Available online at www.sciencedirect.com
ScienceDirect
Information Sciences 176 (2006) 3645–3657
INFORMATION SCIENCES
AN INTERNATIONAL JOURNAL
www.elsevier.com/locate/ins

A coloring fuzzy graph approach for image classification

D. Gómez [a], J. Montero [b,*], J. Yáñez [b]

[a] *Faculty of Statistics, Complutense University, 28040 Madrid, Spain*
[b] *Department of Mathematics, Complutense University, 28040 Madrid, Spain*

Received 2 December 2004; received in revised form 12 July 2005; accepted 29 January 2006

Abstract

One of the main problems in practice is the difficulty in dealing with membership functions. Many decision makers ask for a graphical representation to help them to visualize results. In this paper, we point out that some useful tools for fuzzy classification can be derived from fuzzy coloring procedures. In particular, we bring here a crisp *grey* coloring algorithm based upon a sequential application of a basic *black and white* binary coloring procedure, already introduced in a previous paper [D. Gómez, J. Montero, J. Yáñez, C. Poidomani, A graph coloring algorithm approach for image segmentation, Omega, in press]. In this article, the image is conceived as a fuzzy graph defined on the set of pixels where fuzzy edges represent the distance between pixels. In this way, we can obtain a more flexible hierarchical structure of colors, which in turn should give useful hints about those classes with unclear boundaries.

Keywords: Image classification; Decision making processes; Coloring problem

* Corresponding author.
E-mail addresses: dagomez@estad.ucm.es (D. Gómez), javier_montero@mat.ucm.es (J. Montero), jayage@mat.ucm.es (J. Yáñez).

0020-0255/$ - see front matter
doi:10.1016/j.ins.2006.01.006

1. Introduction

Remotely senses images quite often suggest classification techniques based upon fuzzy models. This is mainly the case when there are no *objects* to be classified. Objects, at least in a standard sense, present clear borders, and classification can be developed based solely upon a boundary analysis and a previous knowledge of the shapes of the different objects under consideration. On the contrary, many classification problems about earth land use, for example, refer to classes showing gradation from one class to the next class: there are no clear boundaries, and each class defines a fuzzy set with no particular shape (see [7]). In fact, there is increasing research on fuzzy sets applied to remote sensing classification problems (see, e.g., [14]). Many different approaches can be found in remote sensing classification literature. In [3,5], for example, the authors propose a classification model based upon a modified outranking model, basically taken from [21]. This approach requires decision makers to estimate quite a number of parameters, and therefore the learning process finally shown is unrealistic. Moreover, non-qualified decision makers find the output information difficult to manage. An important need is to develop fuzzy representation techniques. In particular, we postulate that it would be useful to develop some kind of coloring tool allowing a consistent and informative picture, in order to show possible regions and gradation between classes.

In this article, we propose an unsupervised crisp coloring methodology, to be considered as the basis for a fuzzy classification system in the sense of [2,4]. The coloring procedure we present here represents a pre-processing procedure, to be considered in a future classification, and is based upon a basic divisive crisp binary coloring process introduced in [15] in order to produce a segmentation algorithm for crisp graphs. We now translate this approach into a fuzzy context, considering that the image is described in terms of a fuzzy graph. In this way, we can search for consistent regions and postulate possible fuzzy classes.

The paper is organized as follows: in Section 2 we introduce the basic pixels fuzzy graph associated with an image; in Section 3 we present the coloring problem for fuzzy graphs, and an application to a standard image; in Section 4 we present the coloring algorithm, together with an application to some real images. Finally, final remarks are presented in Section 5.

2. The image and its associated pixels fuzzy graph

Our main objective is to produce a fuzzy-based coloring algorithm, so a meaningful picture can be offered to decision makers in order to identify possible fuzzy classes. A key argument, at least in land cover problems (see [1,3,5]), is that our fuzzy classes have a core of connected pixels, surrounded by pixels showing some degree of membership to that core. Hence, classification is made pixel by pixel, but behavior of surrounding pixels should have a strong influence at every stage. This information should be taken into account in a later supervised analysis where additional information may exist.

Let us consider an image as a bi-dimensional map of pixels, each one of them being characterized by a fixed number of measurable attributes. These attributes can be, for example, the values of the three bands of the visible spectrum (red, green and blue), the whole family of spectrum band intensities, or any other family of physical measurements. Hence, $P = \{(i,j)/1 \leqslant i \leqslant r, 1 \leqslant j \leqslant s\}$ will denote the set of pixel positions of an $r \times s$ image. If each pixel is characterized by b numerical measures, the whole image I can be characterized as $I = \{(x^1_{i,j}, \ldots, x^b_{i,j})/(i,j) \in P\}$. For a given image I, the information of the b measures of any pixel $(x^1_{i,j}, \ldots, x^b_{i,j})$ can be represented by its position, $p = (i,j) \in P$, without confusion. Given such an image I, a standard crisp classification problem pursues a partition in crisp regions, each being a subset of pixels, to be considered a candidate for a class (in case such a region is homogeneous enough). In this way, a crisp classification approach looks for a family of subsets of pixels $\{A_1, \ldots, A_c\}$ such that $P = \bigcup_{k=1}^{c} A_k$ but $A_i \cap A_j = \emptyset$, $\forall i \neq j$ where $A_1, \ldots, A_c$ are the family of crisp classes explaining the image.

Fuzzy uncertainty appears when we consider a dissimilarity measure between pixels in order to identify possible homogeneous regions in the image. In crisp image classification problems, the selection of an adequate distance is a difficult issue that has been studied by many authors. Obviously, any classification process will be strongly dependent on the selection of the appropriate distance, to be chosen taking into account all features of the image under consideration, together with our particular classification objectives. Furthermore, in many instances the distance between two elements includes some lack of precision or ambiguity (it would be the case, for example, when we consider the aggregation of several measures obtained by different experts: a well known problem in remote sensing is to chose an adequate distance in order to compare opinions from different experts). In order to capture the natural fuzzy uncertainty [23], and in order to give more flexibility to other already existing crisp classification procedures, we shall consider that a fuzzy distance (in the sense of [17]) expresses the relation between the measured properties of pixels, $d : P \times P \to \widetilde{[0, \infty)}$, where $\widetilde{[0, \infty)}$ will be here the set of fuzzy numbers with domain in R^+ [10] (see also [18,22]). We will denote by $\widetilde{d_{pp'}} = d(p, p')$ the fuzzy distance between the pixels p and p'. We will denote its membership function by $\mu_{pp'} : R^+ \to [0, 1]$ and by $\widetilde{D} = \{\widetilde{d_{pp'}}/(p, p') \in P \times P\}$ we will denote its associated fuzzy distance matrix.

Given a $r \times s$ image, a planar graph (P, E) can be defined considering P as the set of nodes and E the set of edges linking any couple of adjacent pixels. Two pixels $p = (i,j)$, $p' = (i',j') \in P$ are adjacent if $|i - i'| + |j - j'| = 1$, i.e., if they share one coordinate being the other one contiguous. Let us now denote

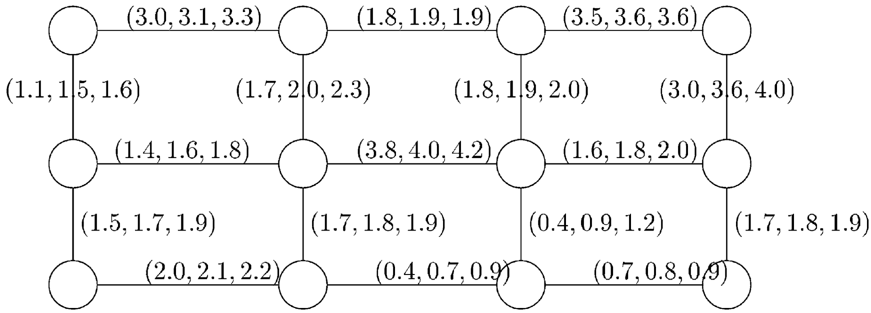

Fig. 1. Pixels fuzzy graph of Example 2.1.

by $\widetilde{G(I)} = (P, \widetilde{E})$ the graph associated to our image I, where $\widetilde{E} = \{\widetilde{d_{pp'}}/(p, p')$ adjacents$\}$ where $\widetilde{d_{pp'}}$ are fuzzy numbers with domain in R^+.

Definition 2.1. Given the image I and a fuzzy distance d, the *pixels fuzzy graph* is defined as the pair $\widetilde{G(I)} = (P, \widetilde{E})$.

Notice that our pixels fuzzy graph $\widetilde{G(I)}$ can be also characterized by the set P plus two $r \times s$ fuzzy matrices, $\widetilde{D^1}$ and $\widetilde{D^2}$, where $\widetilde{D^1_{i,j}} = d((i,j),(i+1,j))$, $\forall (i,j) \in \{1,\ldots,r-1\} \times \{1,\ldots,s\}$; and $\widetilde{D^2_{i,j}} = d((i,j),(i,j+1)), \forall (i,j) \in \{1,\ldots,r\} \times \{1,\ldots,s-1\}$. Since our coloring procedure will be based upon this alternative representation, from now on we shall denote our pixels fuzzy graph $\widetilde{G(I)}$ by $(r, s, \widetilde{D^1}, \widetilde{D^2})$.

Example 2.1. Let $(r, s, \widetilde{D^1}, \widetilde{D^2})$ be a pixels fuzzy graph with $r = 3$, $s = 4$ and

$$\widetilde{D^1} = \begin{pmatrix} (1.1, 1.5, 1.6) & (1.7, 2.0, 2.3) & (1.8, 1.9, 2.0) & (3.0, 3.6, 4.0) \\ (1.5, 1.7, 1.9) & (1.7, 1.8, 1.9) & (0.4, 0.9, 1.2) & (1.7, 1.8, 1.9) \end{pmatrix}$$

$$\widetilde{D^2} = \begin{pmatrix} (3.0, 3.1, 3.3) & (1.8, 1.9, 1.9) & (3.5, 3.5, 3.6) \\ (1.4, 1.6, 1.8) & (3.8, 4.0, 4.2) & (1.6, 1.8, 2.0) \\ (2.0, 2.1, 2.2) & (0.4, 0.7, 0.9) & (0.7, 0.8, 0.9) \end{pmatrix}$$

where each triple (a, b, c) represents a triangular fuzzy number. Such a pixels fuzzy graph is depicted in Fig. 1. The key coloring algorithm proposed in the next section will take advantage of the above alternative representation, which shows the relation between adjacent pixels in the pixels fuzzy graph $\widetilde{G(I)}$.

3. The coloring problem for fuzzy graphs

In the crisp framework, a c-coloring of a graph $G = (V, E)$ (see, e.g., [20]) is a mapping $C: V \to \{0, \ldots, c-1\}$, verifying $C(v) \neq C(v')$ if $\{v, v'\} \in E$. Any

c-coloring induces a crisp classification of the nodes set V, being each class associated to one color: $V_C(k) = \{v \in V/C(v) = k,\ k \in \{0,\ldots,c-1\}$. As a particular case a binary coloring $col: V \to \{0,1\}$.

Our objective is to obtain a classification of pixels through a c-coloring C of the pixels fuzzy graph $\widetilde{G(I)}$: the pixel $(i,j) \in P$ will be classified as $k \in \{0,\ldots,c-1\}$ if its color is $C(i,j) = k$. The coloring problem for fuzzy graphs has been studied by some authors (see [19]). And in order to color a fuzzy graph, we will consider the fuzzy edges that are greater than a prescribed threshold. The problem of ordering fuzzy numbers has been studied by many authors (see, e.g., [11,12,22]). An interesting approach is to transform fuzzy numbers into real numbers by means of a ranking function (see [11]).

In this paper we will say that a fuzzy set $\widetilde{A}$ with membership function μ_A and domain in R^+ is considered as a fuzzy number if and only if (see [22])

- $A_\alpha = \{x \in R^+$ such that $\mu_A \geqslant \alpha\}$ is a convex set, denoted by $[\underline{A_\alpha}, \overline{A_\alpha}]$;
- μ_A is an upper semicontinuous function;
- $\widetilde{A}$ is normal, i.e. there exists $x \in R^+$ such that $\mu_A(x) = 1$;
- $\mathrm{supp}(A) = \{x/\mu_A(x) > 0\}$ is a bounded set of R^+.

Definition 3.1. Let $\aleph$ be the set of fuzzy numbers and let $a, b \in \aleph$. Then $a \widetilde{\geqslant} b \leftrightarrow F(a) \geqslant F(b)$.

Example 3.1. Let a be a triangular fuzzy number in $\aleph$. We will identify a with the triplet (a_1, a_2, a_3). $F_1(a) = \frac{a_1+a_2+a_3}{3}$, $F_2(a) = \sqrt{\int_0^1 g(t)}$ where $g(t) = (c_1 + tc_2)^2 + c_3(1-t)$, $c_1 = \frac{a_1+a_3}{2}$, $c_2 = a_2 - \frac{a_1+a_3}{2}$ and $c_3 = \frac{(a_3-a_1)^2}{12}$, are possible examples of ranking functions.

3.1. The basic binary coloring procedure

A natural way of introducing our basic binary coloring procedure is to classify two adjacent pixels, either as 0 or 1 depending on the fuzzy distance between their properties, if such a distance is lower or greater than a prescribed threshold α (see [15]). In this way, adjacent pixels are classified as *distinct* whenever such distance between them is *high* (notice that a standard approach classifies two arbitrary pixels in the same class if such a distance is *low*, no matter if they are adjacent or not). Once such a threshold α has been fixed, the first binary coloring can be obtained assigning an arbitrary color ("0" or "1") to an arbitrary pixel, and then fix an order in which all neighboring pixels will be colored. The pixel to be colored in first place could be, for example, the pixel with coordinates $(1,1)$ in the left top corner of the image. Then pixels will be colored from left to right and from top to bottom depending on value α in the following way:

$$col(i+1,j) = \begin{cases} col(i,j), & \text{if } \widetilde{d^1_{i,j}} \widetilde{\geqslant} \alpha \\ 1 - col(i,j), & \text{otherwise} \end{cases}$$

$$col(i,j+1) = \begin{cases} col(i,j), & \text{if } \widetilde{d^2_{i,j}} \widetilde{\geqslant} \alpha \\ 1 - col(i,j), & \text{otherwise} \end{cases}$$

In general, given a pixel (i,j) already colored, adjacent pixels $(i+1,j)$ and $(i,j+1)$ will be subsequently colored. Since pixel $(i+1,j+1)$ can be alternatively colored either from pixel $(i+1,j)$ or pixel $(i,j+1)$, a natural constraint is that both coloring must produce the same color in both paths. Otherwise, the coloring will be *inconsistent*.

Definition 3.2. Given the set of pixels P, a *square* is a subset of four pixels

$$sq(i,j) \equiv \{(i,j); (i+1,j); (i,j+1); (i+1,j+1)\}$$

being $i \in \{1,\ldots,r-1\}$ and $j \in \{1,\ldots,s-1\}$.

We shall denote by PS the set of all *squares*,

$$PS = \{sq(i,j)/i \in \{1,\ldots,r-1\}, j \in \{1,\ldots,s-1\}\}$$

Definition 3.3. Given a pixels fuzzy graph $(r,s,\widetilde{D^1},\widetilde{D^2})$, a square $sq(i,j) \in PS$ is *consistent* at level α if given an arbitrary color $col(i,j)$, the above binary coloring procedure assigns the same color to pixel $(i+1,j+1)$, no matter if it is done from pixel $(i,j+1)$ or pixel $(i+1,j)$. Otherwise, the pixel square is *inconsistent*.

Consequently, the above binary coloring of pixels depends on the chosen threshold value α, and we have two extreme cases

$$\bar{\alpha} = \inf\{\alpha/\widetilde{d_{p,p'}} \widetilde{\leqslant} \alpha \ \forall (p,p') \in P\}$$

and

$$\underline{\alpha} = \sup\{\alpha/\alpha \widetilde{\leqslant} \widetilde{d_{p,p'}} \ \forall (p,p') \in P\}$$

Hence, if we fix a threshold $\alpha > \bar{\alpha}$, then the whole picture is considered as a unique class, $col(i,j) = col(1,1)\ \forall (i,j) \in P$. And in case $\alpha < \underline{\alpha}$, the picture looks like a chess board, being all adjacent pixels alternatively classified as "0" and "1" (only the interval $[\underline{\alpha},\bar{\alpha}]$ should be properly considered). Indeed, determining an appropriate intermediate α level is not a trivial task. But once a level α is given, inconsistent squares can be found (see [15] for computational details about the complete crisp algorithm). Then we can consider the whole family of thresholds allowing consistency of our fuzzy graph.

Definition 3.4. Given a value α, the pixels fuzzy graph $(r, s, \widetilde{D^1}, \widetilde{D^2})$ is *consistent at level* α if all squares $sq(i,j) \in PS$ are consistent at level α.

Definition 3.5. Given a pixels fuzzy graph $(r, s, \widetilde{D^1}, \widetilde{D^2})$, its *consistency level*, denoted as α^*, is the maximum value $\alpha \in [\underline{\alpha}, \bar{\alpha}]$ for which the fuzzy graph is consistent.

Existence of such a consistency level α^* is always assured, at least while our image contains a finite number of pixels. If some inconsistency is detected for a certain value α, a decreasing procedure can be introduced in order to find a lower level α^* assuring consistency. Such a procedure can be initialized with $\alpha^* = \bar{\alpha}$, and then we can search among inconsistent pixels $sq(i,j)$, by means of a function *newalpha* (see again [15] for computational details). After the first binary coloring, pixels will be classified either as "0" or "1".

A new iteration can be then defined by applying the same binary coloring procedure, so we can get a more precise color for both families (family "0" will switch either into "00" or "01"). This will be done by alternatively activating elements in only one of those families already colored in the previous stage. Subsequently, new iterations can be considered by applying such a binary coloring process to all those activated pixels under consideration, at each stage (a subset of pixels $P' \subset P$).

3.2. Application to a standard MATLAB image

The above algorithm has been applied, as a first experiment, to a standard image taken from the MATLAB package (a water colored tree, already considered by in [3], reproduced here as a *grey* picture (see Fig. 2)).

In order to model distance uncertainty, in this example we consider symmetric triangle fuzzy numbers for the fuzzy graph, based on the Euclidean distance,

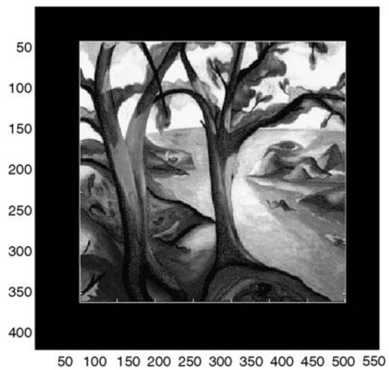

Fig. 2. Original MATLAB image.

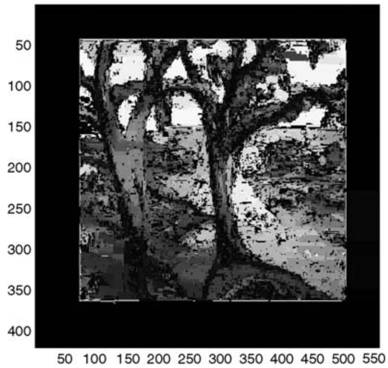

Fig. 3. MATLAB image classified with $it_M = 2$.

$d_{p,p'} = (d - \mathrm{er}, d, d + \mathrm{er})$, where $d = \sqrt{\sum_{k=1}^{b} (x_p^k - x_{p'}^k)^2}$ $\forall (p,p') \in P$ is the deterministic Euclidean distance and "er" is the error measure considered by the expert. Following [11,12], every ranking function over a triangular fuzzy number $a = (a_1, a_2, a_3)$ must verify that $F(a) \in \mathrm{supp}(a)$ and F is non-decreasing with respect to those parameters a_1, a_2 and a_3 (see [18]). For example, we can consider $F(a) = \sqrt{a_x^2 + a_y^2}$, where a_x and a_y are, respectively, the horizontal and vertical coordinates of the centroid of a, i.e.,

$$a_x = \frac{a_1 + a_2 + a_3}{3}, \qquad a_y = \frac{1}{3}\left(1 + \frac{2a_2}{a_1 + 2a_2 + a_3}\right)$$

Hence, a sequence of binary classifications has been obtained, so a binary number is associated with each pixel and the image is divided into regions of adjacent pixels, all of them with the same associated number. In order to be able to visualize these regions, we have painted each region with a color in the RGB color space, obtained as the mean of the original color of pixels in each region (notice that this is indeed a limitation, because we pursue a graphical visualization, and therefore are restricted to three-dimensional painting techniques). In this way we can produce a sequence of *paintings* showing successive partitioning. We show in Fig. 3 the segmentation obtained for the MATLAB image with only two iterations, i.e., four colors.

3.3. Computational complexity

Although the basic coloring procedures are polynomial, the number of color classes grows exponentially, 2^{it-1}, being *it* the number *it* of iterations. In the worst case, the number *it* of iterations reaches the value $r \times s - 1$ (and then all colors except one are void). Consequently, the algorithm is not polynomial.

However, the above coloring algorithm is very inefficient from a computational point of view, when handling real images. An appropriate decreasing scheme of parameter α, allowing an acceptable ratio of inconsistent pixel squares, will be the core of the relaxed coloring algorithm we introduce in the next section.

4. The coloring algorithm

When dealing with real medium size images (not more than 50×50 pixels), the above decreasing procedure for computing α^* may reach very small values so the final image will most probably look like a chess board. The above algorithm needs some improvement in order to avoid such a misbehavior. In particular, we propose to relax consistency constraint in order to produce a new relaxed binary coloring procedure, allowing some few inconsistencies (lets say one percent of squares). Computational complexity problems can be addressed by fixing a small number of iterations *it*, 4 or 5 for example (our computational experience suggests that good enough results could be obtained after some limited number of iterations).

Let *incratio* be defined as the ratio of inconsistencies of a binary coloring process (see [15]), i.e., the number of inconsistent pixels divided by $(r-1)(s-1)$, the total number of squares. On one hand, large values of *incratio* are associated with a low number of classes (the value α does not needs to be decreased). On the other hand, the greater this value *incratio*, the greater the number of pixels with a *wrong* color. Hence, we can look for some compromise between these two arguments. In our experiments, such a compromise appears with small inconsistency ratios, around 0.01. Each inconsistent pixel should be then *isolated* so that its inconsistency does not induce inconsistency of adjacent pixels.

Given any binary coloring *col* (see again [15]), then we proceed to assign new colors to every pixel, still following the same order already considered in the previous binary algorithm (left to right and top to bottom). Again, after the coloring process of first row, from left to right, inconsistencies cannot be detected, neither first vertical coloring. Inconsistencies, if any, will be detected when a *horizontal* arc is being checked with an already *vertically* colored arc. Let ninc be the total number of those inconsistent squares.

Any inconsistent pixel can be arbitrarily colored. Subsequent iterations will smooth the effects of such arbitrary coloring. In order to bound the CPU time, the number of iterations *it* cannot be greater than a fixed value, let it_M be such a value. The number of color classes is then equal to 2^{it_M} (computational experiences till now suggest that it_M equal to 3 or 4 will often be enough). Once the value of it_M has been fixed, different values of α must be selected: let $\{\alpha_{it}\}$ the family of these selected values, $it \in \{1, \ldots, it_M\}$ (these values must be appropriately selected for any particular image after some computational experiences).

The associated relaxed coloring algorithm will be based upon the successive relaxed binary coloring at different levels α_{it}.

4.1. Application to a remote sensing image

The above algorithm has been applied to a real image (see Fig. 4): an ortho-image of Sevilla Province (south Spain) that was taken on August 18, 1987, by the LANDSAT 5 satellite (Worldwide Reference System Spain (WRS) image 202-34-4). This image was taken with the Thematic Mapper sensor, which has a spatial resolution of 30 m (see [5] for a detailed description of this image).

In this case we have consider two distances (Euclidean and Manhattan), normalized in order to build a fuzzy distance. In this example we consider tetrahedral fuzzy numbers, that can be characterized by four numbers

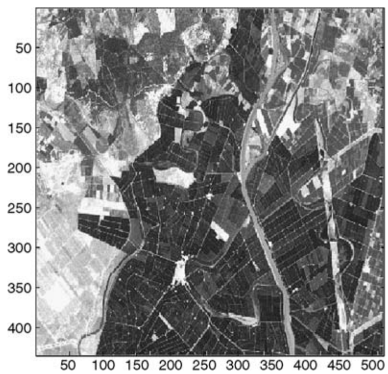

Fig. 4. Sevilla image.

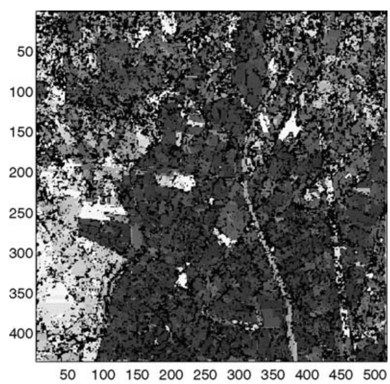

Fig. 5. Sevilla image classified with $it_M = 2$.

(a_1, a_2, a_3, a_4) (see [18]). The chosen ranking function for this example has been $F(a) = \sqrt{a_x^2 + a_y^2}$, where a_x and a_y are respectively, the horizontal and the vertical coordinates of the centroid of a. The coloring obtained by means of our relaxed algorithm taking $it_M = 2$ is shown in Fig. 5. As already pointed out, few iterations were needed in order to capture main features.

The above algorithm has been applied to several standard images, and with few iterations we obtain similar results to those analyzed in [15]. But the approach presented in this paper is ready to be implemented in other problems where the distance is not deduced from crisp measurements (for example, in order to get an aggregated classification from several proposals obtained with different classification methods).

5. Final comments

The final objective of the algorithm we propose in this article is to show decision maker several possible pictures of the image, each one obtained by means of an automatic coloring procedure of each pixel based upon a particular distance. Such a coloring procedure takes into account behavior of each pixel with respect to its surrounding pixels, and each color will suggest a possible class. Of course, an interesting subsequent improvement would be to consider a wider concept of neighboring, from the four surrounding pixels considered in [15] into eight surrounding pixels (considering far beyond adjacent pixels may be required when dealing with extremely *smooth* images).

Our coloring process is based upon a sequential binary procedure initially designed for crisp segmentation (see [15] for more details), which in this paper has been translated into a fuzzy context, representing uncertainty about dissimilarity between pixels by means of fuzzy numbers. These improvements allow a more flexible approach to an indeed complex problem.

Moreover, our approach pursues a nested family of regions each one suggesting a possible class, and the associated colored pictures can be afterwards analyzed by decision makers within a proper classification procedure, where certain *homogeneous* regions can be identified (see [19] for alternative procedures in order to paint fuzzy graphs). A subsequent comparison may lead to a fuzzy classification, if we are able to evaluate, by means of statistical tools (as proposed in [1]) the degree of concordance of each pixel to each one of those identified regions. Of course, other complementary techniques may be also considered (see, e.g., [6,8,9,13,16]).

As any fuzzy representation technique, the tool presented in this paper offers decision makers an additional decision making aid allowing a more accurate description of images involving fuzzy classes. Of course some learning will be always needed, but our hierarchical output offers a systematic sequence of colored images that can be carefully analyzed by decision makers for a more global understanding of the image (depending on their objectives and abilities). There is an absolute need for manageable descriptive tools in order to show fuzzy uncertainty.

Acknowledgments

This research has been partially supported by grants BFM2002-0281 and MTM2005-08982-C04, from the Government of Spain. We also thank one of the anonymous referees for some extremely interesting suggestions for future research.

References

[1] A. Amo, D. Gómez, J. Montero, G. Biging, Relevance and redundancy in fuzzy classification systems, Mathware and Soft Computing 8 (2001) 203–216.
[2] A. Amo, J. Montero, V. Cutello, On the principles of fuzzy classification, in: R.N. Dave, T. Sudkamp (Eds.), Proceedings NAFIPS Conference, IEEE Press, Piscataway, NJ, 1999, pp. 675–679.
[3] A. Amo, J. Montero, G. Biging, Classifying pixels by means of fuzzy relations, International Journal on General Systems 29 (2000) 605–621.
[4] A. Amo, J. Montero, G. Biging, V. Cutello, Fuzzy classification systems, European Journal of Operational Research 156 (2004) 459–507.
[5] A. Amo, J. Montero, A. Fernández, M. López, J. Tordesillas, G. Biging, Spectral fuzzy classification: an application, IEEE Transactions on Systems Man and Cybernetics (C) 32 (2002) 42–48.
[6] S. Bandyopadhyay, U. Maulik, An evolutionary technique based on k-means algorithm for optimal clustering in $\Re^N$, Information Sciences 108 (1998) 219–240.
[7] J.C. Bezdek, J.D. Harris, Fuzzy partitions and relations: an axiomatic basis for clustering, Fuzzy Sets and Systems 1 (1978) 111–127.
[8] H.J. Caulfield, J. Fu, S. Yoo, Artificial color image logic, Information Sciences 167 (2004) 1–7.
[9] H.D. Cheng, M. Miyojim, Automatic pavement distress detection system, Information Sciences 108 (1998) 219–240.
[10] D. Dubois, H. Prade, Fuzzy Sets and Systems, Theory and Applications, Academic Press, New York, 1980.
[11] D. Dubois, H. Prade, Ranking fuzzy numbers in the setting of possibility theory, Information Sciences 30 (1983) 183–224.
[12] G. Facchinetti, R.G. Ricci, A characterization of a general class of ranking functions on triangular fuzzy numbers, Fuzzy Sets and Systems 146 (2004) 297–312.
[13] D. Feng, S. Wenkang, C. Liangzhou, D. Yong, Z. Zhenfu, Infrared image segmentation with 2-D maximum entropy method based on particle swarm optimization, Pattern Recognition Letters 26 (2005) 597–603.
[14] G.M. Foody, The continuum of classification fuzziness in thematics mapping, Photogrammetric Engineering and Remote Sensing 65 (1999) 443–451.
[15] D. Gómez, J. Montero, J. Yáñez, C. Poidomani, A graph coloring algorithm approach for image segmentation, Omega, in press.

[16] N. Ito, R. Kamekura, Y. Shimazu, T. Yokoyama, Y. Matsushita, The Combination of edge detection and region extraction in nonparametric color image segmentation, Information Sciences 92 (1996) 277–294.
[17] O. Kabva, S. Seikkala, On fuzzy metric spaces, Fuzzy Sets and Systems 12 (1984) 215–229.
[18] A. Kaufman, M. Gupta, Introduction to Fuzzy Arithmetic, Van Nostrand Reinhold, New York, 1985.
[19] S. Muñoz, M.T. Ortuño, J. Ramirez, J. Yáñez, Coloring fuzzy graphs, Omega 33 (2005) 211–221.
[20] P.M. Pardalos, T. Mavridou, J. Xue, The graph coloring problem: a bibliographic survey, in: D.Z. Du, P.M. Pardalos (Eds.), Handbook of Combinatorial Optimization, Kluwer Academic Publishers, Boston, 1998, pp. 331–395.
[21] J. Montero, A.D. Pearman, J. Tejada, Fuzzy multicriteria decision support for budget allocation in the transport sector, Top 3 (1995) 47–68.
[22] X. Wang, E.E. Kerre, Reasonable properties for the ordering of fuzzy quantities (I), Fuzzy Sets and Systems 118 (3) (2001) 375–385.
[23] L.A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.