# A four-stage system for blind colour image segmentation

Ezequiel López-Rubio*, José Muñoz-Pérez and José Antonio Gómez-Ruiz
*E.T.S.I. Informática, University of Málaga, Campus de Teatinos, s/n. 29071, Málaga, Spain*
*Tel.: +34 95 213 71 55; Fax: +34 95 213 13 97; E-mail: {ezeqlr, munozp, janto}@lcc.uma.es*

**Abstract**. This paper proposes a new method to split colour images into regions. The only input information is the image to be segmented. Hence, this is a blind colour image segmentation method. It consists of four subsystems: preprocessing, cluster detection, cluster fusion and postprocessing.

Proofs are given for the significant properties that we have found. It is not necessary to specify the number of regions in advance, which is a significant improvement over the standard competitive-style strategies. Finally, simulation results are given to demonstrate the performance of this method for some images.

## 1. Introduction

A fundamental task in computer vision is that of segmenting an image into meaningful regions. It must be ensured that the resulting partition of the image is formed by homogeneous and connected parts. Image segmentation is a specific case of clustering, i.e., there is a need to group a number of samples into clusters. Hence, many general clustering algorithms may be applied to this problem, with some adaptations. There is a wide range of clustering methods, which may be classified in the following groups (see [21]):

a) $K$-means algorithm and competitive neural networks. Their drawback is the need to specify the number of clusters in advance.

b) Hierarchical clustering. The clusters are joined together in several steps. This forms a tree of clusters.

c) Parametric density estimation. This method tries to adjust the data to a mixture of g Gaussians. It has problems if the data is not distributed that way.

d) Nonparametric density estimation. It is not assumed that the distribution of the data is a mixture of Gaussians.

These algoritms are used in a variety of tasks, and not only in image segmentation. See [12] for an introduction to cluster analysis theory.

The procedure we present segments images without any extra information, i.e., it is a blind segmentation procedure. Hence, this is a low-level procedure, where a perfect segmentation can not be expected without further knowledge [22,23]. Anyway, blind segmentation algorithms are often used as the first step in the early stages of a computer vision system, and there are many different strategies to tackle this problem [10,11,15,16, 20,25,26].

Our image segmentation system is based on hierarchical clustering. This approach has been studied recently [13,19]. There are four stages: preprocessing, cluster detection, cluster fusion and postprocessing. The first step prepares the image by removing the noise and detecting the edges. Then we extract a large number of small clusters. The third step joins these clusters into larger parts of a suitable partition of the image. And finally, we remove any small spurious regions.

This paper is organized as follows. We describe the four subsystems of our segmentation system in Sections 2, 3, 4 and 5. We present some relevant properties of our method in Section 6, with the corresponding formal proofs. Computational experiments are considered in Section 7. Finally, Section 8 is devoted to conclusions.

[*] Corresponding author.

ISSN 1069-2509/03/$8.00 © 2003 – IOS Press. All rights reserved

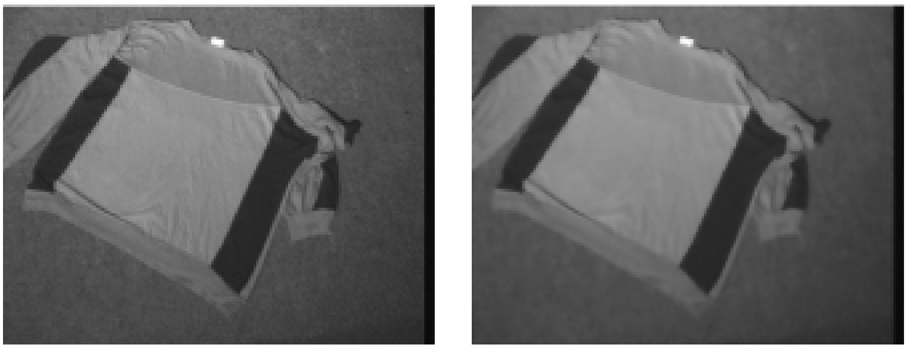

Fig. 1. Original image (left) and Wiener-filtered (right).

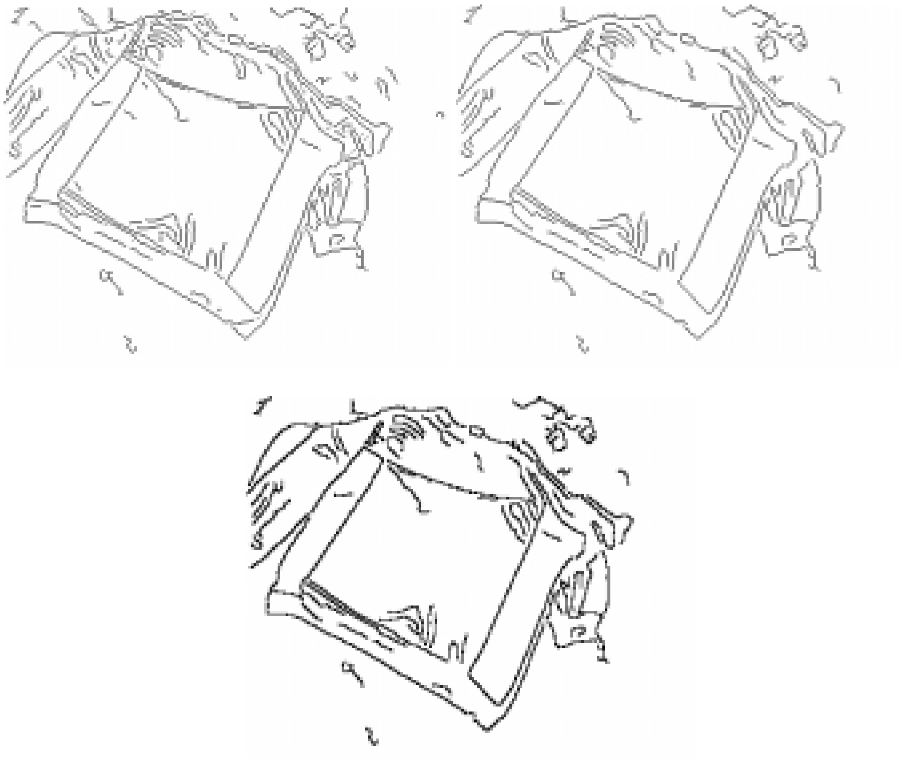

Fig. 2. Binary image with edges detected (left), with the small edges removed (right), and with the diagonals filled (below).

## 2. Preprocessing

In this phase we perform a number of tasks required to prepare the image. This is needed by the following steps of the algorithm.

We start by removing the noise with a suitable filter. In our experiments we have used an adaptative Wiener filter for each of the three RGB colour channels. We have estimated the noise power in $5 \times 5$ windows (Fig. 1). If the images are not very noisy, this filtering could be skipped. Anyway, the result of the edge and group detection which follow is better if we filter the image.

On the other hand, the group detection and fusion stages need information about which pixels belong to the edges of the image. This information is supplied by preparing a binary image where the zeros mean edges (see Fig. 2).

The edge detection has been performed with the Canny method [7], which has been applied to the gray level version of the Wiener filtered image. It is a multistage process. First a lowpass filter is used, and then

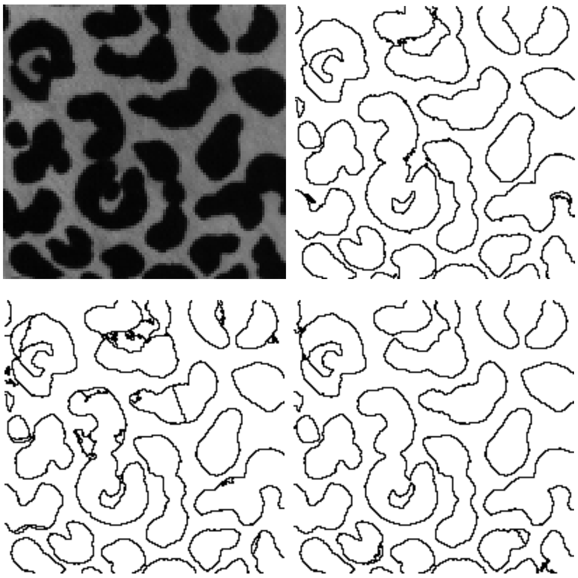

Fig. 3. Snow leopard image and resulting segmentations.

the derivative is estimated (the higher values are candidates to be edge pixels). Finally, the pixels with high values of the derivative in the middle of an edge are selected as edge pixels.

After the edges are detected, it is convenient to remove those edges with a small number of pixels, since it is almost sure that they have been produced by noise. We detect all the 8-connect regions formed by edge pixels, and then we count the number of pixels of them. Those regions with fewer pixels are removed. In the experiments, the regions with less than $M$ pixels have been removed, where $M$ is a fixed parameter.

Finally, we fill the edge diagonals so that the edges disconnect the different parts of the background. This is achieved by a morphological operator.

## 3. Cluster detection

The purpose of this subsystem is to obtain a set of clusters $A$. Let $S$ be the set of pixels of the image. Then $A$ is a partition of $S$. As previously noted, if we use the $k$-means algorithm or competitive neural networks the problem is that we can not know the number of clusters in advance. This means, in competitive neural networks, that it is impossible to say the number of units (neurons) which best fits a particular image. See [1,14, 24,27] for a review of the competitive learning, and [4] for a theoretical study of the method.

We solve this problem by using an oversized number of units. This ensures that the clusters obtained are very homogeneous. Every pixel $\mathbf{x} \in S$ is given by two image co-ordinates and three colour co-ordinates. If we select the RGB colour space we have $\mathbf{x} = (x, y, r, g, b)$. Other colour spaces like CIE L*u*v* could be considered. We consider here only colour similarity, but this can be easily extended to texture features. We modify the weights $\mathbf{w}_i$ by the stardard competitive algorithm:

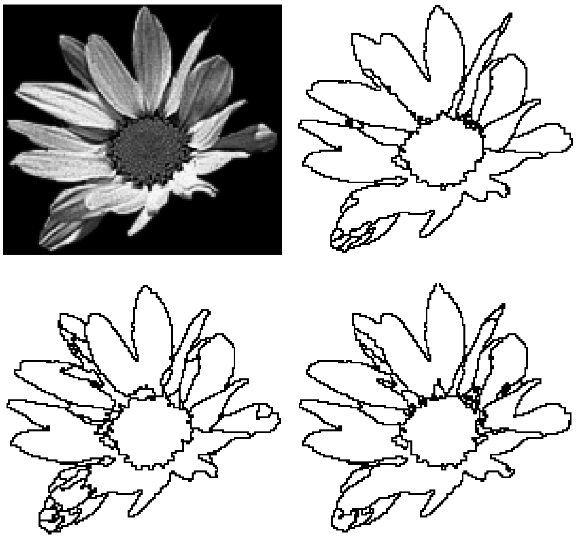

Fig. 4. Daisy image and resulting segmentations.

$$\mathbf{w}_i(k+1) = \begin{cases} \alpha\mathbf{x} + (1-\alpha)\mathbf{w}_i(k), \text{ if } i \text{ wins} \\ \mathbf{w}_i(k), \text{ otherwise} \end{cases} \quad (1)$$

where $i \in \{1, \ldots, N\}$, and $N$ is the number of units. $N$ is chosen ***a priori*** so that the number of regions of the image is much greater. The winning neuron is the one that has maximum *activation potential* $h_i$, where

$$h_i = \sum_{j=1}^{5} w_{ij} x_j - \sum_{j=1}^{5} \frac{w_{ij}^2}{2} \quad (2)$$

Note that

$$h_i > h_j \Leftrightarrow ||\mathbf{x} - \mathbf{w}_i|| < ||\mathbf{x} - \mathbf{w}_j|| \quad (3)$$

The set of pixels $S_i$ associated with a particular unit $i$ is defined by

$$S_i = \left\{ \mathbf{x} \in S | i = \arg\min_j ||\mathbf{x} - \mathbf{w}_j|| \right\} \quad (4)$$

Unfortunately, it is not guaranteed that all $S_i$ are connected sets of pixels. We will consider 8-connectivity here, but this is not essential. A recent paper where either four or eight connected components of images are considered for segmentation can be found in [17]. We need to divide every $S_i$ into maximal 8-connected sets of pixels. This can be achieved by a *blob colouring* strategy, i.e., by marking as “equivalent” all pairs of neighbour pixels that belong to the same $S_i$. It is considered that the edge pixels can not belong to any set. This is aimed to make the edge pixels separate sets of connected pixels.

Then we compute the reflexive, symmetric and transitive closure of this relation. This closure is an equivalence relation, and the corresponding partition is made up of parts that are maximal 8-connected sets. Let $L_j$ be the parts obtained, $L_j \subseteq S \forall j$.

## 4. Cluster fusion

The 8-connected sets $L_j$ will typically be too small, because $N$ is large. Then we need a fusion subsystem, whose task is to merge those sets. We want the resulting regions to be the definitive partition of the image, i.e., one that is acceptable to a human. In other words, our goal is to join similar and adjacent sets of pixels.

We propose to compute the adjacency graph of the clusters $L_j$, in the 8-neighbour sense. Again, the edge pixels can not belong to any cluster, so they separate clusters. Then we sort the pairs of adjacent clusters by their degree of similarity. So, we need to define an easily computable measure of homogeneity. First, we calculate the centroid $c_j$ of cluster $j$ as:

$$\mathbf{c}_j = \frac{1}{|L_j|} \sum_{k \in L_j} \mathbf{z}_k \tag{5}$$

where $\mathbf{z}_k = (r_k, g_k, b_k)$ or $\mathbf{z}_k = (L*_k, u*_k, v*_k)$ depending on the colour space considered. Note that L*u*v* space was designed to closely resemble the human perception of colour. Other spaces can be less adequate, like RGB. Then we take the Euclidean distance $||\mathbf{c}_i - \mathbf{c}_j||$ as a measure of disparity between clusters $i$ and $j$. Next we sort the list of pairs of clusters by their disparity. Finally, we reverse the list obtained to get the final result.

After that, we build several partitions $P_i$ of the image, $i \in \{1, \ldots, M\}$. All of them are the result of joining some clusters together. First we get the ith relation $R_i$ by taking the $i/M$ more homogeneous pairs of adjacent clusters. This can be implemented by having a list of adjacent pairs of clusters sorted by the mentioned criterion. Then the ith partition $P_i$ is obtained by computing the reflexive, symmetric and transitive closure of $R_i$. We denote the parts of the ith partition as $Q_{ij}$. Note that every part of $P_i$ is included into a part of $P_{i+1}, \forall i \in \{1, \ldots, M-1\}$.

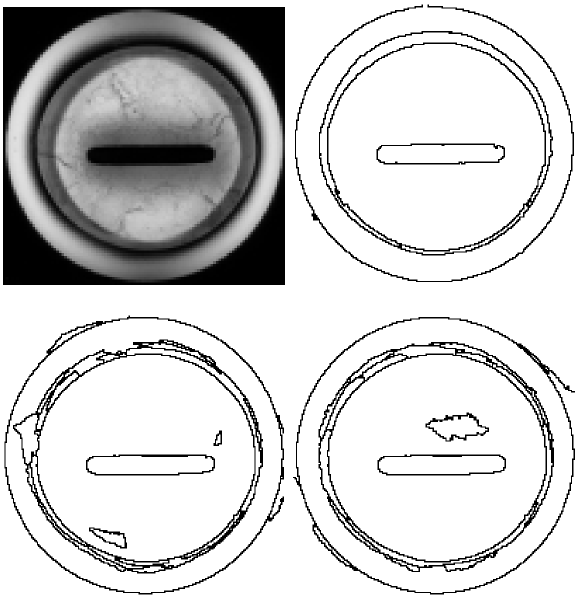
Fig. 5. Alien eye and resulting segmentations.

The final step of the algorithm is to decide which of the partitions $P_i$ is the best. We have selected three possible options:

a) The pixels that lie in the same part should be similar. On the other hand, the pixels that belong to different parts should be different. This approach comes from cluster analysis (see [21]), and is known as the *intra-inter variation ratio*. We need to define the *disparity* measure between a pair of pixels. We take the same option as in the previous method: the Euclidean distance in the colour space. Then we choose some sample pairs of pixels at random, and compute the *intra-part variation* as

$$\delta_{\text{intra}}(P_i) = \frac{1}{N} \sum_{k,h \in Q_{ij}} ||\mathbf{z}_k - \mathbf{z}_h|| \tag{6}$$

where $N$ is the number of samples collected, and the pairs $(k, h)$ are restricted to belong to the same part. Note that there is no need to examine more than a small fraction of the total number of pairs of pixels. The *inter-part variation* is computed as

$$\delta_{\text{inter}}(P_i) = \frac{1}{M} \sum_{\substack{k \in Q_{ij} \\ h \in Q_{im} \\ j \neq m}} ||\mathbf{z}_k - \mathbf{z}_h|| \tag{7}$$

where $M$ is the number of samples collected, and we consider only the pairs of pixels that belong to different parts. The *intra-inter variation ratio* is defined as

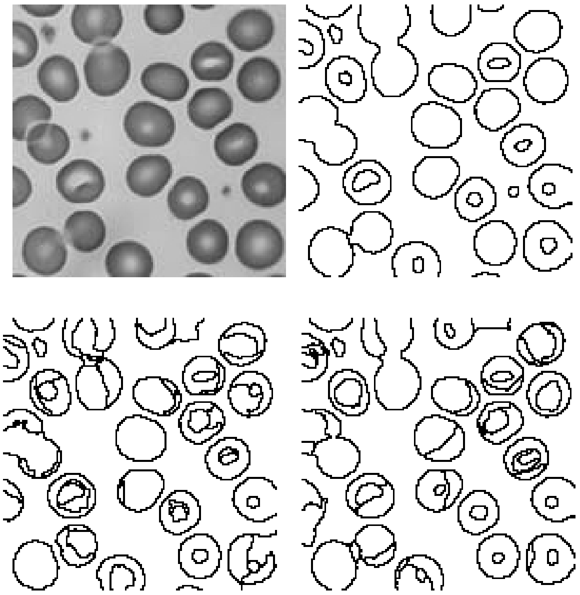

Fig. 6. Red cells image and resulting segmentations.

$$p = \frac{\delta_{\text{inter}}}{\delta_{\text{intra}}} \tag{8}$$

Note that if we have very homogeneous parts, $\delta_{\text{intra}}$ will be low. Conversely, if the parts are very different, $\delta_{\text{inter}}$ will be high. So, $p$ has a large value when the partition is adequate. Then, we select the partition which has the largest value of $p$.

b) We have found in experiments that $\delta_{\text{inter}}$ varies very slowly in the partitions that our procedure generates. This suggests to study only $\delta_{\text{intra}}$. So, there is no need to compute $\delta_{\text{inter}}$, which is a important speedup. Furthermore, there is a typical value of $\delta_{\text{intra}}$ that indicates an optimal partition, which depends on the colour space considered. We note this value $\delta_0$. Then we say that the best partition is that which minimises abs($\delta_{\text{intra}} - \delta_0$), where $abs$ is the absolute value of a real number.

c) Most of the functions proposed in literature to evaluate an image segmentation depend on parameter or threshold values, which must be adjusted by human experience (see [28] for a survey on segmentation evaluation functions). These kind of functions are not suitable for incorporation in our system, which is fully automated. One of the few exceptions is the $F(I)$ function from Liu and Yang [18]. However, Borsotti et al. [5] point out several drawbacks of this function. In particular, $F(I)$ tends to evaluate segmentations with too many small regions favorably. They propose two functions $F'(I)$ and $Q(I)$ as improved versions of $F(I)$. The three functions are defined as:

$$F(I) = \frac{\sqrt{R}}{1000NM}\left(\sum_{i=1}^{R}\frac{e_i^2}{\sqrt{A_i}}\right) \tag{9}$$

$$F'(I) = \frac{1}{10000NM}\left(\sum_{i=1}^{R}\frac{e_i^2}{\sqrt{A_i}}\right) \sqrt{\sum_{A=1}^{\text{Max}}[R(A)]^{1+1/A}} \tag{10}$$

$$Q(I) = \frac{\sqrt{R}}{10000NM}\sum_{i=1}^{R}\left[\frac{e_i^2}{1+\log A_i} + \left(\frac{R(A_i)}{A_i}\right)^2\right] \tag{11}$$

where $N \times M$ is the image size, $R$ is the number of regions, $A_i$ is the area of region $i$, $e_i$ is the sum of Euclidean distances between the RGB color vectors of the pixels of region $i$ and the mean color of region $i$, $R(A)$ is the number of regions having exactly area $A$, and $Max$ is the area of the largest region in the image. The smaller the values of the functions, the better the segmentation should be. The functions $F'(I)$ and $Q(I)$ retain the advantages of $F(I)$ while removing its drawbacks, so any of them can be used to select the optimal partition in our system.

## 5. Postprocessing

This last stage has two tasks: the inclusion of the edge pixels in the best parts, and the removal of too small parts. The overall goal is to achieve a final segmentation with no small regions nor isolated pixels. Hence we look for a enhancement of the segmentation obtained with the optimal partition computed by the previous stage.

The inclusion of the edge pixels is performed by taking each of them and merging them with the 8-adjacent part with the nearest mean colour. This is aimed to get the best possible definition of the region borders.

The removal of small regions is an iterative procedure. In each iteration, the first of all is to count the number of pixels of the parts. Then we label all the parts that do not reach the specified minimum number of pixels, $Z$. For each labelled part, if there are non-labelled parts (i.e., large parts) which are 8-adjacent to it, then it is merged with the most similar part, in

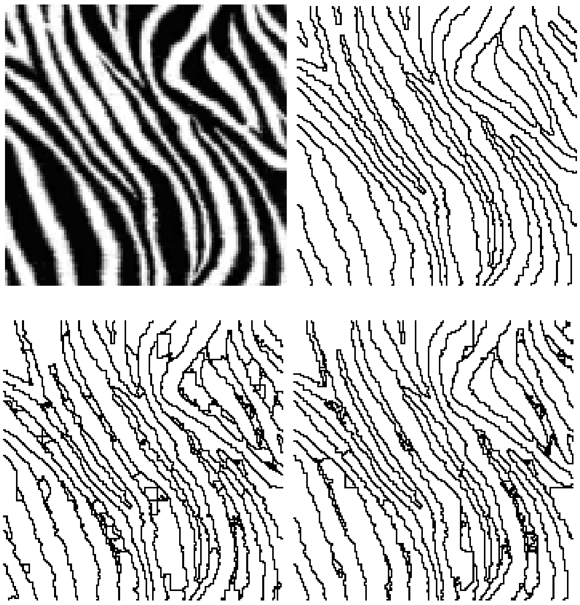

Fig. 7. Zebra image and resulting segmentations.

the mean colour sense. Otherwise, it is merged with the labelled 8-adjacent part with the most similar mean colour. This algorithms executes a fixed number of iterations, $K$.

As we prove in Theorem 4, this algorithm has the property that after the $K$ iterations, the minimum pixel number in a part is $\min(2^K, Z)$. Consequently, if we fix $Z$, it is easy to compute the number of iterations needed to guarantee that there are no parts with less than $Z$ pixels: $K = \text{ceil}(\log_2 Z)$, where $ceil$ rounds towards $+\infty$. It can be observed that the number of iterations needed is always small, since it depends on a logarithm.

## 6. Properties

The proposed method has a number of interesting properties, which are related to the cluster fusion and the postprocessing. The following propositions and theorems prove them.

**Theorem 1.** *It holds that $P_i \subseteq P_{i+1}, \forall i \in \{1, \ldots, M-1.\}$*

**Proof.** *By definition of $R_i, R_i \subseteq R_{i+1}$. Then $r(R_i) \subseteq r(R_{i+1})$, where $r$ stands for the reflexive closure function. So we have that $s(r(R_i)) \subseteq s(r(R_{i+1}))$ and $P_i = t(s(r(R_i))) \subseteq t(s(r(R_{i+1}))) = P_{i+1}$, where $s$ and $t$ stand for the symmetric and transitive closure functions, respectively.* □

**Corollary 1.** *Every part of $P_{i+1}$ can be obtained by merging one or more parts of $P_i$.*

**Corollary 2.** *As $i$ increases, the number of parts decreases towards 1.*

**Proposition 1.** *Every part of $P_i$ has a spanning tree composed by some of the $i/M$ more homogeneous pairs of adjacent clusters.*

**Proof.** *If two clusters a and b are in the same part of $P_i$, there exists an undirected chain in $R_i$ that joins them. By definition of Ri, that chain is composed by pairs which fulfill the specified condition. Finally, by algebra we know that $R_i \subseteq t(s(r(R_i))) = P_i$.* □

**Proposition 2.** *Every part of $P_i$ is 8-connect, i.e., given two clusters which belong to the same part, there exists a 8-connect path which joins them composed by clusters of that part.*

**Proof.** *Again, if two clusters $a$ and $b$ are in the same part of $P_i$, there exists an undirected chain in $R_i$ that joins them. But $R_i$ is composed exclusively by 8-adjacent pairs of clusters.* □

**Theorem 2.** *Let $X' = t(s(r(X)))$ be a partition, where $R_i \subseteq X \subseteq R_{i+1}$. Then $P_i \subseteq X' \subseteq P_{i+1}$.*

**Proof.** *By a reasoning similar to that of Theorem 1, $t(s(r(R_i))) \subseteq t(s(r(X)))$ and $t(s(r(X))) \subseteq t(s(r(R_{i+1})))$.* □

**Theorem 3.** *Let $i \in \{1, \ldots, M\}$. Let $Z$ be a partition whose parts are 8-connect, such that $P_i \subset Z$. Then there exists $(x, y) \in Z$ such that the homogeneity measure of $(x, y)$ is less than the minimal homogeneity on $R_i$.*

**Proof.** *If $P_i \subset Z$, then there exists $(a, b) \in Z | (a, b) \notin P_i$. Let $A$ and $B$ be the parts to which $a$ and $b$ belong, respectively. We know that $A \neq B$, because $P_i$ is a partition. As $Z$ is also a partition, it holds that $A \cup B$ is contained into a single part of $Z$. Since $Z$ is 8-connect, there is a 8-connect chain which joins $a$ and $b$ in $Z$. But then there is a pair in this chain whose homogeneity is less than the minimal homogeneity on $R_i$, because if this was false, then all the chain would be in $R_i$. Then*

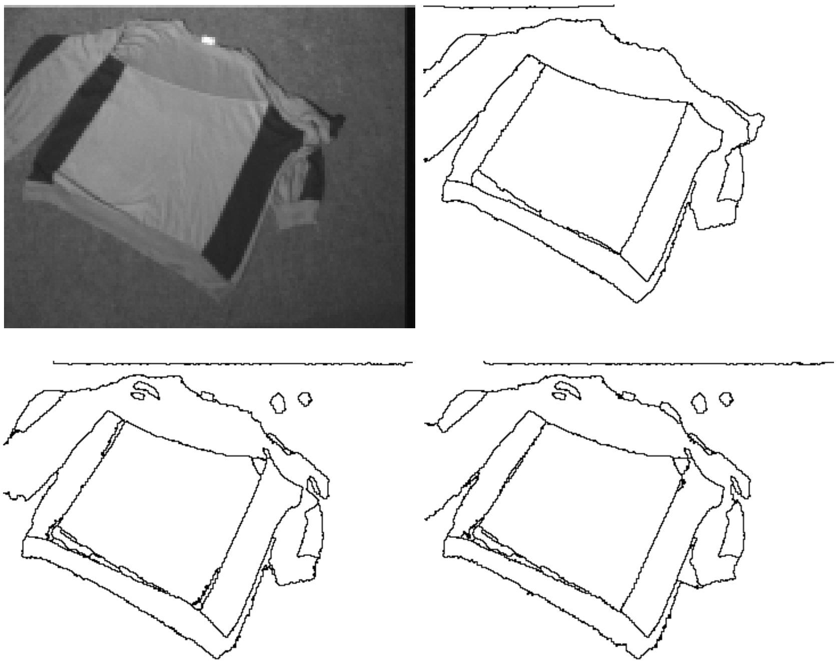

Fig. 8. Segmentations of the Fig. 1 original image.

*$a$ and $b$ would be joined in $R_i$, which leads to $A = B$.* □

**Theorem 4.** *After the removal of small regions (post-processing stage), every part has at least $\min(2^K, Z)$ pixels, where $K$ is the number of iterations and $Z > 0$ the minimum number of pixels requested.*

**Proof.** *By induction on $K$,*

- *Base case, $K = 0$. In this case there is no removal of small regions, so there may be one-pixel parts, $1 = 2^0$.*
- *Inductive hypothesis. We suppose that after $K$ iterations all the parts have at least $\min(2^K, Z)$ pixels.*
- *Inductive step. In the $K+1$-th iteration, we start with parts with at least $\min(2^K, Z)$ pixels, according to the inductive hypothesis. We have two cases:*
  - a) *If $\min(2^K, Z) = Z$, then $\min(2^{K+1}, Z) = Z$, which means that in this iteration there are no labelled regions and the conclusion of the theorem holds.*
  - b) *Otherwise, $\min(2^K, Z) = 2^K$. That is, we start with parts of at least $2^K$ pixels. Every part with less than $Z$ pixels merges with another. In the worst case we would have a part of exactly $2^K$ pixeles which merges with other part of $2^K$ pixels, to form a new part with $2 \cdot 2^K = 2^{K+1}$ pixels. As $2^{K+1} \geqslant \min(2^K, Z)$, the conclusion of the theorem holds.* □

## 7. Experimental results

There is no standard quantitative measure to evaluate the performance of an image segmentation system. Hence, we have tested our system in two different ways. First, we have evaluated from a qualitative point of view, i.e., we have given subjective comments on the obtained segmentations. Then we have used two known evaluation functions to compare our system quantitatively with other proposals.

### 7.1. Qualitative evaluation of the method

We have selected some images to show our procedure. A neural network with 50 units has been chosen in the cluster detection stage. We have used the L*u*v* colour space in this stage. We have considered the intra-part variation approach (with $\delta_0 = 38$), the $F'(I)$ approach and the $Q(I)$ approach in the cluster fusion

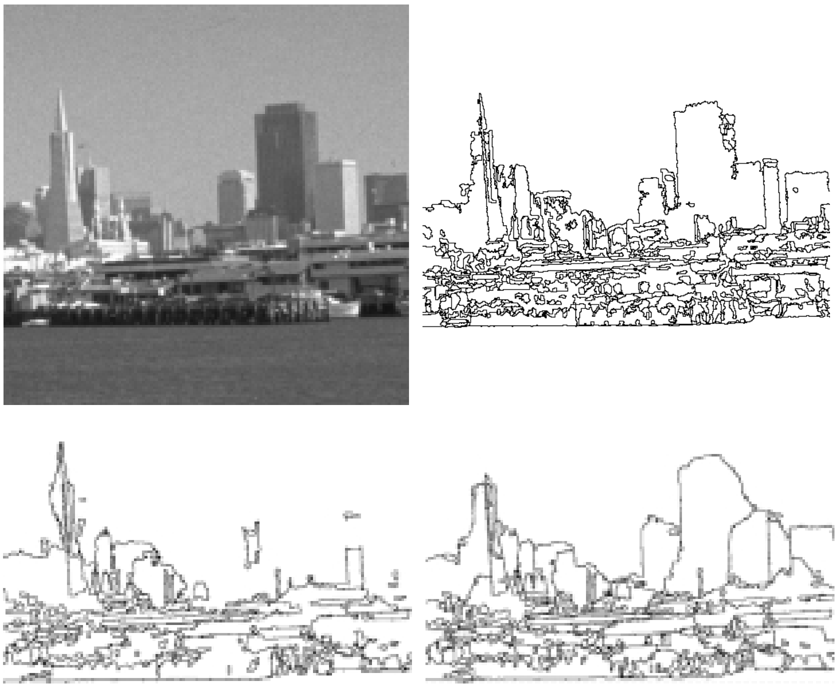

Fig. 9. San Francisco city view and segmentations.

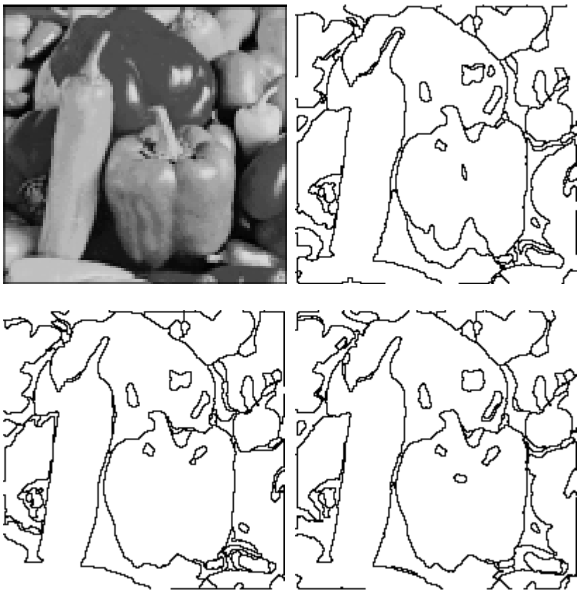

Fig. 10. The ‘peppers’ benchmark image and resulting segmentations.

stage. The parameter values have been $M = 15$ for the preprocessing, and $Z = 20$ for the postprocessing.

The computing time is not excessive: 10 seconds for a $350 \times 300$ image on a PC with a 500 MHz processor. This is due to the fact that the image size has a small effect in the complexity, since the pixels are processed only in the two first stages of the system. The third stage, which is the most time-consuming, only processes regions of pixels. The last stage processes some edge pixels, but there are a small amount of them, since the proportion of edge pixels in an image is usually low. So, the key factor is the number of regions, and not of pixels.

We show in Figs 3 to 9 some original images with their corresponding segmentations. From left to right and from up to down we have: the original image, the resulting segmentation with the intra-part variation approach, the resulting segmentation with the $F'(I)$ approach, and the resulting segmentation with the $Q(I)$ approch. The region boundaries have been drawn in black. Next we comment the results shown in these figures.

We can see on Fig. 3 an example with non-convex, curve-shaped regions. Note that the dark regions are recognised as distinct 8-connected regions. Also, the background is identified as a unique 8-connected region. Only the $F'(I)$ approach has some spurious small regions.

The background of Fig. 4 has been recognised as a region, and so has been the center of the flower. The system is able to separate most of the petals, except where there is no clear separation. Again, the $F'(I)$ approach has some spurious small regions.

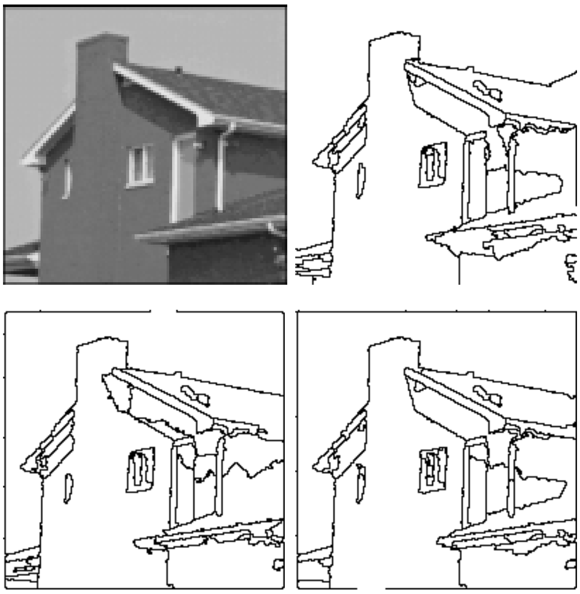

Fig. 11. The ‘house’ benchmark image and resulting segmentations.

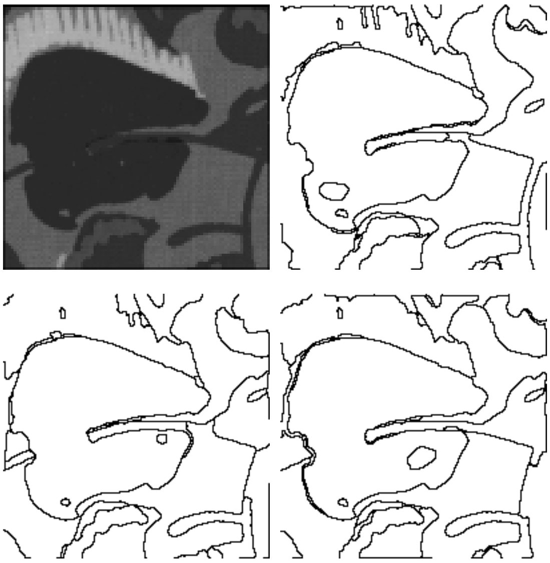

Fig. 12. The ‘strawberry’ benchmark image and resulting segmentations.

The segmentations of Figs 5 and 6 are very good for the intra-part variation approach, while the other two have some wrongly split regions.

The results of Fig. 7 are adequate for all the approaches. However, the intra-part variation function gives clearer boundaries between regions. The original image of Fig. 1 is segmented in Fig. 8, and the main foreground regions are correctly detected, as well as the background. There are only three small spurious regions when using $F'(I)$ and $Q(I)$. The three approaches show a small, elongated region at the top, which reflects a sudden change in the color of the original figure.

Finally, Fig. 9 has a large number of small regions of different colours. Nevertheless, the sky and the sea are recognised as two big regions. As in the previous figures, the intra-part variation result is the best, because the skyline is better defined.

We may conclude that the three proposed approaches yield good results in all the test images considered. However, the intra-part variation function is slightly better in some situations.

### *7.2. Quantitative evaluation of the method*

We have designed a set of experiments to make a quantitative comparison of our system with known approaches to blind image segmentation. All the parameter values and other choices of our segmentation system have been the same as in the qualitative evaluation subsection.

We have selected three benchmark images [5,6] to test our procedure. The original images and the obtained results are shown in Figs 10 to 12. The ordering of our three approaches in these figures is the same as in the previous subsection. The values of the evaluation functions $F'(I)$ and $Q(I)$ for the segmentations of the figures are shown in Tables 1 and 2. Please note that these values have been computed after the post-processing stage. This is the reason because some tabulated values of $F'(I)$ for the $Q(I)$ approach may be better than those of $F'(I)$ approach, and vice versa. It can be seen on these tables that intra-part variation and $Q(I)$ have approximately equal performance, and both of them yield better results than $F'(I)$.

Borsotti et al. have evaluated with the functions $F'(I)$ and $Q(I)$ the results of four known image segmentation systems [5]. The three images considered here were used by them as benchmarks. The segmentation systems are the following: the sart network by Baraldi et al. [3] enhanced with a Hopfield network based technique by Campadelli et al. [6], the competitive learning scheme by Uchiyama and Arbib [2], the histogram analysis by Carlotto [8] and the art2 system by Carpenter and Grossberg [9]. Tables 3 and 4 list the relative ranks of our proposals when compared with the results of Borsotti et al. using $F'(I)$ and $Q(I)$, respectively. Note that the lower the number, the better our approaches, i.e., a ‘1’ means that our proposal is better

Table 1
Values of $F'(I)$ for the segmentations of Figs 10 to 12

| Image | Intra-part variation approach | $F'(I)$ approach | $Q(I)$ approach |
|---|---|---|---|
| peppers | 67.97 | 87.43 | 77.39 |
| house | 24.46 | 9.24 | 8.53 |
| strawberry | 13.33 | 15.27 | 13.63 |

Table 2
Values of $Q(I)$ for the segmentations of Figs 10 to 12

| Image | Intra-part variation approach | $F'(I)$ approach | $Q(I)$ approach |
|---|---|---|---|
| peppers | 466.76 | 655.05 | 573.79 |
| house | 215.04 | 57.84 | 55.37 |
| strawberry | 91.20 | 100.68 | 82.51 |

Table 3
Relative ranks using $F'(I)$ of the segmentations of Figs 10 to 12, with respect to the segmentations considered by Borsotti et al

| Image | Intra-part variation approach | $F'(I)$ approach | $Q(I)$ approach |
|---|---|---|---|
| peppers | 1 | 1 | 1 |
| house | 1 | 1 | 1 |
| strawberry | 1 | 2 | 1 |

Table 4
Relative ranks using $Q(I)$ of the segmentations of Figs 10 to 12, with respect to the segmentations considered by Borsotti et al

| Image | Intra-part variation approach | $F'(I)$ approach | $Q(I)$ approach |
|---|---|---|---|
| peppers | 2 | 3 | 2 |
| house | 4 | 1 | 1 |
| strawberry | 2 | 2 | 2 |

than all the results reported by Borsotti et al., while a '5' would mean that it is worse than all of them. It can be seen that our methods have a good position in $Q(I)$ rankings, and they are the best in all $F'(I)$ rankings but one. The $F'(I)$ function evaluates our methods more favorably because our postprocessing stage removes all the small regions by incorporating them in larger ones. This means that all the terms in the summation under the square root in (10) have a small value. Furthermore, there is little impact in the overall colour error, which is measured by the other summation in (10), because the merged regions are small. The $Q(I)$ function is less sensitive to small regions, and this is the reason because it ranks our procedures in lower positions than $F'(I)$ does.

## 8. Conclusions

We have proposed a new system for blind colour image segmentation. It has four subsystems: preprocessing, cluster detection, cluster fusion and postprocessing. The preprocessing uses a noise removing filter, an edge detection procedure and morphological operations. A competitive neural network detects the clusters, followed by and algorithm which obtains 8-connected clusters. Then the clusters are merged to form regions, with the help of an evaluation function. Several suitable evaluation functions have been discussed. Finally, we remove the isolated pixels and small spurious regions which may remain, in order to get a better segmentation. We have given formal proofs of the most relevant properties of the method. Experimental results have been included for different real images, with qualitative and quantitative evaluations. The obtained results show that our method yields good segmentations with a small computational load, because it processes whole regions, and not pixels, most of the time.

## References

[1] S.C. Ahalt, A.K. Krishnamurphy, P. Chen and D.E. Melton, Competitive Learning Algorithms for Vector Quantization, *Neural Networks* **3** (1990), 277–290.

[2] M.A. Arbib and T. Uchiyama, Color image segmentation using competitive learning, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **16**(12) (1994), 1197–1206.

[3] A. Baraldi and F. Parmiggiani, A neural network for unsupervised categorization of multivalued input patterns: an application to satellite image clustering, *IEEE Trans. on Geoscience and Remote Sensing* **33**(2) (1995), 305–316.

[4] J.S. Baras and A. La Vigna, Convergence of Kohonen's learning vector quantization, *International Joint Conference on Neural Networks* **3** (1990), 17–20. San Diego, CA.

[5] M. Borsotti, P. Campadelli and R. Schettini, Quantitative evaluation of color image segmentation results, *Pattern Recognition Letters* **19** (1998), 741–747.

[6] P. Campadelli, D. Medici and R. Schettini, Color image segmentation using Hopfield networks, *Image and Vision Computing* **15**(3) (1997), 161–166.

[7] J. Canny, A Computational Approach to Edge Detection, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **8**(6) (1986), 679–698.

[8] M.J. Carlotto, Histogram analysis using a scale-space approach, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **9** (1987), 121–129.

[9] G.A. Carpenter and S. Grossberg, ART2: self-organization of stable category recognition codes for analog input patterns, *Applied Optics* **26** (1987), 4919–4930.

[10] K. Chen, D.L. Wang and X.W. Liu, Weight Adaptation and Oscillatory Correlation for Image Segmentation, *IEEE Trans. on Neural Networks* **11**(5) (2000), 1106–1123.

[11] Y. Deng and B.S. Manjunath, Unsupervised Segmentation of Color-Texture Regions in Images and Video, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **23**(8) (2001), 800–810.

[12] B.S. Everitt, S. Landau and M. Leese, Cluster Analysis, 4th ed. (2001). Oxford University Press.

[13] H. Frigui and R. Krishnapuram, Clustering by competitive agglomeration, *Pattern Recognition* **30**(7) (1997), 1109–1119.

[14] A. Gersho, On the Structure of Vector Quantizers, *IEEE Transactions on Information Theory* **28** (1982), 157–166.

[15] R.H. Haralick and L.G. Shapiro, Image segmentation techniques, *Computer Vision Graphics Image Processing* **29** (1985), 100–132.

[16] K. Haris, S.N. Efstratiadis, N. Maglaveras and A.K. Katsaggelos, Hybrid Image Segmentation Using Watersheds and Fast Region Merging, *IEEE Trans. on Image Processing* **7** (1998), 1684–1699.

[17] R. Jones, Connected Filtering and Segmentation Using Component Trees, *Computer Vision and Image Understanding* **75**(3) (1999), 215–228.

[18] J. Liu and Y.-H. Yang, Multiresolution color image segmentation, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **16**(7) (1994), 689–700.

[19] T.P. Minka and R.W. Picard, Interactive Learning with a society of models, *Pattern Recognition* **30**(4) (1997), 565–581.

[20] N.R. Pal and S.K. Pal, A review on image segmentation techniques, *Pattern Recognition* **26**(9) (1993), 1294–1294.

[21] E.J. Pauwels and G. Frederix, Finding Salient Regions in Images, *Computer Vision and Image Understanding* **75**(1/2) (1999), 73–85.

[22] T. Pavlidis and Y.-T. Liow, Integrating region growing and edge detection, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **12**(3) (1990), 225–233.

[23] J. Ton, J. Stricklen and A.K. Lain, Knowledge-based segmentation of landsat images, *IEEE Trans. on Geoscience and Remote Sensing* **29**(2) (1991), 222–232.

[24] N. Ueda and R. Nakano, A New Competitive Learning Approach Based on an Equidistortion Principle for Designing Optimal Vector Quantizers, *Neural Networks* **7**(8) (1994), 1211–1227.

[25] D.L. Wang and D. Terman, Image Segmentation Based on Oscillatory Correlation, *Neural Computation* **9** (1997), 805–836.

[26] J.Z. Wang, J. Li, R.M. Gray, G. Wiederhold, Unsupervised Multiresolution Segmentation for Images with Low Depth of Field, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **23**(1) (2001), 85–91.

[27] E. Yair, K. Zeger and A. Gersho, Competitive Learning and Soft Competition for Vector Quantizer Design, *IEEE Trans. Signal Processing* **40**(2) (1992), 294–308.

[28] Y.J. Zhang, A survey of evaluation methods for image segmentation, *Pattern Recognition* **29**(8) (1996), 1335–1346.