C

Articles d'illustration, partie III

C.1 [Fiorio and Nock, 1998]

Image Segmentation Using a Generic, Fast and Non-Parametric Approach

Christophe Fiorio, Richard Nock
LIRMM, 161 rue Ada, F-34790 Montpellier Cedex 5

Abstract:

In this paper, we investigate image segmentation by region merging. Given any similarity measure between regions, satisfying some weak constraints, we give a general predicate for answering if two regions are to be merged or not during the segmentation process. Our predicate is generic and has six properties. The first one is its independance with respect to the similarity measure, that leads to a user-independant and adaptative predicate. Second, it is non-parametric, and does not rely on any assumption concerning the image. Third, due to its weak constraints, knowledge may be included in the predicate to fit better to the user's behaviour. Fourth, provided the similarity is well-chosen by the user, we are able to upperbound one type of error made during the image segmentation. Fifth, it does not rely on a particular segmentation algorithm and can be used with almost all region-merging algorithms in various application domains. Sixth, it is calculated quickly, and can lead with appropriated algorithms to very efficient segmentation.

Keywords: Vision and Image Processing, AI algorithms, Machine Learning.

1 Introduction

In the early stage of a vision process it is necessary to isolate "objects" in the scene before recognizing them. This step in the analyzing process is called image segmentation. There are several approaches to this problem [1, 2], namely threshold techniques ([3] for example), boundary (such as Canny-Deriche edge detector [4, 5]) and regions based methods ([6] and [7] for more recent results), and hybrid techniques (as for example [8, 9]) combining the two last methods. In this paper we are concerned by region segmentation of intensity (*i.e.* grey levels) images and more precisely region merging. The goal of region segmentation is to divide the image I in regions $R_1, \ldots, R_n$ such that $I = \bigcup_{i=1}^{n} R_i$, $R_i \bigcap R_j = \emptyset$ if $i \neq j$, each region being connected and satisfying an homogeneity criterion. Region merging builds up regions by combining smaller regions, pixels being taken as primary regions, using a predicate $\mathcal{P}$ as a similarity test.

The quality of a segmentation is strongly dependent on the way one answer to a simple question: "should (or should not) be merged two regions ?" Albeit a user can answer to this question, it is hard to give further steps in the definition of a segmentation algorithm. The first reason comes from the difficulty to quantify a mathematical notion of similarity between regions, which is a part of the user's judgment, but rather implicit and semantic dependent in his case. Second, the use of this similarity for answering to the latter question is not straightforward, since it implies defining a mapping from the similarity values to the Boolean values $\{yes, no\}$. Third, the main bottleneck is obviously the design of the whole segmentation algorithm which uses such predicate.

Many segmentation algorithms use statistical or mathematical techniques relying on hypotheses made *a priori* on the image [10, 11, 12]. In that case, the time spent for the calculations can be very important. Furthermore, many of these hypotheses are parametric, introducing distributional assumptions on the image. *A contrario*, Other approaches focuse on algorithmic properties of the image [13, 14], leading to efficient segmentation algorithms. Our approach differs from all these ones in that it is based on the combination of statistical and algorithmic properties of a criterion used for the segmentation. Statistical properties are based on the absence of any parametric hypotheses on the image. Algorithmic properties are based on a quick calculation (constant-time) of the criterion, that can lead to optimal image segmentation (both linear-time and linear-space).

In this paper we give a generic non parametric predicate for image segmentation based on "distance" between regions. It is generic in the sense that it is independent from the chosen distance and independent of a particular region segmentation algorithm. Furthermore, provided the user can give a calculable "distance" relating his distance, a part of the error made during the segmentation process can be controlled. Such errors can arise *e.g.* when merging two regions that represent in fact distinct objects. Moreover, it satisfies the desirable property of being calculable in constant time for a broad class of "distances". So used with an adequate algorithm we can achieve a linear-time and linear-space segmentation process. At last as shown at the end of Section 3.2 our predicate is auto-adaptative since, except for a risk parameter, the user doesn't tune others parameters and the limit value of the similarity is set automatically according to the risk and the image.

In the next section we provide basic definitions, as well as the algorithm `mergeSquare` we use in our application. Then, we present in the following section our generic predicate. In the same section, we discuss about the fast calculation of the predicate criteria. In the last section, we present some similarity measures we used, and some experimental results we obtained using them. We also present an application where some knowledge is put in our criterion, and the quality of the segmentation obtained. We leave in the appendix some theoretical remarks on the error reduction process that occurs during segmentation, using our criterion.

2 Preliminaries

2.1 Definitions

An image I with r rows and c columns is an $r \times c$ matrix taking its entries in $\{1, ..., g\}$ where g is an integer representing the number of grey levels of I. A *region* of I is a non-empty, 4-connected[1] subset of 2^I. Two distinct regions of I are *adjacent* if and only if (iff for short) their union is still a region. A *segmentation predicate* is a predicate $\mathcal{P}(R, R')$ taking as parameters two adjacent regions R and R' and answering "Y" or "N" whether or not R and R' should be merged during the segmentation process. The segmentation predicate uses a *criterion* $\mathrm{C}(R, R')$ measuring the "proximity" of regions R and R'. Formally, C is only required to satisfy the following conditions:

- C is an application from $2^I \times 2^I$ to $\mathbb{R}$, having finite values,
- C is symmetric : $\forall R, R' \in 2^I, \mathrm{C}(R, R') = \mathrm{C}(R', R)$.

Intuitively, the smaller $\mathrm{C}(R, R')$, the more likely R and R' are to be merged according to $\mathcal{P}$. Since it depends on C, we can rewrite $\mathcal{P}$ as $\mathcal{P}_{\mathrm{C}}$.

A basic example of criterion $\mathrm{C}(R, R')$ is the mean difference $\mathrm{MD}(R, R')$:

$$\left| \frac{1}{|R|} \sum_{(x,y)\in R} I(x,y) - \frac{1}{|R'|} \sum_{(x,y)\in R'} I(x,y) \right|$$

and a segmentation predicate using MD would be

$$\mathcal{P}_{\mathrm{MD}}(R, R') = \text{"}\mathrm{MD}(R, R') < 20\text{"}$$

[1]Two pixels are 4-adjacent iff they share an edge. A region is 4-connected iff given any two pixels p and q of it, there exists a path of pixels in the region, $p_1 = p, p_2, ..., p_k = q$, such that $\forall i \in \{1, ..., k-1\}$, p_i is 4-adjacent to p_{i+1}.

$\mathcal{P}_{\mathrm{MD}}(R, R')$ is read as "if $\mathrm{MD}(R, R') < 20$, then R and R' can be merged into one region". The design of $\mathcal{P}_{\mathrm{C}}$ is not straightforward. In particular, it depends on the choice of $\mathcal{C}$. In our example, the constant "20" we give for $\mathcal{P}_{\mathrm{MD}}$ is chosen arbitrarily ; it is *a priori* difficult to know if this constant is better than another one. Note also that the design of $\mathcal{P}$ is not straightforward if we are allowed to choose arbitrary C.

2.2 The algorithm `mergeSquare`

In order to achieve linear-time and linear-space region segmentation algorithm we will use for illustration purpose the `mergeSquare` algorithm (see next page) presented in [14]. This algorithm is based on a union-find structure which represents regions by trees of pixels belonging to the same region (see [15] for more information on union-find algorithms and [16, 14] for application to image processing). It proceeds by a quad-tree-like scan of the pixels of the image, and for each pixel on the boundary of a square the algorithm checks if the region it belongs to can be merge with the region to which belongs the pixel on the other side of the boundary. The merge (the *union*) of two regions is done by linking the root of the tree of one region to the other one. Note that to identify the region a pixel belongs to, it is sufficient to check the root of its tree, the *find* operation allowing us to find quickly the roots of trees. In Algorithm 1, the image is assumed to be a square matrix, and the four sub-squares of a given matrix are called $S_{NW}, S_{NE}, S_{SW}, S_{SE}$.

To be linear this algorithm requires to use *constant-time*

Algorithm 1: `mergeSquare(`S,k`)`

```
Input: an integer k and a square S
if k = 0 then return ;
h⁻ = 2^(k-1) ; h⁺ = 2^(k-1);
for D=NW to SE do mergeSquare(S_D, k-1);
for i = 0 to 2^k do
    left = Find(S[i, h⁻]);
    right = Find(S[i, h⁺]);
    if P(left,right) then Union(left,right);
for i = 0 to 2^k do
    up = Find(S[h⁻, i]);
    down = Find(S[h⁺, i]);
    if P(up,down) then Union(up,down);
```

predicates, i.e. predicates that are calculated in constant-time at each stage of the process. Note that this implies that parameters required for the calculus (such as sums of grey levels and cardinal of a region for the mean criterion for example) must be updated in constant time after a union is done.

3 The generic predicate

3.1 Segmentation criterion and associated predicate

The idea underlying our predicate criterion is the following : two adjacent regions R and R' can be merged during the segmentation process iff

- $\mathrm{C}(R, R')$ is sufficiently small, and
- $\mathrm{C}(R, R')$ is not greater than many $\mathrm{C}(R'', R''')$, where $R'' \neq R'''$ and R'', R''' cover the regions of I.

We create a "meta-criterion" $F(\mathrm{C}(R, R'))$, using a function $F(x)$ whose domain is $\mathbb{R}$, taking its values in $[0, 1]$. It takes place between the classic criterion C and the predicate $\mathcal{P}_{\mathrm{C}}$. Given any user-defined criterion C, F is defined as

$$F(x) = \frac{\left|\left\{R'', R''' \in I : \left[\begin{array}{l} R'' \neq R''' \\ \mathrm{C}(R'', R''') \leq x \end{array}\right.\right\}\right|}{|\{R'', R''' \in I : R'' \neq R'''\}|} \qquad (1)$$

$F(x)$ is the ratio of couple of regions for which the value $\mathrm{C}(R'', R''')$ is smaller than that of $\mathrm{C}(R, R')$. Since we do not make assumptions on the image, we do not suppose that regions R'' and R''' are linked. The predicate criterion $\mathcal{P}_{\mathrm{C}}$ is now the following:

$$\mathcal{P}_{\mathrm{C}}(R, R') \equiv \text{"}F(\mathrm{C}(R, R')) < \alpha_1\text{"} \qquad (2)$$

where α_1 is a risk associated to $\mathcal{P}_{\mathrm{C}}$ (generally, $3\% \leq \alpha_1 \leq 10\%$). α_1 corresponds to the (small) percent of regions that can be merged.

3.2 Representation of $F(x)$

We suppose that the image I is segmented, and contains a set S of regions (at the beginning of the algorithm, each pixel defines one element of S). It is convenient for our purpose to present $F(x)$ as the integral of an estimated density $f(x)$. We build the function $f(x)$ in the following way. Let $\{v_0, v_1, ..., v_k\}$ stands for the set of possible values of $\mathrm{C}(R'', R''')$, $R'' \neq R'''$ and $R'', R''' \in S$. Then $\forall i \in \{1, 2, ..., k\}, \forall x \in]v_{i-1}, v_i]$,

$$f(x) = \frac{\left|\left\{R'', R''' \in I : \left[\begin{array}{l} R'' \neq R''' \\ \mathrm{C}(R'', R''') = v_i \end{array}\right.\right\}\right|}{(v_i - v_{i-1}) \times |\{R'', R''' \in I : R'' \neq R'''\}|}$$

and $\forall x \in [v_0 - 1, v_0]$,

$$f(x) = \frac{\left|\left\{R'', R''' \in I : \left[\begin{array}{l} R'' \neq R''' \\ \mathrm{C}(R'', R''') = v_0 \end{array}\right.\right\}\right|}{|\{R'', R''' \in I : R'' \neq R'''\}|}$$

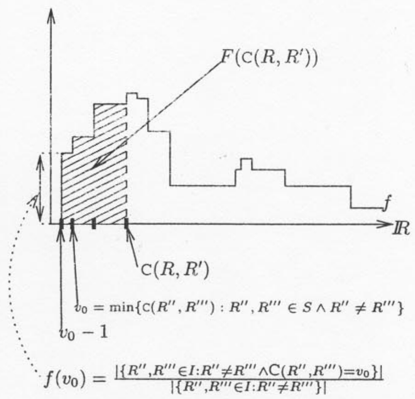

Figure 1: The construction of f and a representation of $F(\mathrm{C}(R, R'))$.

We also set

$$\forall x \in] - \infty, v_0 - 1[\cup]v_k, +\infty[, f(x) = 0$$

Figure 1 shows the construction of f as well as a representation of $F(\mathrm{C}(R, R'))$ that we justify now. $\forall i \in \{0, 1, ..., k\}$, $F(v_i)$ can be calculated using the following formula: $F(v_i) = \int_{-\infty}^{v_i} f(z)dz$. Indeed, by setting $v_{-1} = v_0 - 1$, we obtain

$$\begin{aligned} \forall 0 \leq i \leq k, \int_{-\infty}^{v_i} f(z)dz &= \sum_{j=0}^{i} (v_j - v_{j-1}) f(v_j) \\ &= \sum_{j=0}^{i} \frac{\left|\left\{R'', R''' \in I : \left[\begin{array}{l} R'' \neq R''' \\ \mathrm{C}(R'', R''') = v_j \end{array}\right.\right\}\right|}{|\{R'', R''' \in I : R'' \neq R'''\}|} \\ &= \frac{\left|\left\{R'', R''' \in I : \left[\begin{array}{l} R'' \neq R''' \\ \mathrm{C}(R'', R''') \leq v_i \end{array}\right.\right\}\right|}{|\{R'', R''' \in I : R'' \neq R'''\}|} = F(v_i) \end{aligned}$$

The integral formula for the calculation $F(x)$ shows the monotonocity of F, because f takes no negative value. Given two adjacent regions R, R' of S, note that $F(\mathrm{C}(R, R'))$ is just the area delimited between f and three lines whose equations are $y = 0$, $x = v_0 - 1$ and $x = \mathrm{C}(R, R')$.

The signification of $\mathcal{P}_{\mathrm{C}}$ can be illustrated by Figure 2 (it also shows a function f^* whose role is explained in in the appendix. It it represented here for convenience). The decision to merge (or not to merge) two regions is taken by comparing $F(\mathrm{C}(R, R'))$ and α_1, the two hatched areas in Figure 2. Equivalently, since $F(x)$ is monotonous, the decision consists in comparing $\mathrm{C}(R, R')$ and x_{α_1}, where x_{α_1} satisfies $F(x_{\alpha_1}) = \alpha_1$. In the example of Figure 2, since $F(\mathrm{C}(R, R')) < \alpha_1$ (equivalently, $\mathrm{C}(R, R') < x_{\alpha_1}$), $\mathcal{P}_{\mathrm{C}}(R, R')$ returns "Y" and regions R and R' are merged.

Note that the limit value x_{α_1} is not set by the user. Its calculation depends on the risk α_1, as well as on the image.

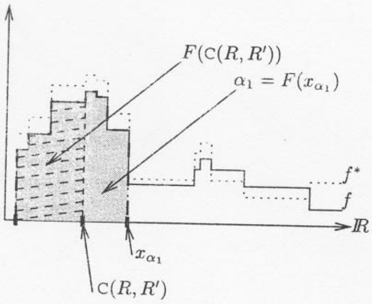

Figure 2: According to $\mathcal{P}_{\mathrm{C}}(R,R')$, regions R and R' can be merged.

Even when setting α_1 to a constant value, different images shall lead to different value of x_{α_1}. In this sense we can say that our predicate is *auto-adaptative*. This last point is the most important. We put in appendix some remarks concerning the limitation of the error during the segmentation process, that are not needed for our application purposes. Of greater importance is the way to calculate rapidly $F(x)$.

3.3 Rapid calculation of F

We define a predicate $\mathcal{P}_{\mathrm{C}}$ (resp. a criterion C) to be a *constant-time predicate* (resp. *constant-time criterion*) iff its calculation from two regions R and R' can be done in constant time. This constant-time property is very important, as C and $\mathcal{P}_{\mathrm{C}}$ are accessed each time the merging of two regions is tested. Reducing as small as possible the complexity of their calculation is a proof of rapidity for the segmentation algorithm.

Obtaining a constant-time predicate $\mathcal{P}_{\mathrm{C}}$ implies necessarily to have

1. a constant-time criterion C, and
2. a constant-time calculation of $F(\mathrm{C}(R,R'))$.

We propose in the following section criteria C that are constant-time, satisfying the first condition. Note however that the calculation of $F(\mathrm{C}(R,R'))$ requires more than constant time : in the worst-case, it is linear in the size of the image. In our calculations, we replace $F(\mathrm{C}(R,R'))$ by an estimator $\hat{F}(\mathrm{C}(R,R'))$ having the constant-time property we look for. It is calculated on the basis of a set P_t of t (constant) random observations of pairs of regions, over the regions of I. $\hat{F}(\mathrm{C}(R,R'))$ equals

$$\frac{\left|\left\{\{R'',R'''\} \in P_t : \left[\begin{array}{l} R'' \neq R''' \\ \mathrm{C}(R'',R''') \leq \mathrm{C}(R,R') \end{array}\right]\right\}\right|}{t}$$

We have [17]:

$$P_r\left(\left|\hat{F}(\mathrm{C}(R,R')) - F(\mathrm{C}(R,R'))\right| < \sqrt{\frac{1}{2t}\log\frac{2}{\alpha}}\right) > 1-\alpha$$

Let $\alpha = \alpha_1$. Let $\alpha_2 = \sqrt{\frac{1}{2t}\log\frac{2}{\alpha_1}}$, where α_2 is a fixed risk parameter. Solving for t leads to

$$t = \frac{1}{2(\alpha_2)^2}\log\frac{2}{\alpha_1} \qquad (3)$$

which is a constant for any fixed values of α_1 and α_2.

An illustration of the consequence of constant-time calculation of $\mathcal{P}_{\mathrm{C}}$ is given by our choice of the algorithm `mergeSquare` to implement our predicate. Indeed, we have [14]:

Theorem 1 *If $\mathcal{P}_{\mathrm{C}}$ is a constant-time predicate,* `mergeSquare` *is linear-time and linear-space with respect to the size of I.*

4 Application

4.1 Specialization and quality improvement of the predicate criterion

The basis of the choice to merge two regions relies on the construction of a distribution f. This distribution is not necessarily built on the image that is currently being segmented:

- it can be another image, in the same domain, chosen because it has better quality (contrast, luminosity, etc...),
- it can be a little part of an image, chosen because it presents important details that should be treated carefully when seen in the segmented image,
- it can be an image taken from a collection of presegmented images, in relevant domains. For example, the segmentation of a boat can be done by constructing f from a presegmented boat image. Similarly, the medical segmentation of some organ can be done using a presegmented image of this organ, or similar images,
- f can be constructed using $i > 1$ images, putting $f = \frac{1}{i}\sum_{j=1}^{i} f_j$ where f_j is the distribution constructed from the j^{th} image.

These three points point out the inner possibility of putting knowledge, and machine learning in our predicate.

453

Given one image (or more) to build f, the predicate $\mathcal{P}_{\mathrm{C}}$ can also be specialized in many ways to integrate knowledge on the image being segmented, in order to fit as precisely as possible to the users behaviour. This is equivalent to constructing conditional densities for f. We can define three types of specializations. The first ones are syntaxic specializations. Among these are all comparisons involving regions characteristics, such as

- using only regions whose grey-levels are limited, or with more general functions involving grey-levels of images,
- considering regions whose shape is constrained, or belongs to some class,
- considering regions not violating some constraint, such as the absence of edges between them if the image has been preprocessed using an edge-detection filter.

More generally, syntaxic specializations can be of any type, provided we can give some precise predicate for their calculation. Figure 3 gives an example of such predicate that can be added to the calculation of $F(x)$ in equation 1.
The second type of specialization are predicates involving a semantic knowledge cf the image segmented, or (more likely) of benchmark images used to build f, as it is precised in the beginning of this section. For example, We may want to merge only regions that can be part of an organ, that would be described on a short benchmark of presegmented images. This makes the segmentation "object-oriented" since it tends to isolate large regions that are likely to belong to the searched object (in our example, an organ). Figure 3 gives another examples of semantic specialization for our predicate criterion.
Finally, the third type of possible specializations of our predicate, as shown in figure 3, relies on both syntaxic and semantic specializations.

Specializations of our predicate criterion drops downs to defining an *homogenity* predicate $\mathtt{H}$, used to constrain the distribution f to more interesting distributions. The calculation of $F(x)$ can therefore be modified, and in the general case, we can rewrite equation 1 by:

$$F(x) = \frac{\left|\left\{R'', R''' \in I : \begin{bmatrix} R'' \neq R''' \\ \mathtt{H}(R'', R''') = \mathtt{True} \\ \mathtt{C}(R'', R''') \leq x \end{bmatrix}\right\}\right|}{\left|\begin{array}{l}\{R'', R''' \in I : \\ R'' \neq R''' \wedge \mathtt{H}(R'', R''') = \mathtt{True}\}\end{array}\right|} \quad (4)$$

For example, in the case where we use an edge-detector to prevent merging regions separated by an edge, $\mathtt{H}(R'', R''')$ would be "there is no edge between R'' and R'''". In the case where we want to limit the grey-levels of merged regions, $\mathtt{H}(R'', R''')$ could be "the average grey-levels of R'' and R''' are not greater than x".

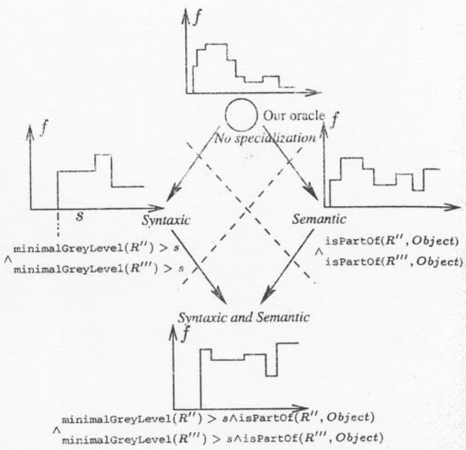

Figure 3: Possible specializations of our criterion.

4.2 Examples of constant-time criteria C

We now give some basic criteria $\mathtt{C}$ that were used to perform our experiments. They are all constant-time criteria. Their calculation is based on the following quantities defined for any region R, whose update is constant-time when merging two regions: the number of pixels of R ($|R|$), the grey level average of R ($\overline{R}$), the maximal grey level of R ($\max_R$) and the minimal grey level of R ($\min_R$). Their update when merging two regions R and R' in a single region $R \cup R'$ is performed in constant-time, since we have $|R \cup R'| = |R| + |R'|$ and $\overline{R \cup R'} = (|R|\overline{R} + |R'|\overline{R'})/(|R| + |R'|)$ and $\max_{R\cup R'} = \max\{\max_R, \max_{R'}\}$ and $\min_{R\cup R'} = \min\{\min_R, \min_{R'}\}$.

The first criterion is the mean difference $\mathtt{MD}$ that we can define as $\mathtt{MD}(R, R') = |\overline{R} - \overline{R'}|$. The second one is the range $\mathtt{RA}$ which traduces how much the histogram of grey levels in the union $R \cup R'$ flattens:

$$\begin{aligned} \mathtt{RA}(R, R') &= \max_{(x,y)\in R\cup R'} I(x,y) - \min_{(x,y)\in R\cup R'} I(x,y) \\ &= \max\{\max_R, \max_{R'}\} - \min\{\min_R, \min_{R'}\} \end{aligned}$$

The third one is the between-regions error $\mathtt{BR}$, which is used in discriminant analyses:

$$\mathtt{BR}(R, R') = |R|(\overline{R} - \overline{R \cup R'})^2 + |R'|(\overline{R'} - \overline{R \cup R'})^2$$

We have tested other parameters, including variations in standard error of means, variations in the sums of grey levels, but the results were never better than those provided by these three criteria.

4.3 Basic experiments

Due to the lack of space, Figures 4, 6 present only a few experiments that were carried out, on three original benchmark images presented in figure 5. Our whole segmentation algorithm includes repeating N times the algorithm `mergeSquare` on each image, in order to decrease the region number. Each time, N was chosen small, never exceeding 15. Parameters α_1 and α_2 were tested with different values, using each of the criteria MD, RA and BR. Each result is presented by a four-tuple $(N, \alpha_1, \alpha_2, \mathcal{C})$. Our experiments show the interest of considering different criteria, leading to segmentations that behave quite differently. This difference is to be seen not on different qualities, but on different varieties. Indeed, unless we try bad segmentation criteria, the segmentations had all similar quality, but highlighted different part of the images, this being important since various domain studies shall focuse on various aspects of images and objects. Furthermore, we carried experiments setting high values α_1 and α_2 (one is showed with both values set to .20), and the quality of the segmentation obtained for these high values accounts for the robustness of our criterion. Real-time calculation never exceeded five minutes for the biggest images, for a non-optimized C program ran on a Pentium Pro 180. Processing time for 256×256 images never exceeded 1 minute.

4.4 An example of specialization of our criterion

We present an application of a syntaxic specialization of our criterion on the image `lena`.

Figure 5 shows the original image. We have first isolated a small part of the image, known as being very difficult to segment : the top of the hat. Figure 6 shows this part. We then have produced a first segmentation using an error criterion, which we denote ER:

$$\mathrm{ER}(R, R') = \sum_{(x,y)\in R\cup R'} \left(I(x,y) - \overline{R\cup R'}\right)^2$$

It is straightforward to show that this criterion is also constant-time, provided we keep for each region the sum of squares of grey levels (which also updates in constant time). We did not use separately this criterion like the three others, since it gave poor results for segmentations like in the preceeding section (the number of regions is generally greater than for the three other parameters). We have executed our criterion building f not on `lena`, but on the small part of the hat. This produces an histogram which is

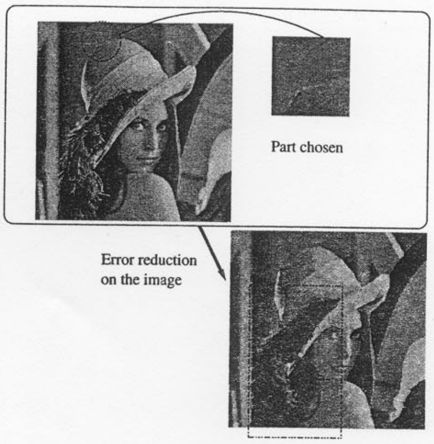

Figure 6: The error reduction in `lena`, with parameters $(20, .80, .10, \mathrm{ER})$.

highly concentrated around the difficult values, and therefore leads to keeping almost perfectly the hat in the final image. The errors due to lena's hair and her hat's feathers are greatly reduced (a rectangle shows the area of interest on figure 6).

This latter image obtained is then segmentated, but using to build f a very noisy image where greylevels are similar to `lena`. The final result is presented in figure 7 (left image). It contains very few regions (less than $0,06\%$ of the original number of regions), and many intuitive regions of the original image are kept almost intact. For comparison, the right image in figure 7 gives the result of a segmentation using the same criterion MD, chosen because it led to good decreasing of the regions number in lena's hair and hat's feathers. It shows that the segmentation using the preprocessed image of figure 6 leads effectively to reducing the error on regions such as lena's hair and hat's feathers.

5 Conclusion

We have presented in this paper a generic non parametric predicate to be used in region segmentation algorithms. It is generic since it for each distance criterion chosen we get a new predicate. This choice of a distance is critical since it determines the final result. The important thing to note is that, provided the distance criterion chosen fit exactly a user criterion segmentation, part of the predicate error with respect to the user is controlled. The stating that the similarity measure is well chosen by the user is strong, but due to the weak constraints on $\mathcal{C}$, many knowledge can be

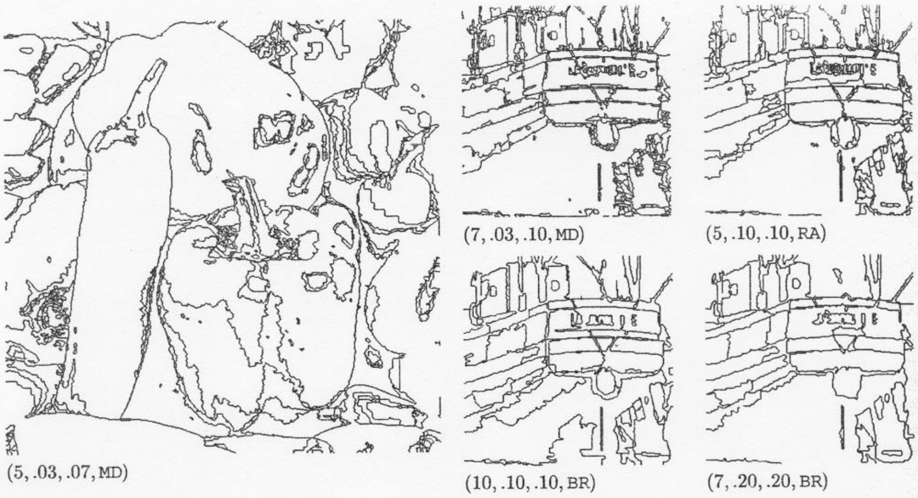

Figure 4: Segmentation produced with various parameters setting, see text.

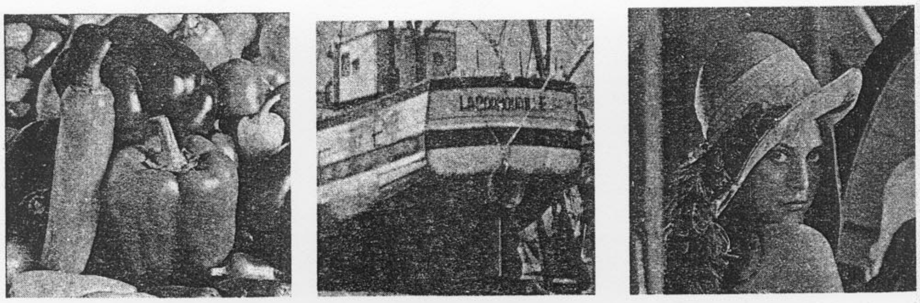

Figure 5: Original images (resp. peppers, cornouaille, lena).

Figure 7: Final segmentations with lena, with error reduction (left), or without (right).

included to improve its behaviour.

Moreover our predicate doesn't need to tune parameters in order to get good result. It is only necessary to set a risk parameter, this risk being generally fixed within 3% to 10%. We claim that it is auto-adaptative since, providing the risk paramater is fixed, a limit value on the distance function will be set automatically according to the distribution of the image. This limit value will determine the similarity notion between regions in the image. It will be different in another image.

At last, if the chosen distance function is a constant-time predicate, we ensure that our predicate will be a constant-time predicate This allows to design efficient segmentation algorithms. In Section 4 we have given some results showing good segmentations with simple distance criteria. This allow us to expect very good results in particular applications where knowledge can be included in the distance criterion.

References

[1] R. M. Haralick and L. G. Shapiro, "Survey: Image segmentation techniques," *Computer Vision, Graphics, and Image Processing.*, vol. 29, pp. 100–132, 1985.

[2] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, vol. 1 and 2. Addison-Wesley Publishing Company, Inc., June 1992.

[3] T. Quiger, P. Miché, and R. Debrie, "Segmentation by auto-adaptative thresholding," in *Proceedings of the 6th Int. Conf. on Image Analysis and Processing*, (Villa Olmo, Como, Italy), pp. 34–42, 4–6 September 1991. Progress in Image Analysis and Processing II.

[4] J. Canny, "A computational approach to edge detection," *IEEE trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[5] R. Deriche, "Separable recursive filtering for efficient multi-scale edge detection," in *Int. Workshop on Machine Vision & Machine Intelligence*, (Tokyo), December 1987.

[6] S. W. Zucker, "Survey: Region growing: Childhood and adolescence," *Computer Vision, Graphics, and Image Processing.*, vol. 5, pp. 382–399, 1976.

[7] Y.-L. Chang and X. Li, "Fast image region growing," *Image and Vision Computing*, vol. 13, pp. 559–571, September 1995.

[8] J. Benois and D. Barba, "Image segmentation by region-contour cooperation for image coding," in *Proc. 11^{t}h Int. Conf. on Pattern recognition (ICPR92)*, pp. 331–334, 1992.

[9] J.-P. Gambotto, "A new approach to combining region growing and edge detection," *Pattern Recognition Letters*, 1993.

[10] S. C. Zhu and A. Yuille, "Region competition : Unifying snakes, region growing, and bayes/MDL for multiband image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 884–900, 1996.

[11] S. Dugelay, C. Graffigne, and J.-M. Augustin, "Segmentation of multibeam acoustic imagery in the exploration of the deep-sea bottom," in *Int. Conf. on Pattern Recognition*, pp. 437–446, 1996.

[12] R. Watzel, K. Braun, A. Hess, W. Zuschratter, and H. Scheich, "Restoration of dendrites and spines with the objective of Topologically Correct Segmentation," in *Int. Conf. on Pattern Recognition*, pp. 472–476, 1996.

[13] W. G. Kropatsch and S. B. Yacoub, "A revision of pyramid segmentation," in *Int. Conf. on Pattern Recognition*, pp. 477–481, 1996.

[14] C. Fiorio and J. Gustedt, "Two linear time Union-Find strategies for image processing," *Theoretical Computer Science*, vol. 154, pp. 165–181, 1996.

[15] K. Mehlhorn, *Data Structures and Algorithms 1: Sorting and Searching*. Springer, 1984.

[16] M. B. Dillencourt, H. Samet, and M. Tamminen, "A general approach to connected-component labeling for arbitrary image representation," *J. of the Association for Computing Machinery*, vol. 39, pp. 253–280, April 1992.

[17] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *J. Am. Stat. Assoc.*, pp. 13–30, 1963.

6 Annex

In this part, we provide an interpretation of the segmentation and predicate criteria.
For large images, under the hypothesis that the criterion C is well chosen, The function f represents a distribution function which is a good approximation of an "ideal distribution function" f^*, that we do not know exactly. f^* satisfies the following property : the answer of $\mathcal{P}_C$ for merging two adjacent regions R and R' of I, calculated using f^* instead of f, would be *exactly* the same as the user's answer on the question "can we merge R and R'?"
This notion of exactitude is very important. It postulates that the predicate $\mathcal{P}_C$ behaves like the user when testing the merging of two regions, and therefore that C is well chosen by the user. It is obviously a very strong hypothesis difficult to ensure. But it has the property that it allows us to upperbound one type of error the algorithm can make when merging two regions. Furthermore, and we emphasize on this later in our conclusion, many parameters and knowledge can be included in the criteria C and $\mathcal{P}_C$, thus adaptatable to fit precisely to the user's behaviour.

If the previous hypothesis is not valid, there is still a partial control of the error made during the algorithm, but it is more basically a control of the error made by $\mathcal{P}_C$ with respect to C, and not with respect to the user's behaviour. From the user's point of view, it is less important, but it is relevant for the properties of our algorithm.

f^* can be seen as a similarity measure between two distinct regions of I, representing the distribution function of a random variable X. Thus, we have for example $F(x) \approx P_r(\mathrm{X} \leq x)$. Suppose we have drawn randomly two adjacent regions R, R' of I. If we make the following hypothesis:

$$\mathcal{H}_0 = \text{“}R \textit{ is different from } R'\text{”}$$

then $\mathrm{C}(R, R')$ is just a realization of X, and so a value in the domain of f^*. If the observed value $\mathrm{C}(R, R')$ satisfies

$$P_r(\mathrm{X} \leq \mathrm{C}(R, R')) \leq \alpha \tag{5}$$

where α is fixed and small, then $\mathrm{C}(R, R')$ is small and belongs to a small set of realizations of X, having a little probability of appearance. These are the realizations of X defined by the set $\{y \leq \mathrm{C}(R, R')\}$. In that case, we can reasonably consider that R and R' in fact belong to the same region, and in that case we can merge them. That means that we reject $\mathcal{H}_0$ for the *alternative hypothesis* $\mathcal{H}_1$:

$$\mathcal{H}_1 = \text{“}R \textit{ and } R' \textit{ belong to the same region”}$$

When testing the merging of two regions R and R', there are two types of errors that can be made:

- keeping $\mathcal{H}_0$ while $\mathcal{H}_1$ is true, or
- rejecting $\mathcal{H}_0$ (*i.e.* accepting $\mathcal{H}_1$) while $\mathcal{H}_0$ is true.

These two events are noted respectively "$\mathcal{H}_0|\mathcal{H}_1$" and "$\mathcal{H}_1|\mathcal{H}_0$". It is not easy to quantify $P_r(\mathcal{H}_0|\mathcal{H}_1)$. However, provided f is a good approximation of f^*, we have:

$$P_r(\mathcal{H}_1|\mathcal{H}_0) \leq \alpha$$

Indeed, if $\mathcal{H}_0$ is true, the probability of rejecting it for $\mathcal{H}_1$ is no more than the probability of drawing two regions R and R' in I satisfying equation 5. Note that the vocabulary in the presentation of the errors is very important:

- when we choose $\mathcal{H}_0$, since it is hard to control $P_r(\mathcal{H}_0|\mathcal{H}_1)$, we say that we *keep* $\mathcal{H}_0$;
- when we choose $\mathcal{H}_1$, since we control $P_r(\mathcal{H}_1|\mathcal{H}_0)$, we can say that we *reject* $\mathcal{H}_0$ for $\mathcal{H}_1$

Thus, we obtain relatively strong bounds on one type of error, provided we choose α sufficiently small. This can be achieved by choosing α_1 in equation 2 small enough, since we also have $P_r(\mathcal{H}_1|\mathcal{H}_0) \leq \alpha_1$.

C.2 [Fiorio and Nock, 2000a]

A Concentration-Based Adaptive Approach to Region Merging of Optimal Time and Space Complexities

Christophe Fiorio
LIRMM
161, rue Ada
34392 Montpellier, France
fiorio@lirmm.fr

Richard Nock
Université des Antilles-Guyane
UFR SEN-Dept of Maths and CS
97159 Pointe-A-Pitre Cedex, France
rnock@univ-ag.fr

Abstract

In this paper, we investigate image segmentation as a statistical and computational problem. The observed image is sampled from a theoretical, unknown image, in which pixels are represented by distributions. Our objective is to approximate as best as possible the region segmentation in the ideal image, where each region has pixels with identical expectations, but adjacent regions have different pixel's expectations. From that model, a concentration-based statistical test for deciding region merging is built, limiting the risk of wrong merges. The analysis is carried out without any assumption on the distributions: we avoid in particular the classics of variance analysis, normality and homocedasticity. A practical approximation of the test is given, of constant time and space computation, which leads in turn to a segmentation algorithm of optimal complexity, easy to implement. Some experiments on various types of images shed light on the quality of the segmentations obtained.

1 Introduction

In the early stage of a vision process it is necessary to isolate "objects" in the scene before recognising them, a step called *image segmentation*. In this paper, we investigate image segmentation by region merging, which consists in building up regions by combining smaller regions, pixels being taken as primary regions. Our ultimate goal is to obtain a good partition of an image I into regions $\{R_1, R_2, ..., R_n\} = P(I)$, each region being 4-connected: between any of its pixels, a path exists, lying in the region, with horizontal or vertical connections.

In the field of image segmentation, many algorithms use statistical or mathematical techniques, models, relying on hypotheses made *a priori* on the image [7, 1, 6]. Many of these hypotheses constrain the models by introducing distributional assumptions on the image. Also, the computational costs can be huge. Other approaches focus rather on topologic and algorithmic properties of the image [4, 3], striving to find fast segmentation algorithms. However, statistical confidence in the results is generally not ensured. In that paper, we are interested in a bridging technique between the two approaches, based on a weakly constrained statistical model of image generation, which ultimately leads to a reliable segmentation algorithm of optimal complexity.

Our model supports the claim according to which an observed image I is principally a snapshot of an ideal object "scene" I^*, taken under particular conditions. Formally, in I^*, pixels are represented by distributions, used to generate the observed image. Ideal regions satisfy the homogeneity criterion that pixel's expectations are equal, expectations that are in turn different between adjacent regions. This model gives an intuitive notion of what is the *optimal segmentation*. This could be obtained by infinite sampling of the distributions, thus under infinitely many "conditions of observations". Averaging over the observed images would produce for each region its theoretical grey-level expectation, and we would observe the exact frontiers between regions. Optimising segmentation to recognise regions of I^* may be of great practical interest, in particular when processing numerous images of similar scenes, such as for real-time segmentation. In that case, we better look for fast results of overall quality, rather than single image optimisation at the possible expense of time complexity.

In the following section, we present our model of image generation. Then, we propose a statistical merging test to recognise if two observed, adjacent regions, belong actually to the same object in I^*. It has the major computational feature of being constant time and space approximable. It has the major statistical feature of being completely adaptive, and relies on a concentration bound whose proof sketch is given. Finally, we provide a segmentation algorithm of optimal complexity implementing this test, along with experiments that were conducted.

2 A Model for Image Generation in I^*

Let I denote the image observed, containing $|I|$ pixels ($|.|$ denotes the cardinality). I is described over a maximum, theoretical number, of g grey levels (generally, $g = 256$), but effectively contains g_I observed grey-levels. The model easily endorses RGB coding, by considering three separate models in the same way. I is an observation of a perfect scene I^* we do not know of, in which pixels are perfectly represented by a family of *distributions*, from which the observed grey-level is sampled. In all that follows, the "*"

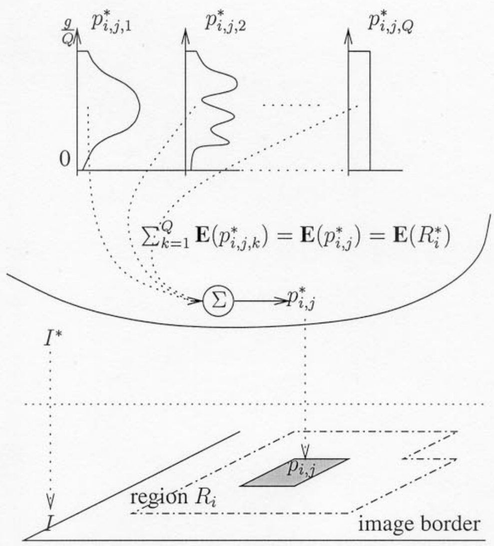

Figure 1: One pixel in I^* and I ($\mathbf{E}(.)$ denotes expectation).

superscript denotes objects taken from I^*, such as R^* for a region of I^*, to which corresponds a region R of I. The j^{th} pixel in the i^{th} region R_i of the image I, $p_{i,j}$, is obtained by the outcome of some random variable (r.v.) $p^*_{i,j} \in I^*$, itself sum of Q independent r.v. $p^*_{i,j,1}, p^*_{i,j,2}, ..., p^*_{i,j,Q}$ of I^* (each of these take values in the set $[0, g/Q]$, see Figure 1). Also, pixels are supposed sampled independently from each other. The role of Q is mainly to ensure the practical tractability of our merging test. Actually, as we shall see, the bigger the image I and the regions tested, the smaller Q can be chosen. As outlined in the introduction, each region in I^* satisfies an homogeneity property, formalised as follows:

- First, $\forall i \in \{1, 2, ..., |I|\}, \forall j \in \{1, 2, ..., |R^*_i|\}, \mathbf{E}(p^*_{i,j}) = \mathbf{E}(R^*_i)$ (expectations of all pixels in R^*_i are equal).
- Second, any couple of adjacent regions (R^*_i, R^*_j) satisfies $\mathbf{E}(R^*_i) \neq \mathbf{E}(R^*_j)$.

3 A statistical merging test in I

The test relies on some definitions related to ordinary variance analysis. In contrast however, our study does not rely on assumptions such as normality or homocedasticity, *i.e.* equal standard deviation between pixels in I^*. Though we do not observe it directly in

I, we denote as $p_{i,j,k}$ the outcome of $p^*_{i,j,k}$, *i.e.* $p_{i,j} = \sum_{k=1}^{Q} p_{i,j,k}$. In all that follows, the bar notation "$\overline{R}$" denotes the *observed grey-level average of some region*, that is, $(\sum_{j,k} p_{i,j,k})/|R_i|$. Similarly, the average grey-level of the observed image I is denoted $\overline{I}$.

Definition 1 *Let $\{R_1, R_2, ..., R_n\} = P(I)$ be a partition of I.*

- *The Intra-Pixel Variability (IPV) in I, is*

$$V_{\mathbf{p}}(I, P(I)) = \sum_{i,j,k} \left(p_{i,j,k} - \frac{p_{i,j}}{Q}\right)^2$$

- *The Intra-Region Variability (IRV) in I, is*

$$V_{\mathbf{i}}(I, P(I)) = \frac{1}{Q} \sum_{i,j} (p_{i,j} - \overline{R}_i)^2$$

- *The Extra-Region Variability (ERV) in I, is*

$$V_{\mathbf{e}}(I, P(I)) = \frac{1}{Q} \sum_{i} |R_i|(\overline{R}_i - \overline{I})^2$$

- *The Overall Image Variability of I is*

$$V(I) = \sum_{i,j,k} \left(p_{i,j,k} - \frac{\overline{I}}{Q}\right)^2$$

Some of the formulae presented before might appear misleading with respect to classical notations of variance analysis, but we have given them in the most compact form. For example, in order to follow the usual conventions, the IRV should be stated

$$V_{\mathbf{i}}(I, P(I)) \quad = \quad \sum_{i,j} Q \left(\frac{\sum_k p_{i,j,k}}{Q} - \frac{\sum_{j,k} p_{i,j,k}}{Q|R_i|}\right)^2$$

This strictly represents the same fomula as in definition 1. Also, classical result of variance analysis is preserved since we have

$$V(I) \quad = \quad V_{\mathbf{p}}(I, P(I)) + V_{\mathbf{i}}(I, P(I)) + V_{\mathbf{e}}(I, P(I)) \qquad (1)$$

(proof straightforward). The membership to some region $R^* \subseteq I^*$ of two adjacent regions R and R' of I is checked by a measure of their "proximity", the increase of IRV when merging them, $\Delta_{\mathbf{i}}(R, R')$ (In RGB coding, it is the sum of three IRV increases in each of the R, G and B coordinates, and each pixel results from the outcomes of $3Q$ distributions). Though we do not have access to any $p_{i,j,k}$, the fact that pixels are themselves "regions" of fixed size Q makes that $V_{\mathbf{p}}(I, P(I))$ is constant, and thus does not influence $\Delta_{\mathbf{i}}(R, R')$. The following lemma shows that the computation of $\Delta_{\mathbf{i}}(R, R')$ is indeed easy (proof straightforward).

Lemma 1 *The increase of IRV when merging R and R' satisfies*

$$\Delta_{\mathbf{i}}(R,R') = \frac{|R| \times |R'|}{Q(|R|+|R'|)}(\overline{R} - \overline{R'})^2 \tag{2}$$

Therefore, as long as we merge regions in the image, $V_{\mathbf{i}}(I,P(I))$ increases by *local* amounts, which can be computed in constant time, as well as for the parameter's update (means and sizes). The deviations of $\Delta_{\mathbf{i}}(R,R')$, when R and R' come from the same region in I^*, can be quantified as follows ($\#R$ is the number of distinct regions of size $|R|$, having grey levels included into those of R):

Theorem 1 *Let R, R' be two adjacent regions in I, belonging to the same region R^* in I^*. Fix*

$$m = \frac{6\sqrt{2}g^2\sqrt{\max\{|R|,|R'|\}}(|R|(|R|+2|R'|)^2+|R'|^3)}{Q^3|R||R'|(|R|+|R'|)}$$

Suppose that $g, Q, |R|, |R'|$ are such that $m < 1/2$. Fix some $\tau > 0$. Then,

$$\Pr[\Delta_{\mathbf{i}}(R,R') \geq \tau] < \frac{8\#R\#R'}{1-2m} \times e^{-\frac{2Q^2(|R|+|R'|)^2\tau}{g^2|R||R'|}} \tag{3}$$

Proof sketch: Though concentration bounds are not so hard to obtain provided unrealistic hypotheses are assumed, the necessity for us to stay as close as possible of a practical model of image generation led us to build a rather long proof for our bound. Moreover, ordinary concentration bounds, in that case, generally lead to too large bounds for practical use: in practice indeed, and we have observed it, this can lead to an over merged image with too few regions. That is why we have developped a bound taylor-made for our model, whose proof steps are the following ones. First, lemma 1 is used to shift from variability analyses to the deviation analysis of $|\overline{R} - \overline{R}'|$ beyond some τ', for some *fixed* regions R and R'. This is of better easiness to upperbound. Second, we remark that

$$(\overline{R \cup R'} - \overline{R}^*)^2 = (|\overline{R} - \overline{R}'|)^2 + Z \tag{4}$$

with R^* being the region of I^* to which R and R' belong, and Z defining the outcome of a r.v. . In other words,

$$\Pr[|\overline{R} - \overline{R}'| \geq \sqrt{\tau'}] < \Pr\left[(\overline{R \cup R'} - \overline{R}^*)^2 \geq \tau'\right] / \Pr[Z \geq 0] \tag{5}$$

$$= \Pr[\mathbf{B}]/\Pr[\mathbf{C}] \tag{6}$$

Then, a lowerbound $\Pr[\mathbf{C}] \geq \frac{1}{2} - m$ is obtained, where m is the median of Z (Z has 0 expectation). A concentration upperbound for the median m is obtained by combining the use of the Independent Bounded Difference Inequality (IBDI, [5]) and a derived version of Talagrand's theorem [5]. Finally, using the IBDI on $\Pr[\mathbf{B}]$ gives the final result of theorem 1. Since the result holds for two *fixed* regions R and R', the multiplication of the bound by the number of possible choices for R and R' gives the final, desired result. □

Theorem 1 can be used to build a merging threshold τ for $\Delta_{\mathbf{i}}(R,R')$, under which we can expect R and R' to come from the same region in I^*. This is done by solving the right inequality of theorem 1 for some particular probability. The merging test for two adjacent regions R, R' is then straightforward: "if $\Delta_{\mathbf{i}}(R,R')$ does not exceed τ, then merge R and R'".

4 Practical approximations and experimental results

Theorem 1 gives a way to practically quantify the risk we make of rejecting the merging of two regions R and R' that would actually be elements of the same region R^* in I^*. Of course, when executing n' rejections (R_1, R'_1), (R_2, R'_2), ..., $(R_{n'}, R'_{n'})$, the probability that the event of theorem 1 occurs can be upperbounded by n' times the upperbound given. Solving for τ the equality between the right member of Ineq. 3 and δ/n' gives the theoretical testing threshold, which is conveniently approximated for practical computability as follows. First, the segmentation algorithm we use is optimal but precludes the knowledge of n' in advance. We replace therefore n' by an upperbound for our segmentation algorithm, $2|I|$. n' appearing in a log factor for τ, this gives no visual difference in the results. Let

$$\#R = g_I^{|R|}/U_R \tag{7}$$

where U_R is a correction factor:

- $U_R = 0.5 * \frac{|R|!}{\lceil |R|/2 \rceil !}$ if $|R| < g_I$,
- $U_R = 0.5 * \frac{|R|!}{\lceil |R|-(g_I/2) \rceil !}$ otherwise (proof omitted due to the lack of space)

Factorials are approximated using Stirling's formula: $x! \approx \sqrt{2\pi x}(x/e)^x$. All tests are processed with reasonable values of Q, m, δ:

$$Q = \sqrt{g} \tag{8}$$

$$m = 1/4 \tag{9}$$

$$\delta = 1/|I| \tag{10}$$

While Q remains sufficiently small, this limits the risk of overmerging for significant, "visible" regions. The threshold τ, computable in constant time, is finally:

$$\tau(R, R') = \frac{g|R||R'|}{2(|R| + |R'|)^2} \log\left(\frac{32|I|^2 g_I^{(|R|+|R'|)}}{U_R U_{R'}}\right) \tag{11}$$

Finally, each image is preprocessed in linear time and space to merge adjacent pixels having the same grey-level: this does not degrade optimal solutions, but increases the average region's size, and thus the efficiency of the merging test. The segmentation algorithm consists in making a single pass of the image, and testing (if possible) the merging of the current pixel (or region) with the upside pixel's region, and then the left one. As showed in [3], a particular data structure allows to obtain optimal (linear) time and space complexities. The running time for the C implementation over a Pentium II PC never exceeded the second for every experiment.

Figures 2, 3 and 4 present some experiments on various images. We emphasise that *all* segmentations were obtained using the *same* tuning for all parameters, even if the images are much different from each other (synthetic, medical, aerial images, portraits, etc.). In particular, the quality of the results does not stem from the optimisation of the algorithm on each image. Also, images are segmented without any preprocessing (*e.g.* noise filtering). Only `street` was Gaussian smoothed according to [2]'s experimental setup, to

make accurate comparisons.
In `cornouaille`, the algorithm manages a great reduction in the boat's noise, particularly on its left part, whereas it keeps almost intact the overall shape and the name of the boat (note also the segmentation of the man in the bottom-right part). In image `synth`, which mainly features a ramp gradient on the left, and a noisy rectangle area on the right, the algorithm detects exactly three regions, approximating those most people would consider as perceptually distinct. The `street` image shows that our algorithm outperforms two other approaches on two aspects: the number of regions selected compared to [2] (*e.g.* the van), and the accuracy of the segmentation compared to [3] (*e.g.* the grass, highly noisy).

References

[1] S. Dugelay, C Graffigne, and J.-M. Augustin. Segmentation of multibeam acoustic imagery in the exploration of the deep-sea bottom. In *Int. Conf. on Pattern Recognition*, pages 437–446, 1996.

[2] P. F. Felzenszwalb and D. P. Huttenlocher. Image segmentation using local variations. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 98–104, 1998.

[3] Christophe Fiorio and Jens Gustedt. Two linear time Union-Find strategies for image processing. *Theoretical Computer Science*, 154:165–181, 1996.

[4] W. G. Kropatsch and S. B. Yacoub. A revision of pyramid segmentation. In *Int. Conf. on Pattern Recognition*, pages 477–481, 1996.

[5] C. McDiarmid. Concentration. In M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, and B. Reed, editors, *Probabilistic Methods for Algorithmic Discrete Mathematics*, pages 1–54. Springer Verlag, 1998.

[6] R. Watzel, K. Braun, A. Hess, W. Zuschratter, and H. Scheich. Restoration of dendrites and spines with the objective of Topologically Correct Segmentation. In *Int. Conf. on Pattern Recognition*, pages 472–476, 1996.

[7] S. C. Zhu and A. Yuille. Region competition : Unifying snakes, region growing, and bayes/MDL for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 884–900, 1996.

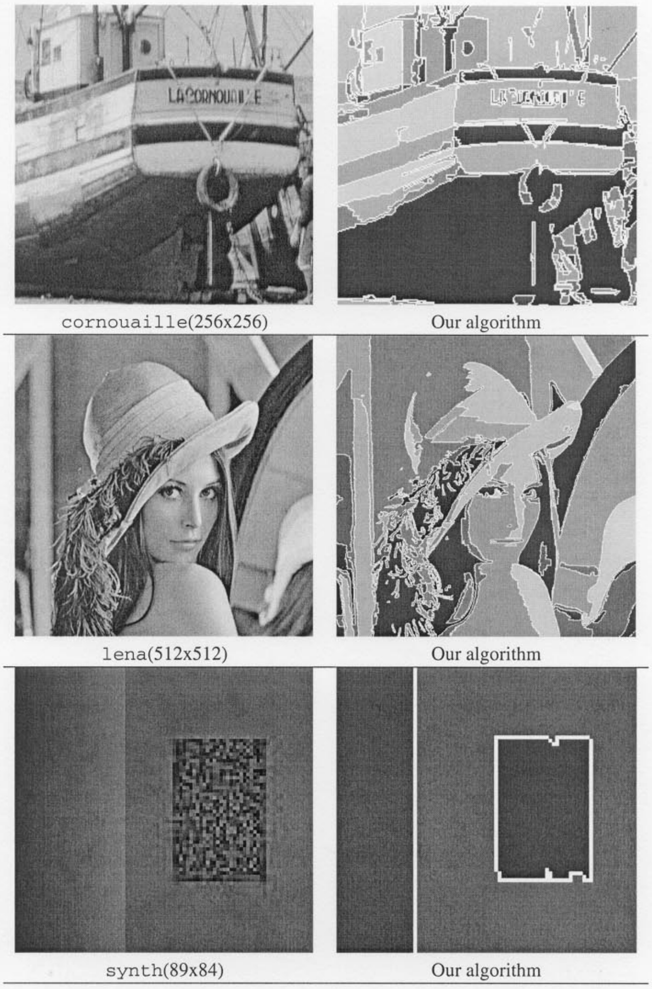

Figure 2: Some experiments conducted (regions are white bordered, and averaged inside).

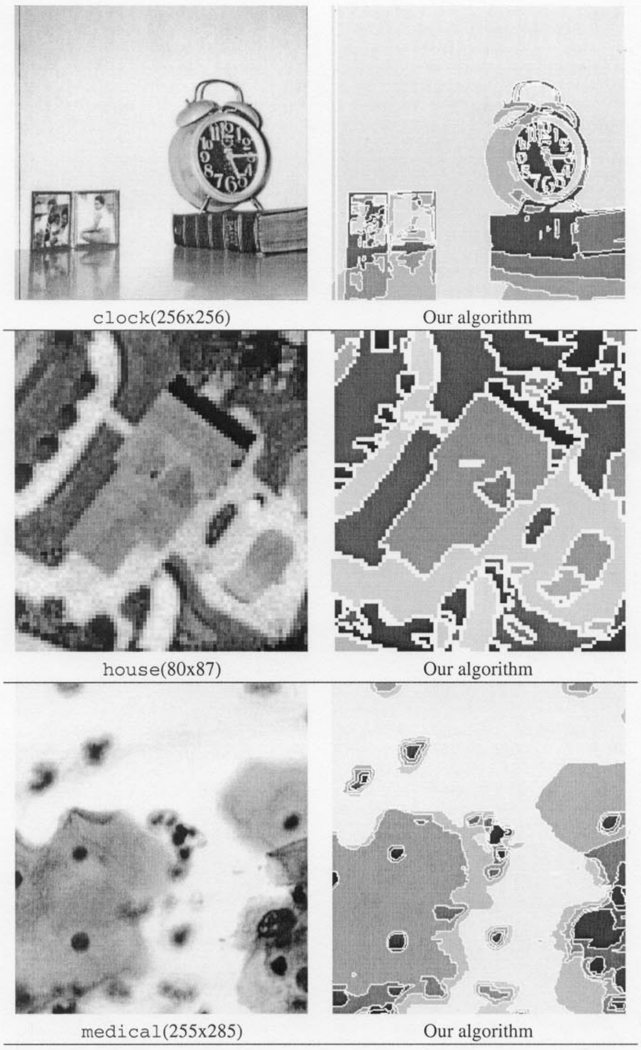

Figure 3: More experiments (regions are white bordered, and averaged inside).

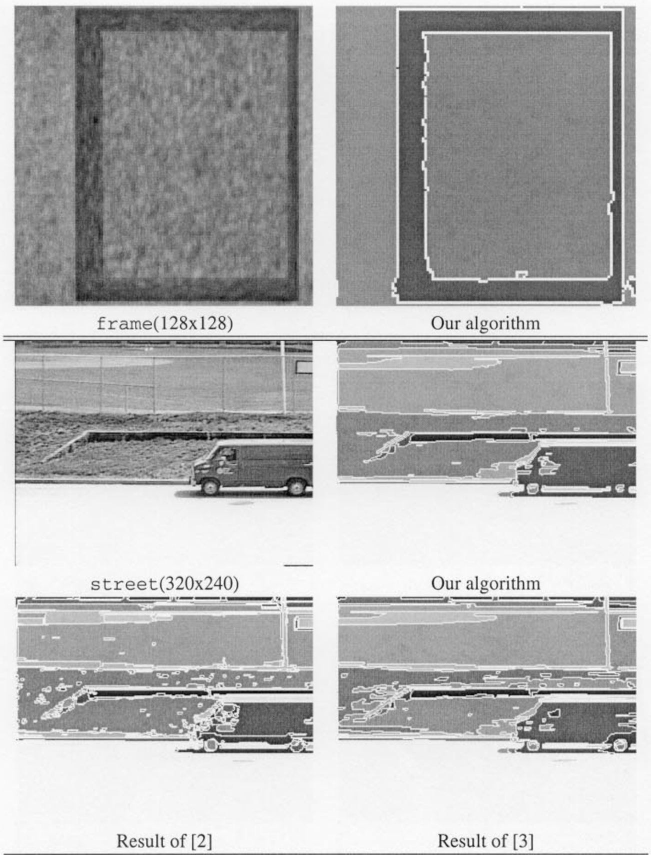

Figure 4: More experiments (regions are white bordered, and averaged inside).

C.3 [Fiorio and Nock, 2000b]

Sorted Region Merging to Maximize Test Reliability

Christophe Fiorio* Richard Nock†

Abstract

In this paper, we discuss an algorithmic approach to region merging which is built on a recent statistical work on the way to decide merging while keeping optimal complexity. In that latter work, a concentration-based statistical test is proposed, having the particularity to reduce the error occurring when rejecting the merging of two observed regions coming from the same true region. In this paper, we propose a preliminary ordered-based algorithmic procedure to cope with the errors occurring when merging two different regions in the first approach, thereby leading to a fast algorithm tailor-made for the reduction of both kinds of error. Experimentations proposed on images used without any preprocessing shed light on the quality of segmentations obtained.

1 Introduction and related work

In the early stage of a vision process it is necessary to isolate "objects" in the scene before recognizing them, a step called *image segmentation*. In this paper, we investigate image segmentation by region merging, which consists in building up regions by combining smaller regions, pixels being taken as primary regions. Our ultimate goal is to obtain a good partition of an image I into regions $\{R_1, R_2, ..., R_n\} = P(I)$, each region being 4-connected: between any of its pixels, a path exists, lying in the region, with horizontal or vertical connections.
In the field of image segmentation, many algorithms use statistical or mathematical techniques, models, relying on hypotheses made *a priori* on the image [8, 1, 7]. Many of these hypotheses constrain the models by introducing distributional assumptions on the image. Also, the computational costs can be huge. Other approaches focus rather on topologic and algorithmic properties of the image [4, 2], striving to find fast segmentation algorithms. However, statistical confidence in the results is generally not ensured. In a recent work, we have been interested in a bridging technique between these two approaches [3]. It was based on recent results about concentration inequalities [5] adapted to image segmentation, themselves derived from a particular model of image segmentation . Informally, this models supports the claim according to which an observed image I is principally a snapshot of an ideal object "scene" I^*, taken under particular conditions. Formally, in I^*, pixels are represented by distributions, used to generate the observed image. Ideal regions satisfy the homogeneity criterion that pixel's expectations are equal, expectations that are in turn different between adjacent regions. This model gives an intuitive notion of what is the *optimal segmentation*. This could be obtained by infinite sampling of the distributions, thus under infinitely many "conditions of observations". Averaging over the observed images would produce for each region its theoretical grey-level expectation, and we would observe the exact frontiers between regions.
From that model and recent concentration results [5], we derived [3] the probability that the intra-region variability (IRV) increase when merging two regions deviates up to "large values" when the two regions come from the same object in I^*. Solving the minimum variability τ for some (small) probability α, we obtain a simple test allowing to decide if two regions can be merged or not: if the increase in IRV exceeds τ, then we do not accept to merge the regions, while being confident up to risk α that the two regions *are not* element of the same object in I^*. The notion of IRV we use borrows almost everything to its equivalent variability notions in Variance Analysis, yet we have adapted it to our model of image generation. Is is interesting to note that it can be computed in optimal (constant) time, and represent therefore a convenient choice of metric for our purposes [3].
Though our test handles the *first order risk*, there is no way to be formally confident when accepting to merge two regions, that is, we cannot be sure that two regions merged come from the same true region. This risk is crucial when testing small regions.
In this paper, we propose an algorithmic approach to solve this problem. We sort the pixels prior to their processing, according to the (increasing) IRV measured in their neighborhood, a task at a reasonable computational expense. By this way, the algorithm, which repeatedly tests the merging

*LIRMM, 161 rue Ada, 34392 Montpellier, France. email: `fiorio@lirmm.fr`
†Université des Antilles-Guyane, UFR SEN-Dept of Maths and CS, 97159 Pointe-A-Pitre Cedex, France. email: `rnock@univ-ag.fr`

of two adjacent pixel's regions, begins by merging regions sharing similar local properties (variability), thus likely to come from the same true region. As regions grow, the test becomes stronger, and is more accurate whenever it rejects a merging. Due to the space constraints, we refer the reader to [3] for the bases of the model, the concentration bounds, and more generally all the statistical material. We concentrate here on the algorithmic material.

2 The algorithm ψScanLine

We use as subprocedure the optimal-time and space algorithm ScanLine (algorithm 1, [2]). It is based on a union-find structure which represents regions by trees of pixels belonging to the same region (see [6] for more information on union-find algorithms). The predicate $\mathcal{P}$ returns "Yes" or "No" depending on whether the two tested regions can be merged or not. The complete modified

Algorithm 1: `ScanLine` (I)

Input: an image I with r rows and c columns
for $i = 0$ ***to*** $r - 1$ **do**
  for $j = 0$ ***to*** $c - 1$ **do**
    if $j > 0$ **then**
      left = `Find` $(I[i, j-1])$;
      current = `Find` $(I[i, j])$;
      if $\mathcal{P}$(*left,current*) **then** `Union` (*left,current*) ;
    if $i > 0$ **then**
      up = `Find` $(I[i-1, j])$;
      current = `Find` $(I[i, j])$;
      if $\mathcal{P}$(*up,current*) **then** `Union` (*up,current*) ;

algorithm, ψScanLine (for Probabilistic Sorted Iterated ScanLine), works as follows. Before the first **for**, we sort the pixels $p, \ldots \in I$ according to a function $f(p), \ldots$. Denote as $R_k(p)$ the $k \times k$ squared region centered on pixel p, or centered as possible if p lies near the border of I. The function $f(p)$ is:

$$f(p) = \sum_{p' \in R_k(p)} \left(I(p') - \overline{R_k(p)} \right)^2$$

Here, $I(p')$ is the grey-level of pixel p'. In the case of RGB images, $f(p)$ sums three similar functions computed over each coordinate. Second, the predicate $\mathcal{P}(R, R')$ is the same as [3]. It is based on the IRV increase, $\Delta_{\mathbf{i}}(R, R')$, which equals $\frac{|R| \times |R'|}{\sqrt{g}(|R|+|R'|)}(\overline{R} - \overline{R'})^2$ in our model, where $|.|$ denotes the size in pixels and $\overline{R}$ is the grey-level average of R (see remark in [3] on the fact that this is not the usual definition from statistics). In RGB images, we calculate three IRVs in each coordinate. The predicate $\mathcal{P}$ returns true iff $\Delta_{\mathbf{i}}(R, R')$ does not exceed a quantity $\tau(R, R')$:

$$\tau(R, R') = \frac{g|R||R'|}{2(|R| + |R'|)^2} \log \left(\frac{32|I|^2 g_I^{(|R|+|R'|)}}{U_R U_{R'}} \right)$$

where for any region R, $U_R = 0.5 * \frac{|R|!}{\lceil |R|/2 \rceil !}$ if $|R| < g_I$ (g_I is the effective number of grey-levels in I), and $U_R = 0.5 * \frac{|R|!}{\lceil |R| - (g_I/2) \rceil !}$ otherwise. As shown in [3], this predicate is actually constant time and space computable, a very convenient property. Finally, algorithm 1 is ran until convergence, *i.e* until there is no more merging on the current iteration (experimentally, the number of iterations did not exceed 6, even on 1024x1024 images). Some experiments conducted with ψScanLine are shown in figure 1 ($k = 13$). Note that the images were used without any preprocessing, and with the same parameter setting for all. The results show the quality of the segmentations obtained, even when using such raw material. They also show that the algorithm is able to handle regions with smooth gradients (`mug`). This partly comes from the fact that $f(p)$ reasonably fluctuates in regions with uniform (or smooth) gradients. Therefore, all pixels are grouped in the ordered list, and if they do not have too different colors, they will be merged by ψScanLine.

References

[1] S. Dugelay, C. Graffigne, and J.-M. Augustin. Segmentation of multibeam acoustic imagery in the exploration of the deep-sea bottom. In *Int. Conf. on Pattern Recognition*, pages 437–446, 1996.

[2] C. Fiorio and J. Gustedt. Two linear time Union-Find strategies for image processing. *Theoretical Computer Science*, 154:165–181, 1996.

[3] C. Fiorio and R. Nock. A Statistical Approach to Region Merging of Optimal Complexity. Technical Report LIRMM-IFA-99101, Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier, 1999. Submitted, available at: `http://www.lirmm.fr/~fiorio`.

[4] W. G. Kropatsch and S. B. Yacoub. A revision of pyramid segmentation. In *Int. Conf. on Pattern Recognition*, pages 477–481, 1996.

[5] C. McDiarmid. Concentration. In M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, and B. Reed, editors, *Probabilistic Methods for Algorithmic Discrete Mathematics*, pages 1–54. Springer Verlag, 1998.

[6] K. Mehlhorn. *Data Structures and Algorithms 1: Sorting and Searching*. Springer, 1984.

[7] R. Watzel, K. Braun, A. Hess, W. Zuschratter, and H. Scheich. Restoration of dendrites and spines with the objective of Topologically Correct Segmentation. In *Int. Conf. on Pattern Recognition*, pages 472–476, 1996.

[8] S. C. Zhu and A. Yuille. Region competition : Unifying snakes, region growing, and bayes/MDL for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 884–900, 1996.

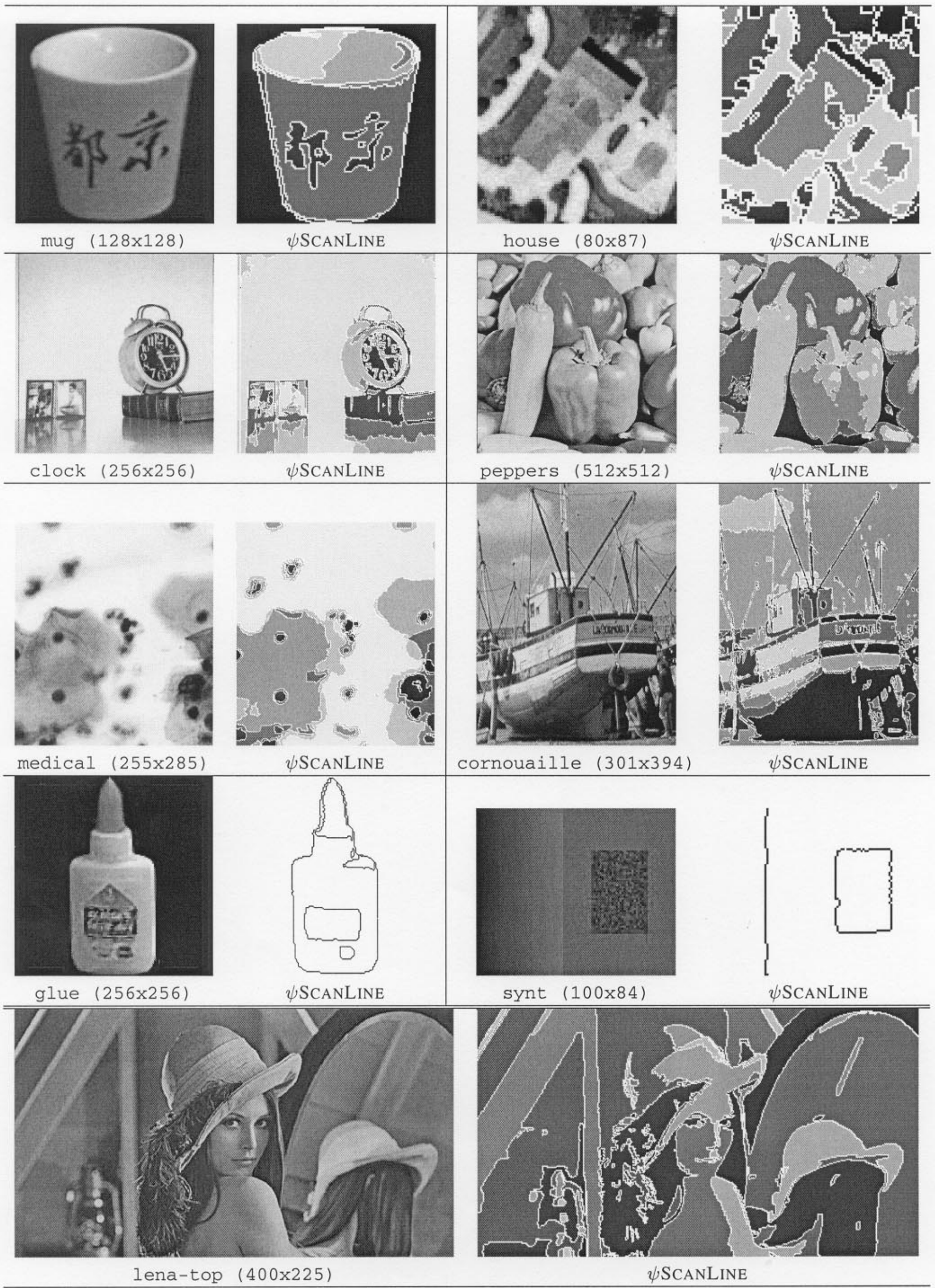

Figure 1. Experiments conducted with ψSCANLINE (regions are either white bordered and averaged inside, or black bordered and white inside).

C.4 [Nock, 2001b]

Fast and Reliable Color Region Merging inspired by Decision Tree Pruning

Richard Nock
Université Antilles-Guyane
Dépt Scientifique Inter-Facultaire
Campus de Schoelcher, B.P. 7209
97275 Schoelcher, France
rnock@martinique.univ-ag.fr

Abstract

In this paper, we exploit some previous theoretical results about decision tree pruning to derive a color segmentation algorithm which avoids some of the common drawbacks of region merging techniques. The algorithm has both statistical and computational advantages over known approaches. It authorizes the processing of 512×512 images in less than a second on conventional PC computers. Experiments are reported on thirty-five images of various origins, illustrating the quality of the segmentations obtained.

1 Introduction

It is established since the Gestalt movement in psychology [14] that perceptual grouping plays a fundamental role in human perception. Even though this observation is rooted in the early part of the XXth century, the adaptation and automation of the segmentation (and more generally, grouping) task with computers has remained so far a tantalizing and central problem for image processing. This is all the more important for computers as grouping authorizes the reduction of the size of data, and, by the way, can reduce dramatically the space and time complexities for some post-processing tasks, in a field where data are easily available and in huge quantities.

Roughly speaking, the problem can be presented as the transformation of the collection of pixels of an image into a meaningful arrangement of regions and objects [8]. But, how can we identify objects ? The Gestalt movement identified some factors leading to the perception of objects : symmetry, similarity, parallelism, etc. [4]. Though these rules are conceptually satisfying to explain the phenomenon at the human level, their lack of details makes it hard to see them as directly implementable algorithms [4], even if some of them can be extended to rigorous algorithms [15]. So far, image segmentation techniques have tackled the problem by studying mathematical properties of the image seen [16], or more rarely by a direct emphasis on algorithmic and computational issues [3].

There are four large categories of approaches to image segmentation [16], one of which is of direct interest to us : region growing and merging techniques. In this group of algorithms, regions are sets of pixels with homogeneous properties and they are iteratively grown by combining smaller regions or pixels, pixels being taken as elementary regions. Region growing techniques usually work with a statistical test to decide the merging of regions [11, 16].

In the field of image segmentation (or more generally, image processing), Machine Learning (ML) techniques might appear tailor-made for high-level tasks, such as in [9]. One might wonder whether applying such techniques or their adaptations to image processing based on low-level cues (typically, image segmentation) might be well worth the try, in a field where the approaches are already numerous, of many kinds (local filtering, balloons, snakes, MDL/Bayesian, region growing, etc. [16]), and where heavy mathematical approaches have obtained significant results (see *e.g.* the results of [16]). We think some clues coming from image processing indicate that ML-derived techniques might be of interest already for low-level processing tasks. First, vision is widely accepted as an *inference* problem, *i.e.* the search of what caused the observed data [4]. Second, as argued by [12], low-level image segmentation should not aim at producing a correct segmentation, but rather a good *approximation* given the post-processing stages of the image. One more reason can give evidence for the possible interest in using ML-related techniques. ML and Computational Learning Theory (COLT) works such as [7] have brought a number of useful results, or found a way to use a number of results, in statistics or probability, which remove many undesirable distribution assumptions on the data. The most commonly used theoretical founding model for ML/COLT is the PAC model of Valiant [13], which removes any such distribution assumptions. In image segmentation, there have been a number of probabilistic approaches to the problem [1, 16] or [4] (Chp. 18). Most of them rely on heavy, penalizing distribution assumptions on the data (such as normality), and finally some recent prominent work in image segmentation has made emphasis on more conventional, non-probabilistic approaches, such as discriminant analysis-type [12, 2].

Our aim in this paper is clearly *not* to propose a ML algorithm to segment images, but rather to exploit some of the common points between image segmentation and decision-tree pruning, and then adapt the theoretical ML/COLT ideas and tools of [7] to our setting, to derive a new segmentation algorithm. This paper consists therefore of an original (and novel) adaptation of previous ML/COLT work to low-level image processing, a rather seldom result in our context. The next section goes in depth in the common points between decision-tree pruning and image segmentation. It is followed by a section presenting a model of image generation. Then, two sections detail respectively the statistical and algorithmic contents of our approach. The last section discusses experiments on numerous images.

2 From pruning to merging

A decision tree is a classifier making recursive partitions of a representation space, according to some classes. Consider fig. 1, showing a 2D representation space restricted to the integer couples of the domain $[0, x_4] \times [0, y_3]$ for some $x_4, y_3 > 0$. Suppose that there are g classes (not shown). The point is that a probability distribution is given to domain×classes. Each couple (observation, class) is called an *example*, and the problem is to fit as best as possible the domain of all examples, using only a potentially small subset of this domain, sampled according to the distribution. Decision trees are very efficient ways to tackle the problem using a very simple formalism [7]. The quality of a decision tree is evaluated by its error probability over the whole domain, also called structural risk, which we can only estimate using the error frequency over the examples seen, also called empirical risk [7]. The classification of an observation is made as follows: each internal node (and the root) of the tree is labeled by a test on a variable of the observations; each leaf is labeled by a class (not shown in fig. 1). The classification process starts from the root of the tree. If the observation satisfies the current test, then it follows the right path, otherwise it follows the left path, until it reaches a leaf, and therefore is given the corresponding class. In fig. 1, an observation for which $x = x_2$ and $y = y_2$ would fall in region A, as shown by the dotted path. The most popular decision tree learning algorithms (CART, C4.5 [7]) proceed by growing a very large decision tree to fit as best as possible the examples seen, and then pruning it to get (hopefully) a good fit of the whole domain. Pruning a

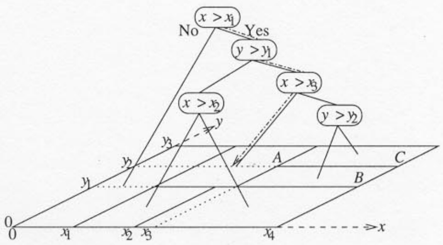

Figure 1. A domain and its recursive partition by a decision-tree (see text for details).

tree consists in removing iteratively an internal node and its subtree, thereby replacing the node by a class label. Pruning the node whose test is "$x > x_3$" in image 1 would boil down to merge regions A, B and C.

If we consider that the color values of pixels in an image are the result of a theoretical value (the one of the region they belong to), combined with a random variability factor (such as noise), then the task of region merging can be loosely reformulated as the recognition of the regions having the same theoretical values in an observed image. This problem is quite similar to decision tree pruning (consider that the image has $x_4 y_3$ pixels in fig. 1); more importantly, the tools used in both problems can be the same: the concentration of random variables to give confidence bounds, either for the theoretical color values of groups of pixels, or for the structural risks in decision-tree pruning.

However, our task appears to be more difficult than pruning, since we generate much more potential configurations. Consider *e.g.* fig. 1: the tree has only 7 possible prunings (including the original tree), whereas there are 42 possible segmentations of the domain with the 6 initial regions.

Image processing has potentially two characteristics which make it a good candidate for the adaptation of particular ML techniques. First, the quality of the pruning strongly depends on the available quantity of data, and small datasets typically lead to uncontrollable over-pruning. In image segmentation, the material is available in huge quantities. Second, image segmentation is a field in which fast algorithms are obviously crucial with the advent of real-time (video) image processing, but practice shows that "fast" is a field-dependent subjective notion: for example, [1] consider a "very fast" algorithm extracting a few dozens of regions in a 512×512 image in less than 10 seconds, but on an high-end UltraSPARC workstation; [12] give execution times on 100×120 images of about 2 minutes on conventional machines, and there are many other examples. We show in that paper that using our ML-derived tools can bring highly competitive (or better) results, in reduced times.

3 Notations and models

The notation $|.|$ stands for cardinal. The observed image, I, contains $|I|$ pixels, each containing **R**ed-**G**reen-**B**lue (**RGB**) values, each of the three belonging to the set $\{1, 2, ..., g\}$. We have deliberately chosen not to use complex formulations of the colors, such as the $L*u*v*$ space [1]. I is an observation of a perfect scene I^* we do not know of, in which pixels are perfectly represented by a family of *distributions*, from which each of the observed color-level is sampled. In I^*, the optimal (or true) regions represent theoretical objects sharing a common homogeneity property:

- all pixels of a given true region have identical expectations for each **RGB** color-level,
- the expectations of adjacent regions are different for at least one **RGB** color level.

I is obtained from I^* by sampling each theoretical pixel for observed **RGB** values. Fig. 2 presents an example of a color-level for one pixel in I^* and how to generate the corresponding observed color-level of the pixel in I. In each pixel of I^*, each color-level is replaced by a set of Q independent random variables (r.v.) taking positive values on domains bounded by g/Q, such that any possible sum of outcomes of these Q r.v. with non-zero probability belongs to $\{1, 2, ..., g\}$.
The sampling of each pixel and its color levels are supposed independent from each other. Our model of image generation does not unfortunately prevent us to depend on this usual assumption (also widely used in ML/COLT). Our experiments shall demonstrate that it does not seem to have an impact on the results of the segmentation. It is important to note that this is the only assumption we make on I^*. In particular, we do not make any further assumption on the nature of each distribution (such as normality, homoscedasticity), which can be therefore different from one another in each pixel of a region, as long as the sum of their expectations is constant for each color inside a true region.
Our goal is then straightforward: find observed regions in I approximating as best as possible the true regions in I^*.
Q is certainly the less intuitive parameter of our model. We have chosen to introduce it for our model and our analyses to be as general as possible (standard analyses would fix $Q = 1$). Practically speaking, Q represents the major advantage to be a trade-off parameter, adjustable to obtain a compromise between the power of the model and the quality of the observed results. Indeed, if Q is small (say, $Q = 1$), then scarcely nothing can be estimated reliably for small regions in I, but our model is the most general possible. On the other hand, if Q is too large, then our model becomes restricted ($Q = g$ brings back a conventional Binomial law), but our estimations are the most reliable. Between, the user can choose a convenient value to keep a powerful enough model while ensuring reliable estimations on I.

4 The merging test and its properties

Our main result is based on the concentration of random variables, a tool widely used in ML/COLT literature [7]. Our model necessitates a recent result due to [10]:

Theorem 1 *(The independent bounded difference inequality, [10]) Let $\boldsymbol{X} = (X_1, X_2, ..., X_n)$ be a family of independent r.v. with X_k taking values in a set A_k for each k. Suppose that the real-valued function f defined on $\prod_k A_k$ satisfies $|f(\boldsymbol{x}) - f(\boldsymbol{x}')| \leq c_k$ whenever vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$ differ only in the k-th coordinate. Let μ be the expected value of the r.v. $f(\boldsymbol{X})$. Then for any $t \geq 0$,*

$$\mathbf{Pr}(f(\boldsymbol{X}) - \mu \geq t) \quad \leq \quad e^{-\frac{2t^2}{\sum_k (c_k)^2}} \qquad (1)$$

In our case, we now prove that with high probability, the observed average $\overline{R}_a$ of any region R in I shall not deviate too much from its theoretical expectation $\mathbf{E}_a(R)$ for a single color-level a. By theoretical expectation, we mean the average of the sum over each of its pixels of the expectations of its Q distributions for color a in I^*.

Theorem 2 *Let R be a region in I. Let $\mathcal{R}_l$ be the set of regions having l pixels in I. Fix an $a \in \{\boldsymbol{R}, \boldsymbol{G}, \boldsymbol{B}\}$. $\forall \delta' > 0$, the probability that there exists a region in $\mathcal{R}_{|R|}$ such that*

$$\left|\overline{R}_a - \boldsymbol{E}_a(R)\right| \quad \geq \quad g\sqrt{\frac{1}{2Q|R|}\left(\log \frac{2|\mathcal{R}_{|R|}|}{\delta'}\right)} \qquad (2)$$

is no more than δ'.

(Proof omitted due to the lack of space). Of course, the probability of occurrence of the event for *some* of the **R**, **G**, or **B** levels, is no more than $3\delta'$. What is interesting in theorem 2 is that region R is not required to belong to a true region of I^*. Indeed, if the region merging algorithm has merged together subsets of true regions $O_1, O_2, ..., O_k$ in R, then the bound of theorem 2 still holds. The use of theorem 2 will be straightforward. The probability that there exists a region in I (regardless of its size) for which the average of some of its **RGB** color deviates from its expectation by more than the right-hand-side of inequality 2 is no more than $3|I|\delta'$, since the region can have size 1, 2, ..., or $|I|$. Then, if we fix

$$\delta \quad = \quad 3|I|\delta' \qquad (3)$$

and

$$b(R) \quad = \quad g\sqrt{\frac{1}{2Q|R|}\left(\log \frac{2}{\delta'} + \log |\mathcal{R}_{|R|}|\right)} \qquad (4)$$

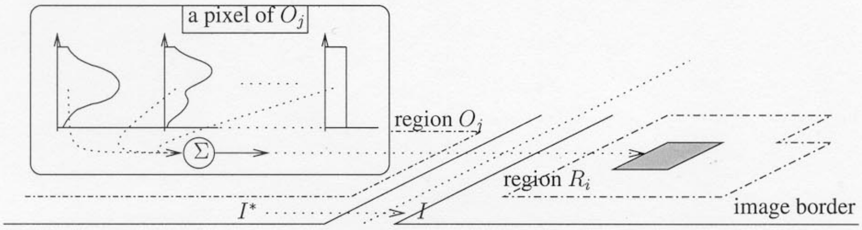

Figure 2. Generation of a single color-level for one pixel from I^* to I.

then we know that, with high probability ($> 1-\delta$), any possible region R will have a bounded variation of $\overline{R}_a$ around $\mathbf{E}(R)_a$ for *all* values $a \in \{\mathbf{R}, \mathbf{G}, \mathbf{B}\}$. Because of the triangle inequality, we know that any two regions R and R' having the same expectations ($\mathbf{E}_a(R') = \mathbf{E}_a(R) = q_a$, $\forall a \in \{\mathbf{R}, \mathbf{G}, \mathbf{B}\}$) shall observe a quantity $|\overline{R'}_a - \overline{R}_a|$ which will be also concentrated around zero, up to radius $b(R) + b(R')$ actually, since $|\overline{R'}_a - \overline{R}_a| \leq |\overline{R'}_a - q_a| + |\overline{R}_a - q_a|$ (triangle inequality). If $|\overline{R'}_a - \overline{R}_a| \leq b(R) + b(R')$ for all $a \in \{\mathbf{R}, \mathbf{G}, \mathbf{B}\}$, then we can suppose that R and R' belong to the same true region in I^*, and merge them. This gives the merging predicate $\mathcal{P}(R, R')$ of our region merging algorithm, returning whether R and R' can be merged or not:

$$\mathcal{P}(R,R') = \begin{cases} \texttt{true} & \text{iff } \forall a \in \{\mathbf{R},\mathbf{G},\mathbf{B}\}, \\ & |\overline{R'}_a - \overline{R}_a| \leq b(R) + b(R') \\ \texttt{false} & \text{otherwise} \end{cases}$$

We also need to compute the value $|\mathcal{R}_{|R|}|$. Because of the fact that exact values are difficult to compute, we have used a practical bound $|\mathcal{R}_l| \approx \left((l+2)^{\min\{g,l\}}\right)/(l+1)$ (proof omitted) which follows as closely as possible the true value, being smaller for the small values of l (thus, for small regions, where typically $l < g$), and larger for the large values of l. The risk in computing an upperbound for $|\mathcal{R}_l|$ was to get a too large upperbound, thereby leading to a too large risk of over-merging (we shall see it in the next section).

5 The algorithm and its properties

Suppose that the image I contains r_I rows and c_I columns. This represents $N = 2r_Ic_I - r_I - c_I$ couples of adjacent pixels (in 4-connexity). Name S_I as the set of these couples. Denote $f(.)$ the function which takes a couple of pixels (p, p'), and returns the maximum of the three color differences (**R**, **G** and **B**) in absolute value between p and p'. Denote as $\texttt{quicksort}(S_I,f)$ the output of the quicksort algorithm for set S_I, in increasing order of function $f(.)$. For any pixel p of I, we denote as $R(p)$ the current region to which p belongs in I. The algorithm is called `PSIS`, for Probabilistic Sorted Image Segmentation.

Algorithm 1: `PSIS(`I`)`

Input: an image I
S'_I=`quicksort(`S_I,f`);`
for $i = 0$ ***to*** $N-1$ **do**
  /*(p_i, p'_i) is the i^{th} couple of S'_I*/
  if $R(p_i) \neq R(p'_i)$ ***and*** $\mathcal{P}(R(p_i), R(p'_i))$=`true` **then**
    `Union(`$R(p_i)$,$R(p'_i)$`);`

One might wonder why we have chosen to order the couples of pixels in `PSIS`. Indeed, such a requirement is *a priori* not clear from our theory, and it brings an $\mathcal{O}(|I| \log |I|)$ complexity (sometimes smaller, [2]) instead of the (almost) linear complexity reachable without this stage. There are basically three kind of errors our algorithm can suffer with respect to the optimal segmentation. First, under-merging represents the case where one or more regions obtained are strict subparts of true regions. Second, over-merging represents the case where some regions obtained strictly contain more than one true region. Third, there is the "hybrid" (and most probable) case where some regions obtained contain more than one strict subpart of true regions. Ordering the pixels test is a way to limit this third kind of error, since we approximate the following property:

(P) All merging tests inside true regions are made before any merging test between true regions.

Obviously, **(P)** is unreachable in practice; however, from a purely theoretical point of view, **(P)** has a key effect when used with our statistical material: `PSIS` only suffers over-merging with probability $> 1 - \delta$ (eq. 3). Indeed, we know that for any regions R, R' coming from the same true region in I^*, we have $\forall a \in \{\mathbf{R}, \mathbf{G}, \mathbf{B}\}, |\overline{R'}_a - \overline{R}_a| \leq |\overline{R'}_a - q_a| + |\overline{R}_a - q_a|$ where $q_a = \mathbf{E}_a(R') = \mathbf{E}_a(R)$. With probability $> 1-\delta$, we also know (theorem 2 and eq. 3) that $|\overline{R'}_a - \overline{R}_a| \leq b(R) + b(R')$ for any regions R, R' coming from the same true region in I^*. Since this is our merging test, and since **(P)** holds, it follows that the segmentation obtained is an under-segmentation of I^*.
Therefore, theoretically speaking, **(P)** allows to eliminate

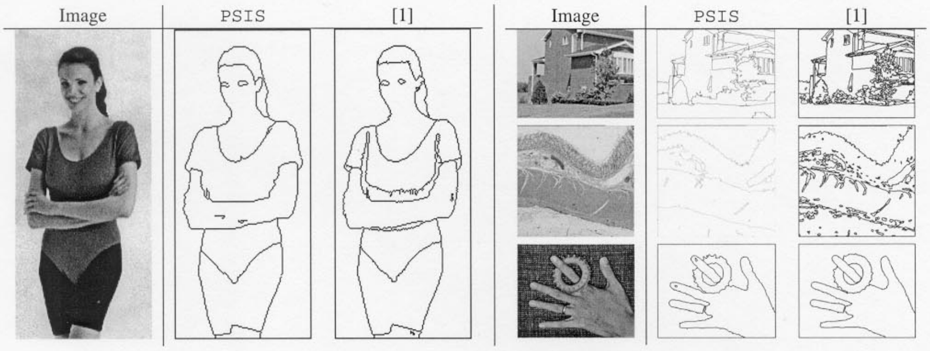

Figure 3. Results obtained by PSIS and [1] on four images. Region are white with black borders.

under-merging and the hybrid error case with high probability; this motivated us in fast approximations of it. In fact, over-merging can somewhat be controlled in practice for regions with size large enough (typically $\geq g$) as long as the bound for $|\mathcal{R}_l|$ is not too large (see Eq. 4). This is the reason for our choice of the upperbound for $|\mathcal{R}_l|$.

6 Experimental results

Due to the lack of space, we report experiments only on thirty-five images among all on which PSIS was tested (figures 3, 4 and 5). While looking at the results, the reader may keep in mind that:

> the values of the parameters of PSIS are the same for all images: $\delta' = 1/(3|I|^2)$ (theorem 2) and $Q = 32$. Furthermore, the images were used as they are, *i.e.* without any preprocessing.

Therefore, the results of PSIS do not stem from any domain- or image-dependent preprocessing or parameter tuning. PSIS was implemented in C. Fig. 3 displays some results to be compared with [1, 16]. While PSIS obtained record times for processing these images, the results are highly competitive with the other approaches. Consider the woman image. Her bust is better segmented than [1]. When looking at [16]'s result for this image, our result appears to be better: after over a hundred iterations, [16]'s technique of region competition does not manage to find the woman's eyes. [16] argue that that the eyes are too small and assimilated to noise. This is clearly not a mistake PSIS has made. The hand image also displays good results, all the better if we consider that the model of image segmentation of PSIS does not explicitly integrate textures. In fig. 4, the left table demonstrates that PSIS obtains again nice results on texture segmentation (tennis and rock images), all the more interesting if we compare them to the approaches of [5, 6], tailor-made for texture segmentation. The right image shows for some images a particular region isolated by PSIS. Remark from lena and squirrel that PSIS is able to isolate regions with high variability (*e.g.* the grass), and obtains results even better than [16] on the squirrel image: their segmentation, although tailor-made for textured images, obtains a segmentation of the grass with many holes, a popular drawback of region-merging techniques [16], see also the result of [2] in fig. 5 (bottom row, region #2) for the grass. Note also the nice segmentation of the truck compared to [2] (fig. 5, bottom row, region #4). The small images in fig. 5 (a-x) display the ability of PSIS to handle reasonable gradients due to lightning and slanted surfaces (images b, f, i, j, k, r, s, u, w, x). This corrects another drawback of many segmentation techniques, relying on the assumption that the image is piecewise constant [2]. Note that in image x, PSIS obtains exactly two regions. The bowl region (x,#2) approximates nicely the true object, in a better way than [2], who find significantly more regions, again with many holes compared to PSIS. In image w, which contains gradients and light effects, PSIS finds exactly three regions, approximating almost perfectly the three parts of the image: the background, the pot, and its lid. The gradients and light effects of image j are much more difficult to handle. In that case, PSIS found four regions (all shown), two of which (#2 and #3) are good approximations of two parts of the object (exterior/interior). More generally, many regions found by PSIS in these small images approximate conceptually distinct parts of the objects: see (a,#4), (b,#3), (c,#4), (e,#3), (f,#3), (n,#4), (s,#3), (t,#3), and many others.

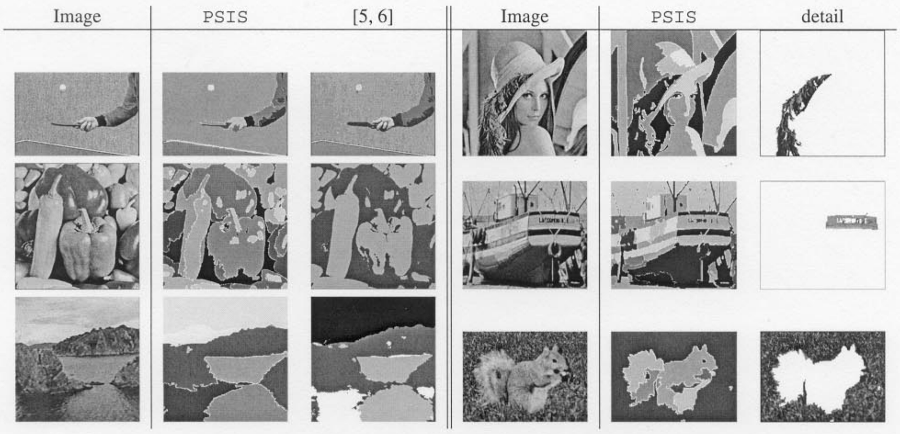

Figure 4. Results obtained by PSIS and [5, 6] on various images. The "detail" column is a region of interest in PSIS's results. PSIS's segmentations are grey-leveled averaged with white borders. Conventions for the results of [5, 6] are various but intuitive.

References

[1] D. Comaniciu and P. Meer. Robust analysis of feature spaces: Color image segmentation. In *Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 750–755, 1997.

[2] P. F. Felzenszwalb and D. P. Huttenlocher. Image segmentation using local variations. In *Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 98–104, 1998.

[3] C. Fiorio and J. Gustedt. Two linear time Union-Find strategies for image processing. *Theoretical Computer Science*, 154:165–181, 1996.

[4] D. Forsyth and J. Ponce. *Computer Vision - A Modern Approach*. Prentice Hall, 2001. Forthcoming.

[5] Z. Kato. *Modélisations Markoviennes Multirésolutions en vision par ordinateur. Application à la segmentation d'images spot.* PhD thesis, Université de Nice-Sophia Antipolis, 1994.

[6] Z. Kato, T. C. Pong, and J. C. M. Lee. Motion compensated color video classification using markov random fields. In *Proceedings of the 3 rd Asian Conference on Computer Vision*, 1998.

[7] M. J. Kearns and Y. Mansour. A Fast, Bottom-up Decision Tree Pruning algorithm with Near-Optimal generalization. In *Proceedings of the 15 th International Conference on Machine Learning*, pages 269–277, 1998.

[8] T. Leung and J. Malik. Contour continuity in region based image segmentation. In *Proceedings of the 5 th Europ. Conf. on Computer Vision*, pages 544–559, 1998.

[9] O. Maron and A.-L. Ratan. Multiple instance learning for natural scene classification. In *Proceedings of the 15 th International Conference on Machine Learning*, pages 341–349, 1998.

[10] C. McDiarmid. Concentration. In M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, and B. Reed, editors, *Probabilistic Methods for Algorithmic Discrete Mathematics*, pages 1–54. Springer Verlag, 1998.

[11] T.-Y. Philips, A. Rosenfeld, and A. C. Sher. O(log n) bimodality analysis. *Pattern Recognition*, pages 741–746, 1989.

[12] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22:888–905, 2000.

[13] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, pages 1134–1142, 1984.

[14] M. Wertheimer. Laws of organization in perceptual forms. In W. B. Ellis, editor, *A Sourcebook of Gestalt Psychology*, pages 71–88. Harcourt, Brace and Company, 1938.

[15] S.-C. Zhu. Embedding gestalt laws in markov random fields. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21:1170–1187, 1999.

[16] S.-C. Zhu and A. Yuille. Region competition: unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18:884–900, 1996.

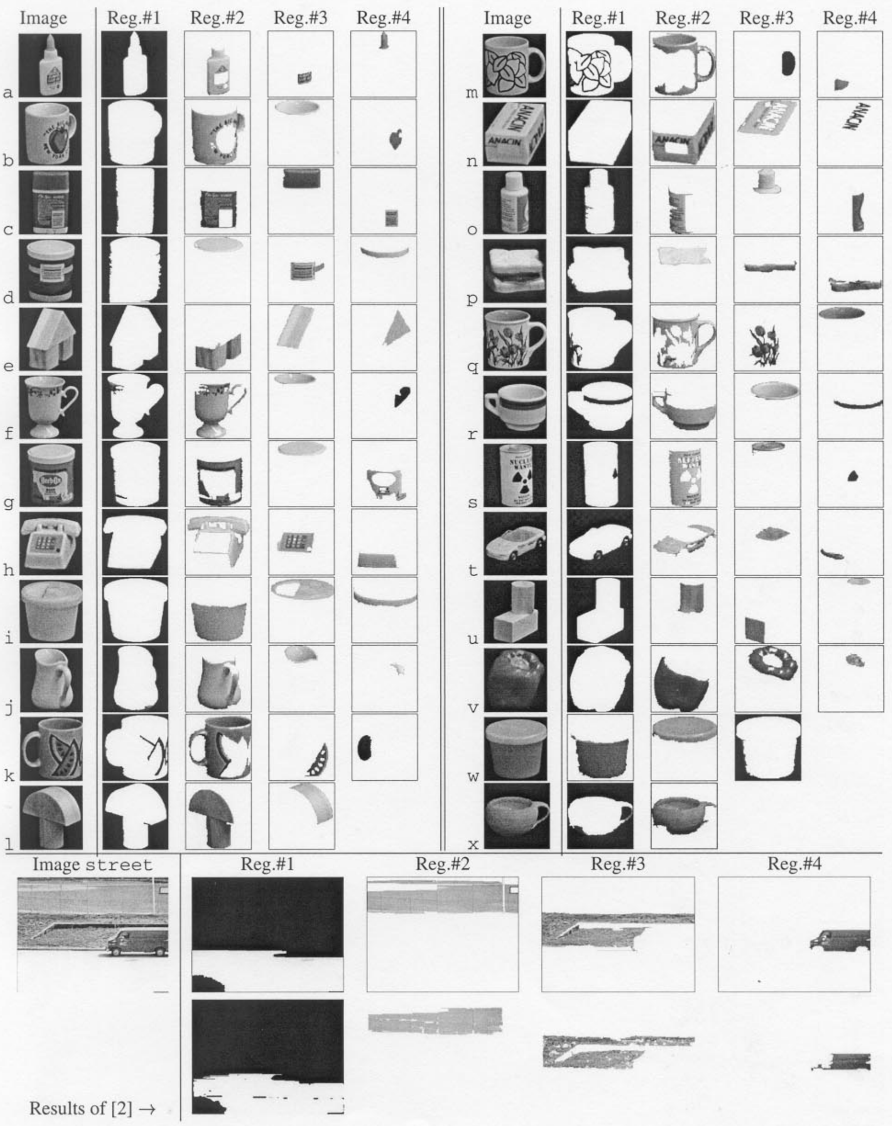

Figure 5. Some images, and some of the most significant regions obtained by PSIS**. The bottom row shows the result of [2] on the** `street` **image. The conventions are the same for all region images (everything that is not the region is white), except for region #1 on the** `street` **results, surrounded by black due to the brightness of the road.**

C.5 [Nock, 2002a]

A FAST, RELIABLE REGION MERGING-LIKE APPROACH HANDLING OCCLUSIONS

Richard Nock

ABSTRACT

This paper brings a novel approach to the problem of segmenting color images having partially occluded objects. It relies on a model of image generation taking into account non connex objects, and removing conventional and restrictive assumptions on distributions. The best segmentation according to this model is approximated in a fast algorithm, having the benefit of simple prototyping and test. More precisely, the algorithm segments 512×512 images in less than a second on conventional PIII computers, using a few kilo-octets C code. Experiments display the quality of the segmentations obtained, in particular on natural images.

1. INTRODUCTION

It is established since the Gestalt movement in psychology that perceptual grouping plays a fundamental role in human perception. So far, the adaptation and automation of the segmentation (and more generally, grouping) task has remained a tantalizing and central problem for image processing. There are four large categories of approaches to image segmentation [2], one of which is of direct interest to us : region growing and merging techniques. In region merging, regions are sets of pixels with homogeneous properties and they are iteratively grown by combining smaller regions or pixels, pixels being elementary regions. Region growing/merging techniques usually work with a statistical test to decide the merging of regions [2]. This test is used in what we call a merging predicate.

As long as the approach is greedy, two essential components participate in defining a region merging algorithm: the merging predicate, and the order followed to test the merging of regions [3]. In a recent paper, the author proposes an efficient algorithm for region merging [1]. This algorithm, called `PSIS`, for `P`robabilistic `S`orted `I`mage `S`egmentation, orders the merging tests in a particular way, and exploits recent results on the concentration of random variables. It is the aim of the present paper to go further into extending the model and the algorithm to a more complex setting, taking into account the fact that some objects shown in an image may actually consist in more than one connex region, because *e.g.* some of them may be partially occluded.

Author's permanent address: Université des Antilles-Guyane, Département Scientifique Interfacultaire, Campus de Schoelcher, B.P. 7209, 97275 Schoelcher, France. Email: `rnock@martinique.univ-ag.fr`. URL:`http://www.martinique.univ-ag.fr/~rnock`.

Extending the approach of [1] consists in two steps: an extension of the underlying model onto which the algorithm is designed, and then an extension of the algorithm itself to cope with this new model. While it gains the benefit of explaining occlusions, the new model keeps the positive aspects of the previous one by [1]: it is general and does not rely on conventional, more or less penalizing, distribution assumptions. Our new segmentation algorithm also keeps the main important features of [1]: its low time/space complexity, and its very fast prototyping and implementation.

After two sections presenting the original model of image generation of [1], and explaining the algorithm `PSIS` of [1], a next section explains the extensions of the model and algorithm. The two last sections present experimentations, and conclude.

2. THE THEORETICAL MODEL

Throughout this paper, the notation $|.|$ stands for cardinal, "$\log$" (resp. "$\ln$") is the base-2 (resp. base-e) logarithm. The observed image, I, contains $|I|$ pixels, each containing three color levels (**R**, **G**, **B**), each of the three belonging to the set $\{1, 2, ..., g\}$. We have deliberately chosen not to use complex formulations of the colors, such as the $L * u * v*$ space [4]. I is an observation of a perfect scene I^* we do not know of, in which pixels are perfectly represented by a family of *distributions*, from which each of the observed color-level is sampled. In I^*, the optimal (or true) regions represent theoretical objects sharing a common homogeneity property:

- all pixels of a given true region have identical expectations for each **RGB** color-level,
- the expectations of adjacent regions are different for at least one **RGB** color level.

I is obtained from I^* by sampling each theoretical pixel for observed **RGB** values. Fig. 1 presents an example of a color-level for one pixel in I^* and how to generate the corresponding observed color-level of the pixel in I. In each pixel of I^*, each color-level is replaced by a set of Q independent random variables (r.v.) taking positive values on domains bounded by g/Q, such that any possible sum of

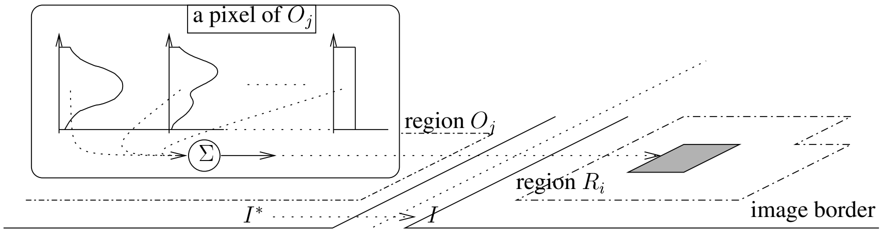

Fig. 1. Generation of a single color-level for one pixel from I^* to I.

outcomes of these Q r.v. with non-zero probability belongs to $\{1, 2, ..., g\}$[1].
Our goal is then straightforward: find observed regions in I approximating as best as possible the true regions in I^*.
Note that in our model, even inside one true object of I^*, the r.v. of the theoretical pixels are not necessarily the same from one pixel to another one, as long as the homogeneity property is satisfied. Therefore, inside a true region, the distributions associated to each pixels can be different from each other. Together with the fact that our model removes conventional distribution assumptions (such as normality), this represents a real freedom in the theoretical image generation. The choice to introduce the parameter Q was also for generality purposes in our theoretical analyzes (instead of making the more conventional choice $Q = 1$). Our experiments, on the other hand, are undergone by picking a single value for Q, and then using it for all experiments.

3. THE GENERAL SEGMENTATION ALGORITHM

We begin by presenting the algorithm `PSIS` of [1]. There are two essential aspects to this algorithm: the merging predicate, and the order to test the merging. Another key feature of the algorithm is that sorting and merging are separated: sorting comes first, and then all merging tests are made to obtain the final segmentation.
The merging predicate comes from the following theorem, which states that, with high probability, the observed average of any region in I shall not deviate too much from its theoretical expectation.

Theorem 1 *([1]) Let R be a region in I. Let $\mathcal{R}_l$ be the set of regions having l pixels in I, and $a \in \{\boldsymbol{R}, \boldsymbol{G}, \boldsymbol{B}\}$. Then the probability that there exists a region in $\mathcal{R}_{|R|}$ such that*

$$\left|\overline{R}_a - \boldsymbol{E}(R)_a\right| \geq g\sqrt{\frac{1}{2Q|R|}\left(\ln\frac{2|\mathcal{R}_{|R|}|}{\delta'}\right)} \qquad (1)$$

is no more than δ'. Here, $\overline{R}_a$ is the observed average for color a of region R in image I, and $\boldsymbol{E}(R)_a$ is the corresponding expectation in I^* for color a of region R.

This theorem says that for any $a \in \{\mathbf{R}, \mathbf{G}, \mathbf{B}\}$, and any region R, there is a concentration between the "true" and the observed average values of color level a in R. This can be used to build a merging predicate $\mathcal{P}(R, R')$ returning whether R and R' can be merged or not:

$$\mathcal{P}(R, R') = \begin{cases} \texttt{true} & \text{iff} \quad \forall a \in \{\mathbf{R}, \mathbf{G}, \mathbf{B}\}, \\ & \quad |\overline{R'}_a - \overline{R}_a| \leq b(R) + b(R') \\ \texttt{false} & \text{otherwise} \end{cases}$$

Here, $b(R) = g\sqrt{(A_I + B_R)/C_R}$, where $A_I = \log(6|I|^2)$, $B_R = g\log(|R| + 1)$ and $C_R = 2Q|R|$ are picked according to [1]. The main idea behind $\mathcal{P}$ is to find a "limit" quantity $l(R, R') = b(R) + b(R')$ such that whenever R and R' come from the same true region in I^*, their colored levels on the observed image I will be limited by $l(R, R')$ with high probability.
Therefore, provided we dispose of a good order for making the merging tests, we shall succeed in merging the subparts of the same true regions altogether, while also limiting the under-merging. Under-merging is the situation in which each true region belongs to exactly one observed region in the segmentation, a segmentation in which some observed regions may contain more than one true region.
This "good" order is approximated by [1] and results in an algorithm which sorts the couples of adjacent pixels (in 4-connexity) as follows. Suppose that the image I contains r_I rows and c_I columns. This represents $K = 2r_Ic_I - r_I - c_I$ couples of adjacent pixels (in 4-connexity). Name S_I as the set of these couples. Denote $f(.)$ the function which takes a couple of pixels (q, q'), and returns the maximum of the three color differences (**R**, **G** and **B**) in absolute value between q and q'. This function is used to sort the K couples of pixels of S_I in increasing values of f, using a quicksort-type procedure (`quicksort`(S_I,f) as shown below in algorithm 1). A last notation used in the algorithm is $R(q)$, denoting current region to which a pixel q belongs in I.

[1] For example, we could suppose that this set of Q r.v. contains for each color level and each pixel 1 r.v. taking values in $\{1, 2, ..., g/Q\}$, and $Q-1$ r.v. taking values in $\{0, 1, ..., g/Q\}$. Provided Q divides g, any outcome for the sum of these Q r.v. falls into the set $\{1, 2, ..., g\}$.

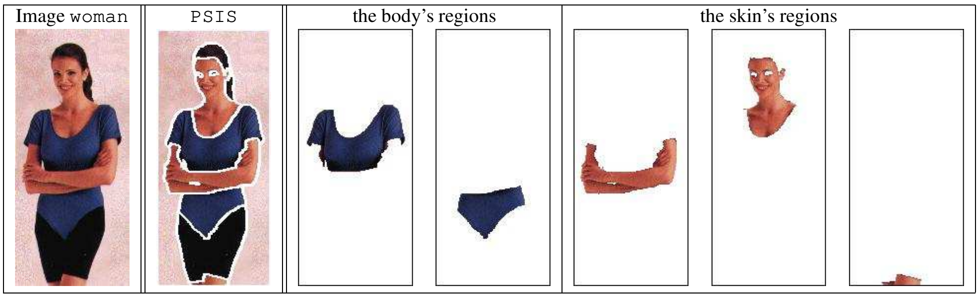

Fig. 2. The segmentation of `PSIS` on the image `woman` (regions are white-bordered), along with some of the largest regions found.

Algorithm 1: `PSIS` (I)

Input: an image I
S'_I=`quicksort`(S_I,f);
for $i = 0$ ***to*** $K - 1$ **do**
  /*(p_i, p'_i) is the i^{th} couple of S'_I*/
  if $R(p_i) \neq R(p'_i)$ ***and*** $\mathcal{P}(R(p_i), R(p'_i))$=`true` **then**
    `Union` ($R(p_i)$,$R(p'_i)$) ;

4. HANDLING OCCLUSIONS

When the theoretical image, I^*, displays an occluded object, the 4-connexity constraint in the regions for `PSIS` makes it impossible to bring all parts of the object into the same region. Figure 2 displays this problem on the image `woman` (here, $Q = 32$). The segmentation obtained has a good quality regarding the perceptually distinct regions on the image, but a human would certainly consider that the two parts of the body (suit) actually come from the same object, and that the three skin regions are actually all the woman's body. Therefore, an even better segmentation of the image would consider these two sets of regions as two objects. The modification of `PSIS` to handle these cases can be obtained by a very slight extension of its model of image generation, extending its homogeneity property. Namely, we replace the former homogeneity property by this one:

- true regions are not necessarily connex regions (in particular, they can be partially occluded),
- all pixels of a given true region have identical expectations for each **RGB** color level,
- the expectations of different true regions are different for at least one **RGB** color level.

When using this property to modify `PSIS`, the main trap to avoid is to replace the couples of adjacent pixels S_I by the set of $\binom{|I|}{2}$ (non-necessarily adjacent) pixels of the initial image (and then run the merging tests). Indeed, the statistical bound of the merging predicate is not strong enough to distinguish for small regions those having similar color values but coming from different true objects. Rather, we have chosen the wiser approach which consists in firstly finding the connex parts of true object (running the original `PSIS`), and then testing the merging between non-adjacent regions found to find true objects possibly occluded. To state this new algorithm, we adopt the following notations. Suppose that the original `PSIS` finds K' regions. Name $\mathcal{S}_I$ as the set of $\binom{K'}{2} = K'(K'-1)/2$ couples of these regions. Denote as f_2 the function which returns, for a couple of regions, the maximum difference (in $\{\mathbf{R}, \mathbf{G}, \mathbf{B}\}$) between their average color levels. Finally, for some region R, let $\mathcal{R}(R)$ denote the (possibly not connex) region to which it belongs. The new algorithm, called $\texttt{PSIS}_{\text{O}}$, is shown in figure 2.

Algorithm 2: $\texttt{PSIS}_{\text{O}}$ (I)

Input: an image I
Run `PSIS`;
$\mathcal{S}'_I$=`quicksort`($\mathcal{S}_I$,f_2);
for $i = 0$ ***to*** $\binom{K'}{2}$ **do**
  /*(R_i, R'_i) is the i^{th} couple of regions of $\mathcal{S}'_I$*/
  if $\mathcal{R}(R_i) \neq \mathcal{R}(R'_i)$ ***and*** $\mathcal{P}(\mathcal{R}(R_i), \mathcal{R}(R'_i))$=`true` **then**
    `Union` ($\mathcal{R}(R_i)$,$\mathcal{R}(R'_i)$) ;

5. EXPERIMENTS

$\texttt{PSIS}_{\text{O}}$ was ran using $Q = 32$ (this is also the value adopted in [1]). Figure 3 shows the result on the `woman` image. In that case, note that the body and the skin are, this time, recognized as single but occluded true objects. Other

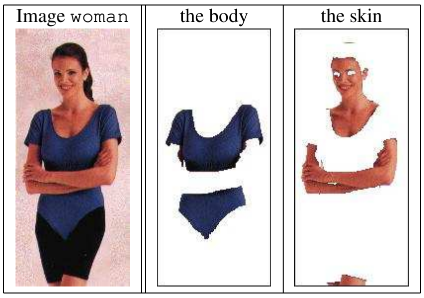

Fig. 3. The two largest regions (right) found by $\mathtt{PSIS}_{\mathrm{O}}$ on the `woman` image (left): this time, the body and the skin are recognized as single, occluded objects.

experiments are shown in figure 4. Note the nice segmentation of the `hand` image. `PSIS` alone finds three regions for the hand, since its connexity is cut by two objects: the ring, and the donut. In the `hand` image, the result of the segmentation of $\mathtt{PSIS}_{\mathrm{O}}$ is also interesting because the algorithm finds exactly three objects, approximating nicely those a human would consider as perceptually distinct: the hand, the donut, and the table.

6. CONCLUSION

The motivation of this paper was to display the strength of the method of [1], in particular to handle the segmentation of images with partially occluded objects. Indeed, a modification of the model of image generation adapting the connexity constraint, along with an adaptation of the algorithm, allows to extend in a natural way the approach of [1] to partially occluded objects, which displays good results on various images. This simple extension deserves however further investigations and adaptations, in particular for complex scenes in which non connex, similar but different objects coexist.

7. REFERENCES

[1] R. Nock, "Fast and Reliable Color Region Merging inspired by Decision Tree Pruning," in *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, 2001, pp. 271–276.

[2] S.-C. Zhu and A. Yuille, "Region competition: unifying snakes, region growing, and bayes/MDL for multiband image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 884–900, 1996.

[3] C. Fiorio and R. Nock, "Sorted region merging to maximize test reliability," in *Procs of the 7th IEEE Int. Conf. on Image Processing*, 2000, pp. 808–811.

[4] D. Comaniciu and P. Meer, "Robust analysis of feature spaces: Color image segmentation," in *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, 1997, pp. 750–755.

[5] P. F. Felzenszwalb and D. P. Huttenlocher, "Image segmentation using local variations," in *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, 1998, pp. 98–104.

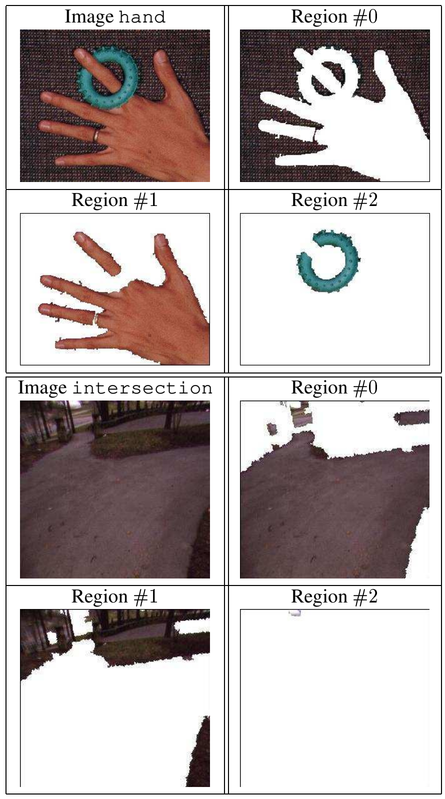

Fig. 4. Results on images `hand` and `intersection`. The largest regions found by $\mathtt{PSIS}_{\mathrm{O}}$ are displayed for each image.