# GRAPH COLORING INCONSISTENCIES IN IMAGE SEGMENTATION

J. YÁÑEZ*, S. MUÑOZ, J. MONTERO

*Faculty of Mathematics, Complutense University of Madrid,*
*28040 Madrid, Spain*
**E-mail: javier_yanez@mat.ucm.es*

In this paper we analyze the structural properties of the inconsistencies detected by the crude algorithm for segmentation of digital images introduced by some of the authors in a previous work. Such analysis will suggest an alternative algorithm for image segmentation.

*Keywords*: Segmentation techniques; Graph Theory; Spanning trees; Decision support systems.

## 1. Introduction

Classical segmentation of digital images can be quite often assimilated to a search for *objects* whenever clear borders exist, i.e., sudden changes when we move from pixel to pixel. Some classification and optimization problems can be modeled as graph coloring problems. In particular, Gómez *et al.*[1] introduced a coloring algorithm for segmentation, susceptible of being considered as a first stage for a posterior classification analysis, specially in the presence of transition zones between classes. This algorithm is based upon the successive application of a basic binary procedure which will produce a nested structure of regions, with a precision to be fixed by decision makers according to their abilities and final objectives.

This paper is organized as follows: in Sec. 2 we outline the crude algorithm for segmentation of digital images proposed in Gómez *et al.*[1]; it requires certain consistency properties which are not verified in general by medium and big size images. A theoretical analysis of the possible inconsistencies that can occur is presented in Sec. 3.

## 2. The image as a pixel network. Crude algorithm

The structural information about the image $I$ under study can be represented in a nondirected graph $G(I)$ whose set of nodes is the pixel set $P_I = \{(i,j) \mid i \in \{1,\ldots,r\},\ j \in \{1,\ldots,s\}\}$, meaning that we are dealing with an image of size $r \times s$. Each pixel is characterized by $b$ attributes ($b = 3$ for the visible spectrum). Two pixels are said to be *adjacent* if one of their coordinates is the same and the other coordinate differs by one. Consequently, we define the set of edges of $G(I)$ as

$$E_I = \{\{(i,j),(i',j')\} \in P_I \times P_I \mid [i = i', |j - j'| = 1] \vee [j = j', |i - i'| = 1]\}.$$

A standard segmentation problem pursues a *nice* partition of $I$ in color regions. This partition will be based upon a distance $d : E_I \longrightarrow [0,\infty)$ on the $b$ measured attributes of the pixels. Of course, an *ad hoc* distance should be defined for each specific problem. In this way, the *pixel network* associated to $I$ is defined as $N(I) = \big(G(I), \{d(e)\}_{e\in E_I}\big)$.

**Example 2.1.** *Consider an image $I$ with $r = 3$, $s = 4$ and associated pixel network $N(I) = (G(I), \{d(e)\}_{e\in E_I})$, where $G(I) = (P_I, E_I)$ (the values of $\{d(e)\}_{e\in E_I}$ are shown in Fig. 1).*

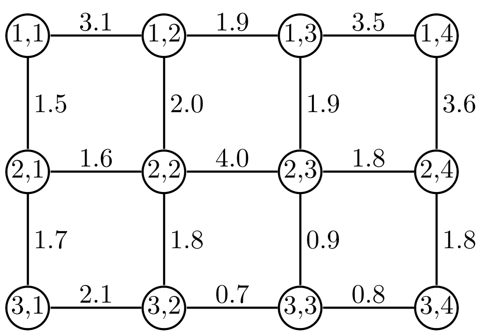


Fig. 1. Pixel network $N(I) = (G(I), \{d(e)\}_{e\in E_I})$

The algorithm proposed in Gómez *et al.*[1] obtains a partition of $P_I$ through a $c$-coloring $C^*$ defined on $P_I$, in such a way that a color $C^*(p) \in \{0,\ldots,c-1\}$ is assigned to each pixel $p \in P_I$. This $c$-coloring $C^*$ is constructed through an iterative basic binary coloring process, leading to a hierarchical coloring of the pixels. The first binary coloring assigns to each pixel $p \in P_I$ either the color "0" or the color "1". A second binary coloring can then be applied, separately, to each connected component of the subgraph of $G(I)$ generated by those pixels previously colored as "0" (obtaining the colors "00" and "01"), and to each connected component of the subgraph of $G(I)$ generated by those pixels previously colored as “1” (obtaining the colors “10” and “11”). Repeating $t$ times this procedure, a $c$-coloring $C^*$ will be defined on $P_I$, where $c = 2^t$. For instance, if some pixel $p \in P_I$ has been colored three successive times as “1”, “0” and “1”, then, taking into account that 5 is the decimal representation of 101, it will be $C^*(p) = 5$.

The binary coloring procedure starts by selecting a threshold $\alpha$ (the choice of this value will be analyzed below) and a pixel $p_0 \in P_I$, and setting $C(p_0) = 0$. Then, for each colored pixel $p$, its noncolored adjacent pixels $p'$ are colored in the following way:

$$C(p') = \begin{cases} C(p) & \text{if } d(p,p') < \alpha \\ 1 - C(p) & \text{if } d(p,p') \geq \alpha \end{cases}$$

Let $\overline{G} = (\overline{P}, \overline{E})$ be a connected component of the subgraph of $G(I)$ considered at the current stage of the coloring process (initially, we consider the graph $G(I) = (P_I, E_I)$), and let $\overline{N} = (\overline{G}, \{d(e)\}_{e \in \overline{E}})$ be its associated pixel network.

If $\overline{G}$ is acyclic, the binary coloring procedure is correct in the sense that every pixel will be uniquely colored; otherwise, a pixel contained in some cycle could be colored with “0” or “1” depending on the pixel used to reach it.

Given a threshold $\alpha$ and a 2-coloring $C$ obtained by applying the binary coloring procedure, an edge $\{p, p'\} \in \overline{E}$ is said to be *$\alpha$-consistent* if $C(p') = \begin{cases} C(p) & \text{if } d(p,p') < \alpha \\ 1 - C(p) & \text{if } d(p,p') \geq \alpha \end{cases}$; otherwise, it is said to be *$\alpha$-inconsistent*. We say that $\overline{N}$ is *$\alpha$-consistent* if every edge of $\overline{E}$ is *$\alpha$-consistent*; otherwise, we say that $\overline{N}$ is *$\alpha$-inconsistent*.

Let $\overline{H} = (\overline{P}, \overline{T})$ be a spanning tree of $\overline{G}$, let us assume that every pixel of $\overline{P}$ has been colored following the binary coloring procedure by using the edges of $\overline{T}$, and let $n_\alpha(\overline{H})$ be the number of $\alpha$-inconsistent edges of $\overline{E}$. Notice that every edge in $\overline{T}$ will be $\alpha$-consistent, and the $\alpha$-consistency of $\overline{N}$ will not depend on the considered spanning tree $\overline{H}$.

**Example 2.1 (cont.).** *Let* $\alpha = 3.1$*, let* $\overline{H} = (P_I, \overline{T})$*, where* $\overline{T}$ *is the set of solid edges depicted in Fig. 2 (notice that* $\overline{H}$ *is a spanning tree of* $G(I)$*), and let* $\overline{C}$ *be the 2-coloring obtained when applying the binary coloring procedure by using the edges of* $\overline{T}$*, starting from the pixel* $(1,1)$ *(the values of* $\{\overline{C}(p)\}_{p \in P_I}$ *are shown in Fig. 2). Then, the edge* $\{(2,1),(2,2)\}$ *is $\alpha$-consistent, since* $d((2,1),(2,2)) = 1.6 < 3.1$ *and* $C((2,1)) = 0 = C((2,2))$*. The $\alpha$-inconsistent edges of* $E_I$ *are the dashed edges depicted in Fig. 2 (notice that* $n_\alpha(\overline{H}) = 5$*).*

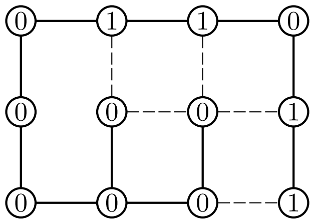


Fig. 2. Spanning tree $\overline{H}$, 2-coloring $\overline{C}$ and $\alpha$-inconsistent edges of $E_I$

In order to propose a suitable value for the threshold $\alpha$, two extreme cases should be taken into consideration: $\alpha > \max\{d(e) \mid e \in \overline{E}\}$ and $\alpha \leq \min\{d(e) \mid e \in \overline{E}\}$. For the first case all of the pixels of $\overline{P}$ are colored with the same color, and for the second case all of the adjacent pixels of $\overline{P}$ are colored alternatively with "0" and "1".

The *consistency level* of $\overline{N}$ is defined as the maximum value $\alpha \in \{d(e) \mid e \in \overline{E}\}$ for which $\overline{N}$ is $\alpha$-consistent.

The crude algorithm proposed in Gómez *et al.*[1] starts by computing the consistency level $\alpha^*$ of $N(I)$ and determining a spanning tree of $G(I)$. The first binary coloring takes $\alpha = \alpha^*$ and, then, the process is repeated separately on each connected component of the subgraphs of $G(I)$ generated by the pixels colored as "0" and "1", and so on. Nevertheless, this algorithm can be inefficient when handling medium and big size images, since, in general, the consistency levels of the considered pixel networks will attain their minimum possible values. Therefore, an appropriate decreasing scheme of $\alpha$, allowing an acceptable ratio of $\alpha$-inconsistent edges, should be applied; for this purpose, a deeper analysis of the inconsistencies is presented in the next section.

## 3. Analysis of the inconsistencies

Let $\alpha$ be a given threshold. If $\overline{N}$ is $\alpha$-inconsistent, then the number of $\alpha$-inconsistent edges of $\overline{E}$ will depend on the choice of the spanning tree $\overline{H} = (\overline{P}, \overline{T})$ of $\overline{G}$.

**Example 2.1 (cont.).** *Let $\hat{H} = (P_I, \hat{T})$, where $\hat{T}$ is the set of solid edges depicted in Fig. 3 (notice that $\hat{H}$ is a spanning tree of $G(I)$), and let $\hat{C}$ be the 2-coloring obtained when applying the binary coloring procedure by using the edges of $\hat{T}$, starting from the pixel* $(1, 1)$ *(the values of $\{\hat{C}(p)\}_{p \in P_I}$ are shown in Fig. 3). The $\alpha$-inconsistent edges of $E_I$ are the dashed edges depicted in Fig. 3 (notice that $n_\alpha(\hat{H}) = 3 < 5 = n_\alpha(\overline{H})$).*

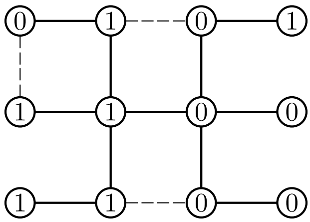


Fig. 3. Spanning tree $\hat{H}$, 2-coloring $\hat{C}$ and $\alpha$-inconsistent edges of $E_I$

Hence, we are interested in finding a spanning tree $\overline{H} = (\overline{P}, \overline{T})$ of $\overline{G}$ such that $n_\alpha(\overline{H})$ is as small as possible. For providing a method for reducing the number of $\alpha$-inconsistent edges of $\overline{E}$ (see Proposition 3.1), we require the following well-known concepts and results of Graph Theory, see Gondran and Minoux[2] for further details:

Let $G = (V, E)$ be a nondirected graph.

Given a subset of nodes $A \subseteq V$, the *incidence* of $A$ is the set $\omega(A) = \{\{v, v'\} \in E \mid v \in A,\ v' \in V \setminus A\}$, i.e, the subset of edges of $E$ that have one endpoint in $A$ and the other one in $V \setminus A$ (notice that $\omega(A) = \omega(V \setminus A)$).

A subset of edges $\theta \subseteq E$ is said to be a *cocycle* of $G$ if $\exists\, A \subseteq V$ such that $\omega(A) = \theta$.

Let $H = (V, T)$ be a spanning tree of $G$, and let $e \in T$. Then, it can be proved that there exists a unique cocycle $\theta^e$ of $G$ such that $\theta^e \cap T = \{e\}$; furthermore, $\theta^e = \omega(A)$, where $A$ is the subset of $V$ that generates one of the two connected components of the graph $(V, T \setminus \{e\})$.

**Proposition 3.1.** *Let $\overline{H} = (\overline{P}, \overline{T})$ be a spanning tree of $\overline{G}$, let $e \in \overline{T}$, let $\theta^e$ be the unique cocycle of $\overline{G}$ such that $\theta^e \cap \overline{T} = \{e\}$, and let $m_\alpha(\theta^e)$ be the number of $\alpha$-inconsistent edges of $\theta^e$. If $|\theta^e| < 2\, m_\alpha(\theta^e)$, then $n_\alpha(\overline{H}') < n_\alpha(\overline{H})$, where $\overline{H}' = (\overline{P}, \overline{T}')$, $\overline{T}' = (\overline{T} \setminus \{e\}) \cup \{e'\}$ and $e'$ is an $\alpha$-inconsistent edge in $\theta^e$.*

**Proof.** Since $e' \in \theta^e$, we have that $\overline{H}'$ is a spanning tree of $\overline{G}$.

Let $\overline{C}$ and $\overline{C}'$ be the 2-coloring obtained when applying the binary coloring procedure by using the edges of $\overline{T}$ and $\overline{T}'$, respectively, starting from a pixel $p_0 \in \overline{P}$, and let $A$ be the subset of $\overline{P}$ that generates the connected component of the graph $(\overline{P}, \overline{T} \setminus \{e\})$ that contains $p_0$. Then $\overline{C}'(p) = \overline{C}(p) \quad \forall p \in A$, since $p_0 \in A$. Now, taking into account that the edge $e'$ is $\alpha$-inconsistent for $\overline{C}$, but it must be $\alpha$-consistent for $\overline{C}'$, it follows that $\overline{C}'(p) = 1 - \overline{C}(p) \quad \forall p \in \overline{P} \setminus A$. Thus, an edge $\overline{e} \in \overline{E} \setminus \theta^e$ will be $\alpha$-inconsistent for $\overline{C}'$ if and only if it is $\alpha$-inconsistent for $\overline{C}$, and an edge $\overline{e} \in \theta^e$ will be $\alpha$-inconsistent for $\overline{C}'$ if and only if it is $\alpha$-consistent for $\overline{C}$. Therefore, we get that $n_\alpha(\overline{H}') = n_\alpha(\overline{H}) - m_\alpha(\theta^e) + |\theta^e| - m_\alpha(\theta^e) < n_\alpha(\overline{H})$. $\square$

**Remark 3.1.** Notice that $|\theta^e| < 2m_\alpha(\theta^e)$ if and only if $|\theta^e| - m_\alpha(\theta^e) < m_\alpha(\theta^e)$, i.e., the number of $\alpha$-consistent edges of $\theta^e$ is smaller than the number of $\alpha$-inconsistent edges of $\theta^e$. Notice also that the value of $n_\alpha(\overline{H}')$ does not depend on the considered $\alpha$-inconsistent edge $e'$.

**Example 2.1 (cont.).** *Let* $e = \{(3,2),(3,3)\}$ *(notice that* $e \in \overline{T}$*). Then* $\theta^e = \{\{(1,3),(2,3)\},\{(2,2),(2,3)\},\{(2,3),(2,4)\},\{(3,2),(3,3)\},\{(3,3),(3,4)\}\}$ *and* $m_\alpha(\theta^e) = 4$ *(see Fig. 2), hence* $|\theta^e| < 2\,m_\alpha(\theta^e)$*. Let* $e' = \{(2,2),(2,3)\}$ *and* $\overline{H}' = (P_I, \overline{T}')$*, where* $\overline{T}' = (\overline{T} \setminus \{e\}) \cup \{e'\}$ *(*$\overline{T}'$ *is the set of solid edges depicted in Fig. 4). By the proof of Proposition 3.1, it will be* $n_\alpha(\overline{H}') = n_\alpha(\overline{H}) + |\theta^e| - 2\,m_\alpha(\theta^e) = 5 + 5 - 8 = 2$*. Indeed, let* $\overline{C}'$ *be the 2-coloring obtained when applying the binary coloring procedure by using the edges of* $\overline{T}'$*, starting from the pixel* $(1,1)$ *(the values of* $\{\overline{C}'(p)\}_{p\in P_I}$ *are shown in Fig. 4); the* $\alpha$*-inconsistent edges of* $E_I$ *are the dashed edges depicted in Fig. 4.*

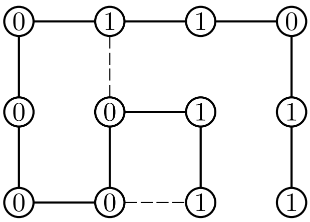


Fig. 4. Spanning tree $\overline{H}'$, 2-coloring $\overline{C}'$ and $\alpha$-inconsistent edges of $E_I$

Proposition 3.1 will be applied successively to $\overline{G}$ until obtaining a spanning tree $\overline{H} = (\overline{P}, \overline{T})$ of $\overline{G}$ such that $|\theta^e| \geq 2\,m_\alpha(\theta^e) \quad \forall e \in \overline{T}$. Then, a possible way for handling the remaining $\alpha$-inconsistent edges, if any, is to *isolate* the pixels that cause these $\alpha$-inconsistencies. Anyhow, the analysis introduced in this work could be a crucial issue to improve the crude algorithm for segmentation of digital images introduced by Gómez *et al.*[1] (see also Gómez *et al.*[3]).


## Acknowledgments

This research has been partially supported by the Government of Spain, under grant TIN2006-06190.


## References


1. D. Gómez, J. Montero, J. Yáñez and C. Poidomani, A graph coloring approach for image segmentation, *Omega* **35**, 173–183 (2007).
2. M. Gondran and M. Minoux, *Graphs and Algorithms* (John Wiley & Sons, Chichester, 1984).
3. D. Gómez, J. Montero and J. Yáñez, A coloring fuzzy graph approach for image classification, *Information Sciences* **176**, 3645–3657 (2006).