polynomials of index j and degree n* , given by:

$$B_j^n(t) = \binom{n}{j} t^j (1-t)^{n-j}$$

and t is the *curve parameter*, defined on a finite interval $[0, 1]$. In this paper, $0! = 1$ and vectors are denoted in bold.

Suppose now a given set of data points $\{\mathbf{Q}_i\}_{i=1,\dots,m}$ in \mathbb{R}^2 . Our goal is to obtain the free-form parametric Bézier curve $\mathbf{C}(t)$ that fits these data points better in the discrete least-squares sense. To do so, we have to compute the control points \mathbf{P}_j ($j = 0, \dots, n$) of $\mathbf{C}(t)$ by minimizing the least-squares error, E , defined as the sum of squares of the residuals:

$$E = \sum_{i=1}^m \left(\mathbf{Q}_i - \sum_{j=0}^n \mathbf{P}_j B_j^n(t_i) \right)^2 \quad (5)$$

where we need the parameter value t_i associated with each \mathbf{Q}_i , $i = 1, \dots, m$. Considering the column vectors $\mathbf{B}_j = (B_j^n(t_1), \dots, B_j^n(t_m))^T$, $j = 0, \dots, n$, where $(\cdot)^T$ means transposition, and $\mathbf{Q} = (\mathbf{Q}_1, \dots, \mathbf{Q}_m)$, Eq. (5) becomes the following system of equations (called the *normal equation*):

$$\begin{pmatrix} \mathbf{B}_0^T \cdot \mathbf{B}_0 & \dots & \mathbf{B}_n^T \cdot \mathbf{B}_0 \\ \vdots & \ddots & \vdots \\ \mathbf{B}_0^T \cdot \mathbf{B}_n & \dots & \mathbf{B}_n^T \cdot \mathbf{B}_n \end{pmatrix} \begin{pmatrix} \mathbf{P}_0 \\ \vdots \\ \mathbf{P}_n \end{pmatrix} = \begin{pmatrix} \bar{\mathbf{Q}} \cdot \mathbf{B}_0 \\ \vdots \\ \bar{\mathbf{Q}} \cdot \mathbf{B}_n \end{pmatrix} \quad (6)$$

which can be compacted as:

$$\mathbf{M} \mathbf{P} = \mathbf{R} \quad (7)$$

with $\mathbf{M} = \left[\sum_{j=1}^m B_l^n(t_j) B_i^n(t_j) \right]$, $\mathbf{R} = \left[\sum_{j=1}^m \mathbf{Q}_j B_l^n(t_j) \right]$ for $i, l = 0, \dots, n$.

If values are assigned to the t_i , our problem is a classical linear least-squares minimization, with the $\{\mathbf{P}_i\}$ as unknowns. It can readily be solved by standard numerical techniques. On the contrary, if the t_i are unknowns, the problem becomes much more difficult. Since the blending functions $B_j^n(t)$ are nonlinear in t , it is a nonlinear continuous optimization problem. In many cases the number of data points can be extremely large, so we have to deal with a large number of unknowns. In other words, we are also confronted with a high-dimensional problem. It is also a multimodal problem, since there might be several data parameterization vectors leading to the optimal solution. As a conclusion, solving the

parameterization problem is the key to obtain an optimal Bézier fitting curve of data, but it also leads to a very difficult multimodal, multivariate, continuous, nonlinear optimization problem. In this work, we seek to solve this general problem. In our approach we make no assumption about the values of data parameters; instead, we compute them at full extent.

B. Our Approach

In this paper we are provided with a set of data points $\{\Theta_\mu\}_{\mu=1,\dots,\eta}$ selected by a trained dermatologist from dermoscopy images as belonging to the border between a skin lesion and the background. The list $\{\Theta_\mu\}$ is always sorted and defines an outline curve enclosing a region containing the skin lesion under analysis. Our goal is to compute a Bézier curve approximating the data points accurately. The approach is particularly adequate for this problem because it describes the border of the lesion through a smooth mathematical equation instead of a list of data points connected by straight lines.

Our method consists of applying the CSA described above to determine suitable parameter values for the least-squares minimization in (5). To this aim, each host nest, representing a potential solution, corresponds to a parametric vector $\mathcal{T}^k = (t_1^k, t_2^k, \dots, t_\eta^k) \in [0, 1]^\eta$, ($k = 1, \dots, \mathcal{P}$) where the $\{t_\mu^k\}_{\mu=1,\dots,\eta}$ are strictly increasing parameters. These parametric vectors are initialized with random values and then sorted. Application of our method yields new nests representing the potential solutions of this problem. The process is performed iteratively for a given number of generations \mathcal{N}_{iter} . The nest with the best global value for our fitness function is taken as the final solution of our minimization problem.

C. Parameter tuning

A major limitation of all nature-inspired metaheuristic optimization techniques is that they depend on a number of parameters that have to be properly tuned. This is a very difficult task for several reasons: on one hand, the field lacks sufficient theoretical studies about this problem; on the other hand, the optimal choice of parameter values is problem-dependent, so good values for a particular problem might be no longer adequate for other problems. In this sense, the CSA is particularly favorable because of its simplicity. In contrast to other metaheuristic methods that need a large number of parameters, the CSA only requires three parameters:

- the number of iterations, \mathcal{N}_{iter} ,
- the population size \mathcal{P} , and
- the probability p_a .

We carried out several hundreds of executions to determine good values for these parameters empirically. From them, we concluded that $\mathcal{N}_{iter} = 10,000$ are more than enough to reach convergence for our final choice of the other parameters. We also observed that a population of $\mathcal{P} = 100$ host nests, representing the number of candidate solutions for the method, is adequate for this problem in terms of the quality of the solutions and the CPU time. Regarding the parameter p_a , our choice is completely empirical: we carried out some simulations for different values of this parameter, and found that the

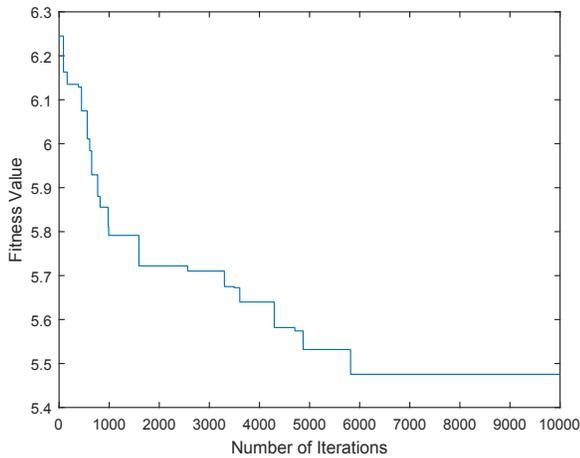
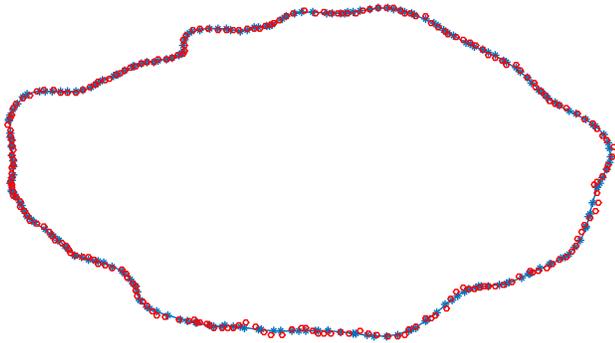
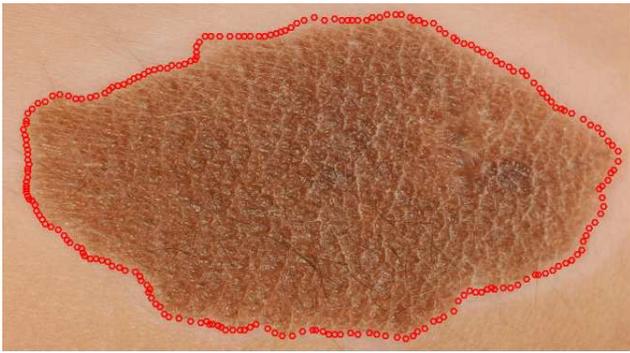


Fig. 1: Example of a benign skin lesion (a naevus): (top) dermoscopy image with feature points (red empty circles); (middle) feature points (in red), best fitting Bézier curve (solid blue line) and reconstructed feature points (blue stars); (bottom) convergence diagram.

results do not change significantly in any case. However, we observed that values around $p_a = 0.25$ reduce the number of iterations required for convergence, so this is the value taken in this paper.

D. Implementation issues

All the computational work in this paper has been performed on a personal PC with a 2.6 GHz. Intel Core i7 processor and 8 GB. of RAM. The source code has been implemented by the authors in the programming language of the popular

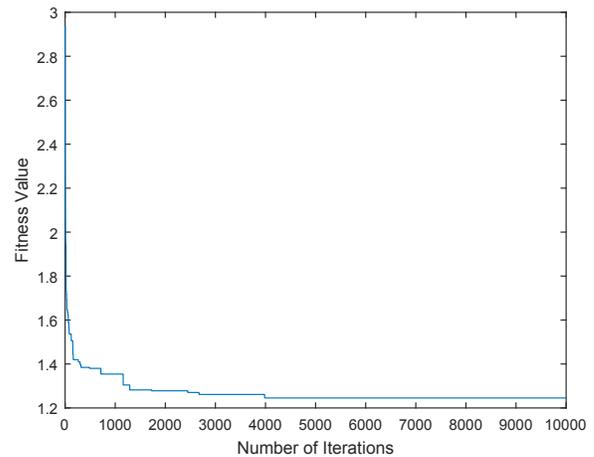
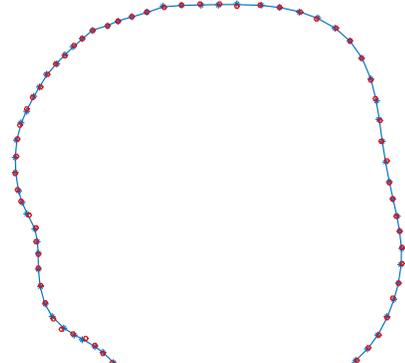


Fig. 2: Example of a malignant skin lesion (a melanoma): (top) dermoscopy image along with feature points (red empty circles); (middle) feature points (in red), best fitting Bézier curve (solid blue line) and reconstructed feature points (blue stars); (bottom) convergence diagram.

numerical program *Matlab*, version 2015b. We remark that an implementation of the cuckoo search algorithm was already presented in [33]. However, our implementation follows a (more efficient) vectorized implementation freely available in [25], but adapted to the problem in this paper.

V. EXPERIMENTAL RESULTS

Our method has been applied to several examples of skin lesion images. In this paper we analyze only two of them because of limitations of space. The two examples correspond to a benign (a naevus) and a malignant (melanoma) skin lesions, respectively. They were obtained from the digital image archive of the Department of Dermatology of the University Medical Center of Groningen (The Netherlands).

First example is displayed in Figure 1: it corresponds to a dermoscopy image of a naevus (top), from which a collection of 223 feature points have been marked by a specialist. They are displayed as red empty circles superposed on the image for better visualization. We applied our method to this example by using Bézier curves of different degrees ranging from $n = 20$ to $n = 70$ with step-size 2, and then selecting the one minimizing the least-squares functional in Eq. (5). The best fitting Bézier curve obtained by our method, corresponding to $n = 56$, is displayed as a blue solid line in Fig. 1 (middle). The figure also displays the feature points reconstructed by our method as blue stars. As the reader can see, the method obtains a very good fitting of the data points. This observation is confirmed by our numerical results, where we obtain a value of $E = 5.4792$. We also computed the RMSE (root-mean square error), as:

$$RMSE = \sqrt{\frac{E}{\eta}} \quad (8)$$

and obtained a value of 1.5674×10^{-1} . We also noticed that the approximation is not optimal yet, as expected from an approximation method. In particular, the original data seems to be slightly more oscillating (see, for instance, the mid region of the bottom part of the curve). We remark, however, that a perfect matching between the original and the reconstructed features points is not actually required for clinical practice; the accuracy of our results (even if not totally optimal) is adequate for early diagnosis purposes. Figure 1(bottom) shows the convergence diagram of our method (i.e. the fitness function value E vs. the number of iterations) for this example.

Second example, corresponding to a real melanoma, is displayed in Figure 2. The interpretation of the figure is similar to Fig. 1 and hence omitted here to avoid redundancy. Geometrically, this example is simpler and only 77 feature points have been marked, as the skin lesion is smaller and with a smoother outline curve. The best fitting curve is obtained for $n = 25$ with an error value $E = 1.2213$ and RMSE of 1.2594×10^{-1} . Once again, the fitting curve approximates the data points with very good accuracy.

We carried out a comparison of our method with two classical techniques in the field: clustering [6] and thresholding [28]. The corresponding results are graphically shown in Fig. 3. The output of both methods is a polyline with a less smoother shape and several changes of concavity. As shown in Table II, they have a larger perimeter while covering a smaller area. We remark that a smooth large area (even in excess but not dramatically) is usually preferred in medical

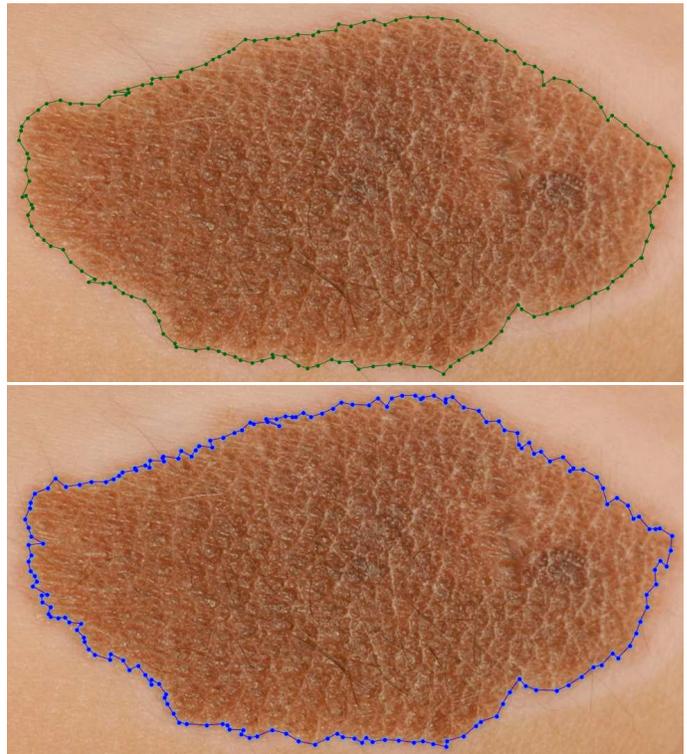


Fig. 3: Outline curve reconstruction of the lesion in Fig. 1 though two techniques: (top) clustering; (bottom) thresholding.

TABLE II: Comparative results for the example in Fig. 1.

Method	Perimeter	Area	RMSE error	CPU time (s.)
Clustering	32.051	59.706	1.9739×10^{-1}	686
Thresholding	35.514	57.762	2.1615×10^{-1}	833
Our method	29.497	63.128	1.5674×10^{-1}	931

settings to ensure that it encloses the lesion at full extent. They also have a worse fitting error compared to our method. We can conclude that our method outperforms both approaches in terms of smoothness and fitting error. Furthermore, this improvement is not at the expense of the computation time. Although our method takes longer than the other two methods, the CPU times are still of the same order, indicating that our method is quite competitive with two of the most classical state-of-the-art approaches.

VI. CONCLUSIONS AND FUTURE WORK

This paper computes a good approximation of the outline curve of skin lesions from dermoscopy images. Given a set of feature points selected by a specialist, the method applies a powerful nature-inspired metaheuristic optimization method called cuckoo search algorithm to obtain the free-form parametric Bézier curve that fits the points better in the least-squares sense. The method has been tested on two dermoscopy images of skin lesions with good results. Comparative analysis with two classical methods in the field show that our method

outperforms them in terms of smoothness and fitting error with no dramatic increase of the CPU times.

Main limitations of this method are related to the degree n of the fitting curve. A first problem is to determine the best value of n . Currently, we execute the algorithm for different values of n within a range and then select that minimizing either (5) or (8). However, we noticed that this process can lead to local self-intersecting shapes (loops) due to over-fitting, so we finally imposed the additional constraint of yielding a smooth shape with no loops. This process performs well but it is obviously time-consuming. On the other hand, the resulting Bézier curves tend to have a high degree, which is undesirable in practice. Finally, we rely on human intervention to determine the feature points by an expert dermatologist.

The present method can be improved in some ways. We plan to extend this approach to the cases of rational curves (which provide higher flexibility for shape reconstruction by involving extra degrees of freedom, the weights), and piecewise curves (e.g., cubic splines and B-splines), thus allowing us to select a low degree for the curve, a very desirable feature in practical applications. We also plan to analyze recent methods based on fusion thresholding [1] to obtain the feature points without human intervention to automate the whole process.

ACKNOWLEDGMENT

This research work is funded by projects: PDE-GIR of the European Union's Horizon 2020 programme, Marie Skłodowska-Curie grant agreement No 778035; #TIN2017-89275-R (Computer Science National Program) of the Spanish Agencia Estatal de Investigación and European Funds ERDF (AEI/FEDER); and #JU12, of public body SODERCAN and European Funds ERDF (SODERCAN/FEDER).

REFERENCES

- [1] Q. Abbas, M. E. Celebi, I. F. Garcia, M. Rashid: Lesion border detection in dermoscopy images using dynamic programming, *Skin Research and Technology*, **17**(1), 91–100 (2011).
- [2] A.A. Abbas, X. Guo, W.H. Tan, H.A. Jalab: Combined spline and B-spline for an improved automatic skin lesion segmentation in dermoscopy images using optimal color channel. *Journal of Medical Systems*, **38**, 80–80 (2014).
- [3] G. Argenziano, H.P. Soyer, V. De Giorgi: *Dermoscopy: A Tutorial*. EDRA Medical Publishing & New Media (2002).
- [4] Barhak, J., Fischer, A.: Parameterization and reconstruction from 3D scattered points based on neural network and PDE techniques. *IEEE Trans. on Visualization and Computer Graphics*, **7**(1) (2001) 1-16.
- [5] Barnhill, R.E.: *Geometric Processing for Design and Manufacturing*. SIAM, Philadelphia (1992).
- [6] M. E. Celebi, H. Iyatomi, G. Schaefer, W. V. Stoecker: Lesion border detection in dermoscopy images. *Computerized Medical Imaging and Graphics*, **33**(2), 148–153 (2009).
- [7] Dierckx, P.: *Curve and Surface Fitting with Splines*. Oxford University Press, Oxford (1993).
- [8] Engelbrecht, A.P.: *Fundamentals of Computational Swarm Intelligence*. John Wiley and Sons, Chichester, England (2005).
- [9] B. Erkol, R. H. Moss, R. J. Stanley, W. V. Stoecker, E. Hvatum: Automatic lesion boundary detection in dermoscopy images using gradient vector flow snakes. *Skin Research and Technology*, **11**(1), 17–26 (2005).
- [10] Farin, G.: *Curves and surfaces for CAGD (5th ed.)*. Morgan Kaufmann, San Francisco (2002).
- [11] Gálvez, A., Iglesias A.: Efficient particle swarm optimization approach for data fitting with free knot B-splines. *Computer-Aided Design*, **43**(12) (2011) 1683-1692.
- [12] Gálvez A., Iglesias A.: Firefly algorithm for explicit B-Spline curve fitting to data points. *Mathematical Problems in Engineering*, (2013) Article ID 528215, 12 pages.
- [13] Gálvez A., Iglesias A.: From nonlinear optimization to convex optimization through firefly algorithm and indirect approach with applications to CAD/CAM. *The Scientific World Journal*, (2013) Article ID 283919, 10 pages.
- [14] Gálvez A., Iglesias A.: New memetic self-adaptive firefly algorithm for continuous optimization. *International Journal of Bio-Inspired Computation*, **8**(5), 300–317 (2016).
- [15] Gálvez A., Iglesias A., Avila, A., Otero, C., Arias, R., Machado, C.: Elitist clonal selection algorithm for optimal choice of free knots in B-spline data fitting. *Applied Soft Computing*, **26** (2015) 90-106.
- [16] Gálvez, A., Iglesias, A., Cobo, A., Puig-Pey, J., Espinola, J.: Bézier curve and surface fitting of 3D point clouds through genetic algorithms, functional networks and least-squares approximation. *Lectures Notes in Computer Science*, **4706** (2007) 680-693.
- [17] Gu, P., Yan, X.: Neural network approach to the reconstruction of free-form surfaces for reverse engineering. *Computer-Aided Design* **27**(1) (1995) 59-64.
- [18] Hoffmann, M.: Numerical control of Kohonen neural network for scattered data approximation. *Numerical Algorithms*, **39**, (2005) 175-186.
- [19] Iglesias, A., Echevarría, G., Gálvez, A.: Functional networks for B-spline surface reconstruction. *Future Generation Computer Systems*, **20**(8) (2004) 1337-1353.
- [20] Jing, L., Sun, L.: Fitting B-spline curves by least squares support vector machines. In: *Proc. of the 2nd. Int. Conf. on Neural Networks & Brain*. Beijing (China). IEEE Press (2005) 905-909.
- [21] Jupp, D.L.B.: Approximation to data by splines with free knots. *SIAM Journal of Numerical Analysis*, **15** (1978) 328-343.
- [22] Kennedy, J., Eberhart, R.C., Shi, Y.: *Swarm Intelligence*. Morgan Kaufmann Publishers, San Francisco (2001).
- [23] Z. Ma, J.M. Tavares: A novel approach to segment skin lesions in dermoscopic images based on a deformable model. *IEEE Journal of Biomedical and Health Informatics*, **20**, 615–623 (2016).
- [24] D.A. Machado, G. Giraldo, A.A. Novotny: Multi-object segmentation approach based on topological derivative and level set method. *Integrated Computer-Aided Engineering*, **18**, 301–311 (2011).
- [25] Matlab Central repository: <http://www.mathworks.com/matlabcentral/fileexchange/29809-cuckoo-search-cs-algorithm> (last retrieved on May 6th 2018).
- [26] Park, H.: An error-bounded approximate method for representing planar curves in B-splines. *Computer Aided Geometric Design* **21** (2004) 479-497.
- [27] Park, H., Lee, J.H.: B-spline curve fitting based on adaptive curve refinement using dominant points. *Computer-Aided Design* **39** (2007) 439-451.
- [28] Sezgin, M., Sankur, B.: Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, **13**, 146–165 (2004).
- [29] Ulker, E., Arslan, A.: Automatic knot adjustment using an artificial immune system for B-spline curve approximation. *Information Sciences*, **179** (2009) 1483-1494.
- [30] Wang, W.P., Pottmann, H., Liu, Y.: Fitting B-spline curves to point clouds by curvature-based squared distance minimization. *ACM Transactions on Graphics*, **25**(2) (2006) 214-238.
- [31] Yang, X.-S.: *Nature-Inspired Metaheuristic Algorithms (2nd. Edition)*. Luniver Press, Frome, UK (2010).
- [32] Yang, X.-S.: *Engineering Optimization: An Introduction with Metaheuristic Applications*. Wiley & Sons, New Jersey (2010).
- [33] Yang, X.S., Deb, S.: Cuckoo search via Lévy flights. In: *Proc. World Congress on Nature & Biologically Inspired Computing (NaBIC)*. IEEE (2009) 210-214.
- [34] Yang, X.S., Deb, S.: Engineering optimization by cuckoo search. *Int. J. Math. Modelling and Numerical Optimization*, **1**(4) (2010) 330-343.
- [35] Yoshimoto, F., Moriyama, M., Harada, T.: Automatic knot adjustment by a genetic algorithm for data fitting with a spline. *Proc. of Shape Modeling International '99*, IEEE Computer Society Press (1999) 162-169.
- [36] Yoshimoto, F., Harada T., Yoshimoto, Y.: Data fitting with a spline using a real-coded algorithm. *Computer-Aided Design*, **35** (2003) 751-760.
- [37] H. Zhou, G. Schaefer, A. Sadka, M. E. Celebi: Anisotropic mean shift based fuzzy C-means segmentation of dermoscopy images. *IEEE Journal of Selected Topics in Signal Processing*, **3**(1) 26–34 (2009).