Segmentation of Mosaic Images Based on Deformable Models Using Genetic Algorithms

Alberto Bartoli, Gianfranco Fenu, Eric Medvet[✉], Felice Andrea Pellegrino, and Nicola Timeus

Department of Engineering and Architecture, University of Trieste, Trieste, Italy
`emedvet@units.it`

Abstract. Preservation and restoration of ancient mosaics is a crucial activity for the perpetuation of cultural heritage of many countries. Such an activity is usually based on manual procedures which are typically lengthy and costly. Digital imaging technologies have a great potential in this important application domain, from a number of points of view including smaller costs and much broader functionalities. In this work, we propose a mosaic-oriented image segmentation algorithm aimed at identifying automatically the tiles composing a mosaic based solely on an image of the mosaic itself. Our proposal consists of a *Genetic Algorithm*, in which we represent each candidate segmentation with a set of quadrangles whose shapes and positions are modified during an evolutionary search based on multi-objective optimization. We evaluate our proposal in detail on a set of real mosaics which differ in age and style. The results are highly promising and in line with the current state-of-the-art.

Keywords: Multi-objective optimization · Cultural heritage · Image processing

1 Introduction

A *mosaic* is a painting made with small pieces of stone, ceramic, glass or other similar materials of various shapes and colors (called *tiles* or *tessellae*) tied together by plaster or other binder to form geometrical or figurative decorative compositions. Mosaics have a long history, became widespread in ancient times and represent an essential component in the cultural heritage of many countries. Preservation and restoration of ancient mosaics is thus an important activity, usually requiring lengthy and costly procedures based on manual acquisition of the contour of each single tile over a semi-transparent paper superimposed to the mosaic.

Modern *digital imaging* technologies may be a great help in this respect, as tile contours could be collected much more quickly, more cheaply and more accurately with a suitable *image segmentation algorithm* applied to a digital image of the mosaic. In fact, a "digital model" of a mosaic may enable a full range of radically novel approaches and solutions in this area, ranging from

analysis of the distribution in large mosaics of tile shapes or of *filler* between tiles, to construction of 3D models to be browsed and zoomed interactively.

To the best of our knowledge, there is only one mosaic-oriented segmentation algorithm that has been proposed in the literature [1]. The performance of the cited work has been assessed and compared to general purpose segmentation algorithms recently [2]. In this work, we propose and evaluate a novel mosaic-oriented image segmentation method. Our proposal consists of an evolutionary procedure based on a *Genetic Algorithm* (GA) [3,4]. We represent each candidate segmentation as a fixed-size set of quadrangles and quantify its quality based on two indexes that can be calculated without knowing the ideal segmentation in advance. This property is essential in order to make our approach really practical. Such indexes assess the average color dissimilarity within each quadrangle (i.e., tile) and the average color similarity within adjacent quadrangles. We construct a set of different candidate segmentations and then modify such candidate segmentations stochastically for a predefined set of iterations, according to an evolutionary paradigm. We drive evolution with a multi-objective optimization algorithm aimed at minimizing the two indexes mentioned above and associated with each candidate segmentation.

We assess our proposal on 5 real mosaics which differ in age and style and have been used in [2]. We analyze the behavior of our proposal in different flavours, by varying the representation of a candidate segmentation, the criterion for choosing among the resulting candidate segmentations and the number of candidate segmentations. The results are highly promising and in line with the best result in the literature [1,2].

2   Related Work

Image segmentation amounts to partitioning the pixels into groups according to some criterion, e.g., color similarity. Another, more difficult to evaluate, partitioning criterion consists in grouping pixels depending on the object they belong to. Image segmentation is one of the oldest and most extensively studied problems in Computer Vision, dating back to the early seventies [5]. A general overview can be found, for instance, in [6,7].

One of the many possible approaches to image segmentation is based on *optimization*: the segmented image acts as the minimizer of a suitable objective function, depending on the original image and on the labeling of individual pixels in the segmented image itself. Choice of a suitable objective function is clearly crucial. Optimization-based segmentation methods may be classified depending on (i) the representation of the candidate solutions, and (ii) the optimization technique used for finding the minimizer.

The simplest representation for a candidate segmentation is a labeled image, i.e., a matrix containing the label assigned to each of the pixels. A more compact and widely employed kind of representation is based on *deformable models*. A taxonomy of deformable models in the context of optimization based image segmentation can be found in [8]. Roughly speaking, deformable models can

be thought of as flexible shapes that can be placed over the image and suitably adapted to match the regions in it. There may be either a fixed or a variable number of models and they may be defined either implicitly (for instance by means of the level sets of a proper function [9]) or explicitly (for instance as polygons whose vertices are decision variables [10]).

Concerning the optimization technique, the approaches used in practice rely on an objective function that is not convex and thus may be characterized by many local extrema. Furthermore, the solution space (i.e., the set of all possible segmentations for a given image) is huge and cannot be exhaustively explored. These facts motivate the resort to metaheuristics [11], which can be partitioned in (i) trajectory methods, in which the search process can be seen as the evolution in discrete time of a dynamical system; and, (ii) population-based methods, which iteratively modify a set of candidate solutions thus the search process can be seen as the evolution in discrete time of a set of points in the solution space.

The method presented in this work is based on a particular kind of deformable models, i.e., parametrized convex polygons, and a population-based metaheuristic, i.e., GA. An extensive survey of optimization-based segmentation methods based on deformable models and metaheuristics is reported in [8], while a review of segmentation methods based on GA can be found in [12].

3   Our Method

We consider the problem of the segmentation of a mosaic image, i.e., assigning a label to each pixel of the image: adjacent pixels in the segmented image with the same label belong to the same *region*. Ideally, regions in a mosaic image which has been correctly segmented exactly correspond to mosaic tessellae.

Formally, we denote by S the segmentation method, i.e., a function which assigns a label $S(p)$ to each pixel p of the image I. We assume that the range of S includes a special value $\varnothing$ which should be assigned to pixels which do not belong to any mosaic tessella, but instead correspond to the *filler*, i.e., the cemented network between tessellae.

The quality of a segmentation of a mosaic image can be assessed objectively by comparing it against a manual labeling of that image [2]. In detail, let $\mathcal{T}_I$ be the *ground truth* for a mosaic image I and let $S(I)$ be the segmentation obtained by applying a method S to I. Three objective indexes can be computed to compare $S(I)$ against $\mathcal{T}_I$, the average tile precision $\mathrm{Prec}(S(I), \mathcal{T}_I)$, the average tile recall $\mathrm{Rec}(S(I), \mathcal{T}_I)$, and the tile count error $\mathrm{Count}(S(I), \mathcal{T}_I)$:

$$\mathrm{Prec}(S(I), \mathcal{T}_I) = \frac{1}{|\mathcal{T}_I|} \sum_{T \in \mathcal{T}_I} \frac{\max_{R \in S(I)} |R \cap T|}{|R|} \tag{1}$$

$$\mathrm{Rec}(S(I), \mathcal{T}_I) = \frac{1}{|\mathcal{T}_I|} \sum_{T \in \mathcal{T}_I} \frac{\max_{R \in S(I)} |R \cap T|}{|T|} \tag{2}$$

$$\text{Count}(S(I), \mathcal{T}_I) = \frac{\text{abs}(|\mathcal{T}_I| - |S(I)|)}{|\mathcal{T}_I|} \tag{3}$$

where $|R|$ is the number of pixels which belong to a region R and $|R \cap T|$ is the number of pixels which belong both to R and T. In an optimal segmentation, precision and recall are equal to 1 and the tile count error is equal to 0. The former two indexes may be summarized in the F-measure index, which is the harmonic mean of precision and recall.

The goal of this work is to propose a new segmentation method which improves the state-of-the-art on mosaic images segmentation. We propose to use GA for segmenting mosaic images: a population of individuals—i.e., candidate segmentations—is iteratively evolved trying to maximize their quality. In next sections, we describe how we represent a candidate segmentation $S(I)$ in the GA framework and how we assess it; finally, we discuss other general GA-related choices.

3.1 Solution Representation

A segmentation $S(I)$ is a set of regions of the image: since we are interested in segmenting mosaic images, regions should be able to describe mosaic tessellae, which, in general, are convex polygons, often with 4 vertexes. For this reason, we chose a representation which consists of a fixed-size set of convex quadrangles $S(I) = \{q_1, \ldots, q_n\}$, defined by means of their position within the image I and their size. We explored two variants: one in which each quadrangle q of $S(I)$ is a "rotated rectangle" and one in which each quadrangle q is a "rotated and deformed square".

More in detail, in the rectangle-based representation, each quadrangle is defined as $q^{(i)} = (\Delta x^{(i)}, \Delta y^{(i)}, w^{(i)}, h^{(i)}, \phi^{(i)})$ where $\Delta x^{(i)}$ and $\Delta y^{(i)}$ are the offsets of the rectangle center of gravity with respect to a fixed point $(x^{(i)}, y^{(i)})$ in the Cartesian coordinate system of the image, $w^{(i)}$ and $h^{(i)}$ are the rectangle width and height, and $\phi^{(i)}$ is the angle of rotation of the rectangle. The domain of each component of the rectangle-based representation is the same for each rectangle and depends on a parameter s whose value has to be determined before the segmentation of the image I: s represents the size in pixel of an ideal average squared tessella of the mosaic and can be roughly estimated by a human operator who inspects the mosaic image. In particular, $\Delta x^{(i)}, \Delta y^{(i)} \in \left[-\frac{1}{2}s, \frac{1}{2}s\right]$, $w^{(i)}, h^{(i)} \in \left[\frac{1}{2}s, 2s\right]$ and $\phi^{(i)} \in \left[-\frac{1}{2}\pi, \frac{1}{2}\pi\right]$, for each i. The number n of rectangles in the segmentation and the reference position $(x^{(i)}, y^{(i)})$ of each rectangle are determined using s, the width w_I, and height h_I of the input image I, as follows. First, we determine the number $n_w = \left\lfloor \frac{w_I}{s} \right\rfloor$ and $n_h = \left\lfloor \frac{h_I}{s} \right\rfloor$ of rectangles along the x and y axes of the image, respectively. Then, we set $n = n_w n_h$, $x^{(i)} = ((i-1) \bmod n_w) \frac{w_I}{s} + \frac{1}{2}s$, and $y^{(i)} = \left\lfloor \frac{i}{n_w} \right\rfloor \frac{h_I}{s} + \frac{1}{2}s$, where $i \bmod n_w$ is the remainder of the division between i and n_w, and $\left\lfloor \frac{i}{n_w} \right\rfloor$ is the floor value of $\frac{i}{n_w}$. In other words, given s, a grid of $n_w \times n_h$ square cells is built on the image I and each $(x^{(i)}, y^{(i)})$ is the center of a grid cell.

In the deformed-square-based representation, each quadrangle is defined as $q^{(i)} = (\Delta x^{(i)}, \Delta y^{(i)}, \Delta x_1^{(i)}, \Delta y_1^{(i)}, \dots, \Delta x_4^{(i)}, \Delta y_4^{(i)}, \phi^{(i)})$ where $\Delta x^{(i)}$, $\Delta y^{(i)}$, and $\phi^{(i)}$ have the same meaning and domain as in the rectangle-based representation; similarly, the number n of quadrangles in the segmentation and their reference positions $(x^{(i)}, y^{(i)})$ are determined from s, w_I, h_I as described above. Concerning the other components, each pair $\Delta x_j^{(i)}, \Delta y_j^{(i)}$ represents the offsets of the jth quadrangle vertex with respect to the corresponding vertex of a square with the same center and with side size equal to s, before the rotation. We set a single domain for all those components which is $\left[-\frac{1}{4}s, \frac{1}{4}s\right]$, i.e., the actual position of each vertex of the quadrangle can be moved, in each direction, at most a quarter of side size away from its reference position. It can be proven that $\left[-\frac{1}{4}s, \frac{1}{4}s\right]$ is the largest domain for which the convexity for a quadrangle defined in this way is granted: despite the convexity of the region is not an intrinsic property of mosaic tessellae, we chose to impose this constraint because it allows a faster computation of aggregate features of pixels within the quadrangle, which is what we do for computing the fitness (see next section). By the way, our experience suggests that mosaic tessellae are in general convex.

3.2 Fitness Function

The quality of a candidate segmentation $S(I)$ can be expressed in terms of precision, recall, and count error. Those indexes can be computed only if a ground truth $\mathcal{T}_I$ is available for the image I. However, when segmenting a mosaic image in a real deployment, $\mathcal{T}_I$ is not available; hence, it follows that they cannot be used to drive the evolution.

In order to overcome this limitation, we use two other indexes as fitness: the *in-tile color dissimilarity* D_{in} and the *out-tile color dissimilarity* D_{out}. Intuitively, the former quantifies, for each region of the candidate segmentation, how different is the color among the region pixels; the latter quantifies, for each region, how different is the average color inside the region with respect to the average color of an external band around the region. If a region correctly superimposes a mosaic tessella on the image, the region in-tile color dissimilarity is low and the out-tile color dissimilarity is high. In order to drive the evolution considering the entire segmentation, D_{in} and D_{out} are the averages across all regions in $S(I)$:

$$D_{\text{in}} = \frac{1}{|S(I)|} \sum_{R \in S(I)} \left(\sigma_{L^*}^R + \sigma_{a^*}^R + \sigma_{b^*}^R\right)$$

$$D_{\text{out}} = \frac{1}{|S(I)|} \sum_{R \in S(I)} \sqrt{\left(\mu_{L^*}^R - \mu_{L^*}^{\bar{R} \setminus R}\right)^2 + \left(\mu_{a^*}^R - \mu_{a^*}^{\bar{R} \setminus R}\right)^2 + \left(\mu_{b^*}^R - \mu_{b^*}^{\bar{R} \setminus R}\right)^2}$$

where $\sigma_{L^*}^R$, $\sigma_{a^*}^R$, and $\sigma_{b^*}^R$ are the standard deviations of the L^*, a^* and b^* channel values of the pixels within the region R (in the CIE-Lab color space), $\mu_{L^*}^R$, $\mu_{a^*}^R$, and $\mu_{b^*}^R$ are the corresponding mean values, and $\bar{R}$ is a region which has the same center of gravity of R but is scaled by a factor $\beta = 1.2$—i.e., $\bar{R}$ is 20% larger

than R, hence the out-tile color dissimilarity D_{out} is computed considering the impact on color of the region expansion.

It can be noted that the objectives deriving from the two indexes are not strictly anti-correlated by design. On the other hand, there is a region of the search space in which D_{in} does not improve and D_{out} does worsen, corresponding to regions slightly smaller than the actual tile. Indeed, we observed experimentally that this trade-off is beneficial to the overall evolution.

3.3   GA Parameters and Best Selection

We used NSGA-II [13] as the actual GA-based optimization algorithm. Concerning the genetic operators used to generate new offspring, we used three operators: crossover, uniform mutation, and Gaussian mutation, respectively applied with a probability of 0.8, 0.1, and 0.1.

The crossover operator works as follows: given two segmentations $S'(I)$ and $S''(i)$, a new segmentation $S(I)$ is generated such that the ith component of the jth region of $S(I)$ has one of the two values of the corresponding ith components in the jth regions of the parents $S'(I)$ and $S''(i)$, with equal probability. Note that, in this way, the value of each component in $S(I)$ is granted to stay within the proper domain.

The uniform and Gaussian mutation operators work as follows: given a segmentation $S'(I)$, a new segmentation $S(I)$ is generated such that the ith component of the jth region of $S(I)$ has the same value of the corresponding component in $S'(I)$ with probability $1 - \frac{2}{|S'(I)|n_c}$ and has a new randomly generated value with probability $\frac{2}{|S'(I)|n_c}$, where n_c is the number of components in the representation for each region (i.e., $n_c = 5$ in the rectangle-based representation and $n_c = 11$ in the deformed-square-representation). The new value is chosen according to a uniform distribution within the component domain by the uniform mutation operator, and with Gaussian distribution by the Gaussian mutation operator. For the latter, the mean is set to 0 and the standard deviation is equal to $\frac{1}{10}$ of the component domain width—we set this value after preliminary experimentation. Values mutated with the Gaussian mutation operator are checked and possibly adjusted to stay within their domain. Note that the number of regions in the segmentation is never affected by the genetic operators.

Concerning the criterion for selecting the best individual upon the last generation, i.e., the actual proposed segmentation $S(I)$ for the image I, we explored two options, based on the two fitness function components. Considering the subset of the population consisting of the individuals belonging to the first Pareto front, we choose as best (a) the individual with the lowest in-tile color dissimilarity D_{in}, or (b) the individual with the greatest out-tile color dissimilarity D_{out}.

4   Experimental Evaluation

We were interested in investigating the effectiveness of our proposed segmentation method in general and with respect to its main design choices: the solution

representation (rectangle-based vs. deformed-square-based), the best individual selection criterion (min D_{in} vs. max D_{out}), and the population size n_{pop}.

To this end, we considered 5 mosaics which differ in age and style (see Fig. 1) and which have already been used in [2]. For all the 5 mosaic images, we obtained the corresponding ground truth segmentations.
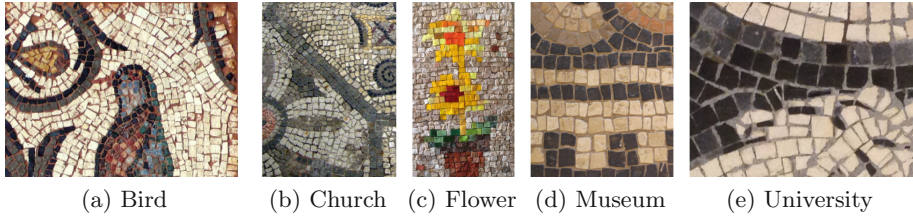

(a) Bird   (b) Church   (c) Flower   (d) Museum   (e) University

Fig. 1. The five mosaics of our experimental evaluation.

We performed several experiments: in each experiment, we applied our method to each mosaic image three times, by varying the random seed used in the evolutionary search. For each mosaic image, we set the parameter s—which represents the side length, in pixel, of an ideal squared tessella (see Sect. 3.1)—to $s = \sqrt{\frac{w_{\text{img}} h_{\text{img}}}{n_{\text{img}}}}$, where w_{img} and h_{img} are the width and height of the image, in pixels, and n_{img} is the number of tessellae in the corresponding grount truth segmentation.

Table 1 shows the main results of our experimentation obtained with $n_{\text{pop}} = 50$, deformed-square-based representation, and max D_{out} best selection criterion. The table shows the values, averaged across the three repetitions, of the objective quality indexes presented in Sect. 3: tile count error (Count), average tile precision (Prec), average tile recall (Rec), and F-measure (Fm). In order to provide a baseline, we also applied the state-of-the-art method, Tessella-oriented Segmentation (TOS) [1], to the same mosaics: concerning TOS, as suggested in [2], we set the main parameter α to the value which led to the best F-measure on each mosaic image—differently from our method, for which we used the same parameter values for all the mosaics.

The main finding of the results in Table 1 is that our method is, on the average, competitive with TOS. However, there is no clear winner between the two segmentation methods: the Count index is remarkably better for our method (0.03 vs. 0.33), whereas Fm is better for TOS (0.658 vs. 0.544). In general, hence, our method is better in capturing the number of tessellae, but is not particularly accurate in determining the boundaries of each tessella in the image.

A different point of view on the effectiveness of our method, beyond the analysis of objective indexes, is offered by Fig. 2, which shows two segmentations of the University mosaic image obtained with our method (Fig. 2a, in the variant of Table 1) and TOS (Fig. 2b). Again, a clear conclusion about the best method cannot be drawn. However, it can be seen that the polygon-based representation,

Table 1. Results of a selected variant of our method ($n_{\mathrm{pop}} = 50$, deformed-square-based representation, and max D_{out} best selection criterion) compared to TOS [1].

Mosaic	Our method				TOS			
	Count	Prec	Rec	Fm	Count	Prec	Rec	Fm
Bird	0.01	0.411	0.659	0.506	0.03	0.528	0.817	0.642
Church	0.03	0.418	0.629	0.502	0.54	0.564	0.719	0.632
Flower	0.07	0.503	0.626	0.558	0.06	0.494	0.678	0.572
Museum	0.03	0.503	0.760	0.605	0.14	0.644	0.873	0.741
University	0.03	0.459	0.674	0.546	0.90	0.632	0.785	0.701
Average	0.03	0.459	0.669	0.544	0.33	0.572	0.774	0.658

which our method bases on, allows to obtain a segmentation in which pixels deemed to belong to a tessella are clearly distinguishable from pixels deemed to belong to the filler, the criterion being their belonging to a polygon. On the other hand, such discrimination is not possible based on the segmentation obtained with TOS, in which every pixel belongs to a single region and that region provides no information about its nature (filler vs. tessella).

(a) Our method   (b) TOS
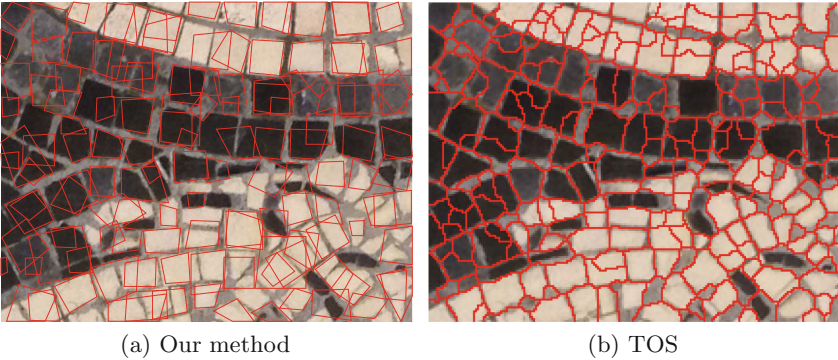
Fig. 2. Two segmentations of the University mosaic image obtained with our method (left) and TOS (right).

Tables 2 and 3 show how our main design choices affect the method effectiveness. The Count index is not shown in these tables since it depends only on s (which determines the number n of polygons in the segmentation) and is hence not affected by these design choices.

The impacts of different representations and best selection criteria are shown in Table 2. It can be seen that the representation slightly impacts on Fm: deformed-square-based representation scores 0.543, on average, vs. 0.539 with rectangle-based representation. Differently, the best selection criterion does not

affect the average Fm, whereas it can be seen that it does impact on Prec and Rec (largest Prec with min D_{in} and largest Rec with min D_{out}): this finding is consistent with the nature of D_{in} and D_{out} which are intrinsically related to precision and recall, respectively.

Table 2. Precision, recall, and F-measure obtained with $n_{\text{pop}} = 50$, different representations and best selection criteria.

Mosaic	Rectangle max D_{out}			Deformed-square min D_{in}			Deformed-square max D_{out}		
	Prec	Rec	Fm	Prec	Rec	Fm	Prec	Rec	Fm
Bird	0.404	0.669	0.504	0.406	0.654	0.501	0.406	0.655	0.502
Church	0.408	0.635	0.497	0.419	0.630	0.503	0.419	0.632	0.504
Flower	0.492	0.630	0.553	0.509	0.621	0.559	0.506	0.624	0.559
Museum	0.529	0.734	0.615	0.525	0.722	0.608	0.518	0.730	0.606
University	0.440	0.652	0.526	0.474	0.643	0.546	0.470	0.653	0.546
Average	0.455	0.664	0.539	0.467	0.654	0.543	0.464	0.659	0.543

Finally, Table 3 shows how the population size n_{pop} affects the effectiveness of our method: we experimented with three values (20, 50, and 100). The results suggest that the choice of $n_{\text{pop}} = 50$ is good: a significant decrease in Fm is obtained by reducing n_{pop} to 20, whereas no significant improvement appears to be achievable by doubling the population size. Moreover, in the latter case, the time needed to perform a segmentation (fourth column of each group) roughly doubles.

Table 3. Precision, recall, F-measure, and elapsed time (in s) obtained with different population sizes, the deformed-square-based representation, and the max D_{out} best selection criterion.

Mosaic	$n_{\text{pop}} = 20$				$n_{\text{pop}} = 50$				$n_{\text{pop}} = 100$			
	Prec	Rec	Fm	Time	Prec	Rec	Fm	Time	Prec	Rec	Fm	Time
Bird	0.379	0.671	0.484	2554	0.406	0.655	0.502	4824	0.411	0.659	0.506	12507
Church	0.391	0.643	0.486	2937	0.419	0.632	0.504	7482	0.418	0.629	0.502	16254
Flower	0.480	0.636	0.547	1990	0.506	0.624	0.559	5117	0.503	0.626	0.558	12864
Museum	0.490	0.728	0.585	59	0.518	0.730	0.606	158	0.503	0.760	0.605	409
University	0.416	0.672	0.514	54	0.470	0.653	0.546	156	0.459	0.674	0.546	346
Average	0.431	0.670	0.523	1519	0.464	0.659	0.543	3548	0.459	0.669	0.544	8476

5   Concluding Remarks

We have proposed and evaluated experimentally a novel mosaic-oriented image segmentation method. The method is completely unsupervised, in the sense that

does not require any ground truth, and is based on a GA which evolves a fixed-size set of candidate segmentations according to a multi-objective optimization algorithm. The performance indexes to be minimized during the search are a measure of the average color dissimilarity within each tile and of the average color similarity between adjacent tiles.

We have assessed our method on a set of real mosaics which differ in age and style, by using objective measures of segmentation quality (i.e., which require a ground truth associated with each pixel). The results are highly promising and in-line with the existing state-of-the-art.

We believe our method may still be improved and we plan to investigate, in particular, (a) a mechanism for taking into account the possible overlapping among regions of the candidate segmentation, (b) the possibility of varying the number of candidate tiles during a search dynamically, (c) the impact of suitable image preprocessing techniques on segmentation quality.

References

1. Benyoussef, L., Derrode, S.: Tessella-oriented segmentation and guidelines estimation of ancient mosaic images. J. Electron. Imaging **17**(4), 043014 (2008)
2. Fenu, G., Jain, N., Medvet, E., Pellegrino, F.A., Namer, M.P.: On the assessment of segmentation methods for images of mosaics. In: Proceedings of 10th International Conference on Computer Vision Theory and Applications, Institute for Systems and Technologies of Information, Control and Communication (2015)
3. Man, K.F., Tang, K.S., Kwong, S.: Genetic Algorithms: Concepts and Designs. Springer Science & Business Media, Heidelberg (2012)
4. Goldberg, D.E.: Genetic Algorithms. Pearson Education India, Delhi (2006)
5. Brice, C.R., Fennema, C.L.: Scene analysis using regions. Artif. Intell. **1**(3), 205–226 (1970)
6. Szeliski, R.: Computer Vision Algorithms and Applications. Springer, London (2011)
7. Forsyth, D.A., Ponce, J.: Computer Vision: A Modern Approach. Prentice Hall, Upper Saddle River (2003)
8. Mesejo, P., Ibáñez, O., Cordón, O., Cagnoni, S.: A survey on image segmentation using metaheuristic-based deformable models: state of the art and critical analysis. Appl. Soft Comput. **44**, 1–29 (2016)
9. Sethian, J.A.: Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science, vol. 3. Cambridge University Press, Cambridge (1999)
10. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: active contour models. Int. J. Comput. Vis. **1**(4), 321–331 (1988)
11. Glover, F.W., Kochenberger, G.A.: Handbook of Metaheuristics, vol. 57. Springer Science & Business Media, Heidelberg (2006)
12. Maulik, U.: Medical image segmentation using genetic algorithms. IEEE Trans. Inf Technol. Biomed. **13**(2), 166–173 (2009)
13. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J.J., Schwefel, H.-P. (eds.) PPSN 2000. LNCS, vol. 1917, pp. 849–858. Springer, Heidelberg (2000). doi:10.1007/3-540-45356-3_83