THEORETICAL ADVANCES

# A non-stationary MRF model for image segmentation from a soft boundary map

Max Mignotte

Received: 20 June 2011/Accepted: 19 March 2012 © Springer-Verlag London Limited 2012

**Abstract** In this paper, we address the problem of estimating a segmentation map into regions from a soft (or possibly probabilistic) boundary representation. For this purpose, we have defined a simple non-stationary MRF model with long-range pairwise interactions whose potentials are estimated from the likelihood of the presence of an edge at each considered pair of pixels. Another contribution of this paper is also to demonstrate that an efficient and interesting alternative strategy to complex and elaborate region-based segmentation models consists in averaging several (quickly estimated) soft contour maps (obtained, for example, with simpler edge-based segmentation models) and to use this MRF reconstruction model to achieve a reliable and accurate segmentation map into regions. This model has been successfully applied on the Berkeley image database. The experiments reported in this paper demonstrate that the proposed segmentation model from an edge map is reliable and that this segmentation strategy is efficient (in terms of visual evaluation and quantitative performance measures) and performs well compared with the best existing state-of-the-art segmentation methods recently proposed in the literature.

**Keywords** Color textured image segmentation  $\cdot$  Non stationary Markovian (MRF) model  $\cdot$  Segmentation into regions from edge map  $\cdot$  Segmentation from soft contour  $\cdot$  Energy-based model  $\cdot$  Berkeley image database

e-mail: mignotte@iro.umontreal.ca

#### **1** Introduction

Image segmentation is a frequent pre-processing step whose goal is to simplify the representation of an image into meaningful and spatially coherent regions with similar attributes such as consistent parts of objects or of the background. This low-level vision task, which changes the representation of an image into something that is easier to analyze, is often the preliminary and also crucial step in the development of many image understanding algorithms and computer vision systems.

A review of literature indicates that most of segmentation algorithms can be generally divided into two categories, namely the so-called region-based and edge-based segmentation approaches. Region-based segmentation methods attempt to group spatially coherent regions with similar attributes. They include segmentation methods exploiting the connectivity information between neighboring pixels such as MRF-based statistical [1-3] or graphbased models [4, 5], Mean-Shift-based techniques [6], clustering schemes [7–10], region growing strategies [11, 12], variational schemes [13] or finally region-based split and merge procedures, sometimes directly expressed by a global energy function to be optimized [14]. On the other hand, edge-based segmentation methods rely on the prediction of local edge fragments which are simply defined by significant localized changes or discontinuities in some image features. In this way, classical edge detectors, such as Canny's [15] search for discontinuities in the luminance or color intensity while more recent approaches also uses texture information and rely on a preliminary learning step for the optimal cue combination [16]. Another strategy consists in grouping edges according to perceptual laws [17, 18] and, for example, to combine local edge features with either a morphological operator which uses

M. Mignotte (🖂)

Département d'Informatique et de Recherche Opérationnelle (DIRO), Université de Montréal, Montréal, QC H3C 3J7, Canada

context-dependent structuring elements [18], or with a voting framework that uses deformable tensors [17], to identify long curvilinear structure or perceptual structures in the edge map (a review of these edge-based segmentation methods is given in [19]). Due to the local nature of such approaches and thus, their inherent sensitivity to noise artifacts, boundary detection algorithms inevitably produce false and disconnected contours (excepted in [20]). By this fact, and contrary to the result given by a region-based segmentation method, the resulting soft boundary detection map does not generally exhibit, for a given threshold, a set of closed curves corresponding to the boundaries of a segmentation into regions (each one associated with the consistent parts of objects present in the scene). Consequently, this resulting soft edge map often remains more difficult to exploit in a high-level image analysis system compared with a classical region map.

In this paper, we report an efficient algorithm for estimating a region segmentation map from a soft (or possibly probabilistic) boundary representation. This scheme is based on a non-stationary MRF model expressed by a Gibbs distribution whose the long-range pairwise (second order) interaction potentials are spatially variant and preliminary estimated from the likelihood of the presence of an edge (given by the edge map) at each discrete location. Finally, our segmentation model from an edge potential map emerges as an optimization problem of a complex (non-convex) cost function with several local extrema over the label parameter space. In our application, this final optimization task is performed by a robust multi-resolution coarse-to-fine minimization strategy efficiently relying on a hard constraint enforcing the spatial continuity during the iterative labeling process. A quantitative performance evaluation has been carried out whose interest is twofold. First, it will allow to validate our model and second, it will also show that a good alternative to complex and elaborate region-based segmentation models existing in the literature consists in averaging several (quickly estimated) soft contour maps (obtained for example, with simpler edge-based segmentation models) and to use our segmentation model from edge map to achieve a (more) reliable and accurate segmentation map into regions.

## 2 Proposed model

Let *S* be the final segmentation result into several classes and  $\{l_i\}$  designates the set of class labels associated with this segmentation map. Our generative model of segmentation into different classes  $S = \{l_i\}$  is simply defined by the following Gibbs distribution expressed as

$$P(S = \{l_i\}) = \frac{1}{Z} \exp\left(-\underbrace{\sum_{\langle i,j \rangle} p_{ij} \,\delta(l_i, l_j)}_{U(.)}\right) \tag{1}$$

where Z is the (constant) partition functions,  $\delta$  is the delta Kronecker function and  $\langle i, j \rangle$  is the set of second-order cliques (i.e., binary cliques) of a Markov Random Field (MRF) model defined on a *complete* graph (each node or pixel  $\mathbf{x}_i$  is connected to all other pixels contained within a (square) *search window* of fixed size  $N_s$  pixels centered around  $\mathbf{x}_i$ ) and in which the long-range pairwise interaction potentials  $\{p_{ij}\}$  are spatially variant and preliminarily computed from the soft edge map. In order to favor the same class label for the pixels *i* and *j* when these one are not separated by an edge in the edge map and *vice-versa*, a good and empirical choice for the estimator of  $p_{ij}$  is given by

$$p_{ij} = 1 - \beta \Psi(i,j) \tag{2}$$

where  $\Psi(i, j)$  denotes the maximal contour potential found on the straight line existing between the pixels *i* and *j* in the (soft) edge map plane (normalized between [0 1] and on which a preliminary no maximal suppression step [15] was done),  $\beta$  is an internal parameter of our segmentation model which acts as the inverse of a regularization parameter, i.e., favoring over segmentation (for large value of  $\beta$ ) or, on the contrary, merging regions (for small value of  $\beta$ ). In order to find a particular configuration of  $\hat{S}$ , that efficiently minimizes our energy based model, we use the ICM-based [2] multi-resolution approach proposed in [21] (Fig. 1).



**Fig. 1** From *top to bottom* and *left to right*; a natural image (number 134052) from the Berkeley database, its soft edge map and six pairs of sites represented by six line segments whose the size is proportional to the value of the edge potential  $\Psi(i, j)$  crossed by the line segment. In our two-level multi-resolution model, we consider, in fact (as indicated in Sect. 4.1), a complete graph for the reduced-resolution level and a non-complete graph for the full resolution level by considering that each pixel is connected with its four nearest neighbors and a fixed number of connections (70 in our application), regularly spaced between all other pixels located within a squared *search window* (of fixed size  $N_s = 80$  pixels) centered around the pixel



•  $S_p \leftarrow \text{ICM}$  [regul. parameter  $\beta$  : graph  $\mathcal{G}_{N_s=80,N_c=70}$  : init. by  $S_{\text{init}}$ ] on  $U(\{p_{ij}\})$  with the region hard constraint with parameter  $\xi$  (majority label in each { $\mathcal{R}egion(E)$ })

**REGION LEVEL** 

• ICM [regul. parameter  $\beta$ : init. by  $S_p$ ] on each region at the finest level (consisting in merging the couple of adjacent regions leading to a reduction of the energy function [Eq. (1)] until convergence)

## Algorithm 1

Besides, to further help the optimization process to succeed in finding an optimal solution, a hard constraint enforcing the spatial continuity of each (likely) region is imposed during the iterative ICM labeling process. To this end, the most likely regions in the edge map are easily estimated by identifying the sets of connected pixels whose edge potential is below a given (and low) threshold  $\xi$ . In this way, we easily identify small regions in which few edges or contours or only small gradient magnitudes have been detected. This procedure allows us to define a map of likely homogeneous textural regions. The hard constraint enforcing the spatial continuity of each of the ICM's cluster is simply done by assigning the majority class label in each (pre-estimated and likely homogeneous) region for each iteration of the ICM procedure. In order to get a set of reliable regions, even in the case of the use of soft edge maps exhibiting only very few contours (which are possibly unclosed with one pixel thick) on a uniform black background (i.e., with gray levels equal to zero), we have decided to use only the most reduced resolution level (associated with the soft edge map) to identify the sets of connected pixels whose edge potential is below  $\xi$ . At the next higher resolutions, the set of regions are then interpolated (see [21]). Since the contours are more likely closed at the upper level, this strategy ensures reliable regions and thus a reliable hard constraint.

The minimization is then refined in a second step by identifying each region of the resulting segmentation map (by assigning a different region to each set of spatially connected pixels belonging to a same class). This creates a region adjacency graph which allows to finally perform a merging procedure at region level. To this end, we apply, once again, the ICM relaxation scheme but this time on each region [i.e., by simply merging the couple of adjacent regions leading to a reduction of the energy function of our model (Eq. 1) until convergence]. We provide details of our multi-resolution optimization strategy in Algorithm 1.

# 3 Edge map used in our model

# 3.1 Texture information used

As discriminant texture cues, we have simply used the set of values of the re-quantized color histogram (with equidistant binning) estimated around the pixel to be classified. In our application, this local histogram is equally requantized (for each of the three color channels with  $q_b = 5$ equally spaced bins) in a final  $N_b = q_b^3$  bin descriptor  $(N_b = q_b^3 = 5^3$ , in our application), computed on an overlapping squared fixed-size  $(N_w = 7)$  neighborhood<sup>1</sup>

<sup>&</sup>lt;sup>1</sup> Let us note that  $N_w$  must be large enough to efficiently model the *texton* feature but should also be not too large in order not to affect (too much) the accuracy of the boundary estimation between distinct textured regions. A good compromise, between good classification and contour accuracy, seems to be the value  $N_w = 7$ . Some tests have shown that  $N_w = 5$  and  $N_w = 9$  give slightly similar performance results and a size value greater than  $N_w = 11$  affects the accuracy of the boundary location. This observation was also noticed in [29].

centered around the pixel to be classified. This histogram is then normalized so that it integrates to one (see [8]). In this simpler model, a *texton* (i.e., the fundamental repetitive micro-structures of a textured image) is herein characterized by the values of the re-quantized (local) color histogram (thus encoding a non-parametric mixture of colors). This model is simple to compute, allows significant data reduction while being robust to noise and local image transformations and has already demonstrated efficiency for tracking applications [22].

#### 3.2 Canny-based binary edge map

In order to estimate a reliable binary edge map, which will then be used to estimate a soft boundary representation (see Sect. 3.4), we first rely on a classical Canny's edge map [15] whose gradient magnitude at each pixel (located at row i and column j) is herein replaced by the following simple distance:

$$\mathcal{D}(h_{i-(N_w/2),j},h_{i+(N_w/2),j}) + \mathcal{D}(h_{i,j-(N_w/2)},h_{i,j+(N_w/2)})$$

where *h* is the  $N_b$ -bin vector, i.e., the re-quantized local color histogram, located at row *i* and column *j* (see Sect. 3.1), and  $\mathcal{D}(h_{i-d,j}, h_{i+d,j})$  is the Manhattan distance  $(L_1 \text{ norm})$  between vectors (or bin descriptors)  $h_{i-d,j}$  and  $h_{i+d,j}$  computed on a squared  $N_w$ -size window centered, respectively, at location (i - d, j) and (i + d, j).

We have used a classical no maximal suppression step [15] which makes all edges in one pixel thick. Besides, the thresholding is classically done with hysteresis [15], and the high threshold is estimated as being the value of the Manhattan distance for which its repartition function (computed on the whole image) reaches a certain threshold value  $\tau_h(<1)$ . Finally, the ratio of the high to low threshold is two to one (as suggested in [15]). This procedure allows to make the Canny procedure dependent on only one threshold,  $\tau_h$ , which is closely related to the percentage of edge that will be detected in image ( $\tau_h = 0.86$  in our application).

## 3.3 K-means-based binary edge map

The set of  $q_b^3$ -bin descriptors are also grouped into different clusters (corresponding to each class of the image) by the classical *K*-means algorithm [7] (K = 7 in our application<sup>2</sup>) with the Manhattan distance ( $L_1$  norm). This give us

a K-means-based region map with which we can extract another binary edge map.

#### 3.4 Soft boundary map used

In order to propose a reliable soft boundary map, which will then be used to validate our segmentation model, we have averaged the different Canny-based and K-meansbased binary edge maps obtained for different color spaces of the input images. In our application, ten different color spaces were used, namely RGB, HSV, YIQ, XYZ, LAB, LUV, *I*<sub>1</sub>*I*2*I*3, *H*<sub>1</sub>*H*2*H*3, YC<sub>b</sub> C<sub>r</sub>, TSL [23– 25] (which can be viewed as different image channels provided by various sensors or captors<sup>3</sup>). Finally, we average this resulting soft edge map with the one obtained by multiplying this latter with the soft edge map obtained by computing the color gradient magnitude (normalized between [0 1]). Figure 3 shows an example of a soft boundary map obtained by this strategy for, respectively, one and a combination of ten different color spaces. We can easily see the improvement of considering several color spaces.

#### 3.5 Image complexity-based adaptive regularization

In order to adaptively favoring over segmentation or, on the contrary, merging regions, for a given input image (when this one is available), we have defined our regularization parameter  $\beta$  of the following manner

$$\beta = \beta^{\star} + \tau \tag{3}$$

where  $\tau$  is a weighting factor used to adaptively weight our regularization parameter  $\beta^*$ . This parameter  $\tau$  aims at measuring the complexity of a natural color image or more precisely its complexity in terms of number of different texture types. Our goal is to favor numerous regions in the case of a complex image and inversely. To this end, we have defined this complexity as the measure of the mean absolute deviation (i.e., the Manhattan distance or  $L_1$  norm) of the equally re-quantized color histogram (using  $N_b$ equidistant bins) of each overlapping squared fixed-size

<sup>&</sup>lt;sup>2</sup> We have experimentally noticed that a *K*-means clustering scheme with  $K \approx 7$  classes allows us to obtain a set of resulting segmentation maps with, on average, a number of regions (each region or segment being defined as a set of connected pixels belonging to the same class) approximately equal to 20, agreeing with the average number of regions found in the manual segmentation maps given by the human observers in the Berkeley database [10].

<sup>&</sup>lt;sup>3</sup> The main criterion in the choice of these different color spaces is that they are not linearly related. This allows us to promote diversity and thus encourage different properties in the estimation of the gradient magnitude or in the *K*-means clustering of the data (such as data decorrelation, decoupling effects, perceptually uniform metrics, compaction and invariance to various features, etc. [8, 21]) and consequently obtain slightly different soft and binary edge maps which are then averaged to increase the signal-to-noise ratio of the final soft boundary map used in our application (see Fig. 3 [second row], for an example of denoising result using this strategy).



Fig. 2 Examples of complexity values on some images of the Berkeley database. From *left to right*,  $\tau = 0.235, 0.364, 0.515, 0.661, 0.803$ 

 $(N_w)$  neighborhood contained within this image. This measure ranges in [0 1]. An image with several different texture types will result in value of  $\tau$  close to one. Figure 2 shows several examples of images and their complexity value. This parameter is estimated when the input color image is available; else, in the case where a single edge map is available, this parameter is set to zero (in this case, parameter  $\beta$  is not adaptive and set to a value which is the same for all the set of edge maps).

#### 4 Experimental results

#### 4.1 Setup

In order to decrease the computational load of our multiresolution optimization procedure, we only use two levels of resolution in our pyramidal structure (see Fig. 3): the full resolution and a decimated (i.e., four times smaller) potential contour and a related label field (corresponding to the second upper level of the pyramid structure). We do not consider a complete graph. For the full resolution, we consider that each node  $\mathbf{x}_i$  (or pixel) is connected to its four nearest neighbors (for the set of cliques whose distance is greater or equal to 2) and a fixed number of connections (70 in our application), regularly spaced between all other pixels located within a (squared) search window of fixed size  $N_s$  pixels centered around  $\mathbf{x}_i$  ( $N_s = 80$  in our application). For the reduced-resolution, each node is totally connected with all pixels contained in our reduced search window of fixed size, four times smaller. In our application, a search window of fixed size  $N_s$ , four times smaller than the size of each resolution level, seems to be a reasonable compromise between accuracy of the final reconstruction result (from a soft boundary map) and computational requirements.

We have decided to initialize the lower (or second upper) level of the pyramid with a sequence of 15 different

random segmentations<sup>4</sup> (with *K* classes). The full resolution level is then initialized with the duplication of the best segmentation result in the model sense (i.e., the one associated with the lowest Gibbs energy U) obtained after convergence of the ICM at this lower resolution level (see Fig. 3).

4.2 Comparison with state-of-the-art methods

In these experiments, we have to test our segmentation algorithm on the Berkeley segmentation database [26] consisting of 300 color images of size  $481 \times 321$  (divided into a training and test sets of, respectively, 200 and 100 images). For each color image, a set of benchmark segmentation results, provided by human observers, is available and will be used to quantify the reliability of the proposed segmentation algorithm. In order to ensure the integrity of the evaluation, the internal parameters of the algorithm are tuned on the train image set. The algorithm is then bench-marked using the optimal training parameters on the independent test set.

We have compared our segmentation algorithm (called SFSBM<sub> $[\beta^{\star}|\xi]$ </sub> for Segmentation From a Soft Boundary Map,  $\beta^{\star}$  and  $\xi$  being its two internal parameters) against several unsupervised algorithms. For each of these algorithms, the internal parameters are set to optimal values

<sup>&</sup>lt;sup>4</sup> It is worth recalling that the solution obtained by our two-level multi-resolution optimization procedure depends closely on the estimation obtained on the coarser resolution level (which, after interpolation, will guide the solution of the full resolution level). Furthermore, due to the ICM-based deterministic local optimization used at coarser level, this latter coarse solution will depend on the random initialization. Experimentally (after some tests) we have noticed that a sequence of 15 different random initializations allowed us to find, at least, one initialization leading to a reliable (coarse) solution in all cases tested.

**Fig. 3** From *top to bottom* and *left to right*; A natural image from the Berkeley database ( $n^0$  134008), its soft boundary map (one and ten color space) and the formation of its region process (algorithm

SFSBM<sub>[ $\beta$ \*=2.3 |  $\xi$ =30]) at the (l = 2nd) upper level of the pyramidal structure at iteration [0-2], 3 (the last iteration) of the ICM optimization algorithm. Duplication and result of the ICM relaxation scheme at the finest level of the pyramid at iteration 0, 1 and 7 (last iteration) and final ICM segmentation result at region level</sub>



and/or correspond to the internal values suggested by the authors. All color images are normalized to have the longest side equals to 320 pixels. The segmentation results are then supersampled to obtain segmentation images with the original resolution ( $481 \times 321$ ) before the estimation of the performance metric.

The comparison is based on the PRI performance measure [27] which seems to be also highly correlated with human hand-segmentations [10] (a score equal to PRI = 0.78, for example, simply means that, on average, 78 % of pairs of pixel labels are correctly classified in the segmentation results).

$\begin{array}{llllllllllllllllllllllllllllllllllll$	
Its internal parameters on the Berkeley image database [26]SFSBM $_{[\beta^*=2.30]\xi=30]}$ 0.787 [train $\triangleright$ 0.790 test $\triangleright$ 0.781] $\approx 60 \text{ s} (320 \times 215)$ SFSBM $_{[\beta^*=2.30]\xi=0]}$ 0.776 [train $\triangleright$ 0.778 test $\triangleright$ 0.772] $\approx 60 \text{ s} (320 \times 215)$ SFSBM $_{[\beta^*=3.00]\xi=30]}$ 0.785 [train $\triangleright$ 0.786 test $\triangleright$ 0.781] $\approx 60 \text{ s} (320 \times 215)$ SFSBM $_{[\beta^*=1.80]\xi=30]}$ 0.772 [train $\triangleright$ 0.775 test $\triangleright$ 0.764] $\approx 60 \text{ s} (320 \times 215)$ SFSBM $_{[\beta^*=1.80]\xi=30]}$ 0.772 [train $\triangleright$ 0.775 test $\triangleright$ 0.764] $\approx 60 \text{ s} (320 \times 215)$ CTex [9]0.800 $\approx 85 \text{ s} [184 \times 184] [9]$ PRIF [21]0.800 $\approx 80 \text{ s} (320 \times 215)$ MIS $_{[\lambda} = 50]$ [28]0.798 $\approx 2.9 \text{ s} (320 \times 215)$ SCKM $_{[\kappa=9]\xi=0.37]}$ [29]0796 $\approx 20 \text{ s} (320 \times 215)$ FCR $_{[\kappa_1=13]\kappa_2=6[\kappa=0.135]}$ [8]0.788 $\approx 60 \text{ s} (320 \times 215)$	
$\begin{array}{llllllllllllllllllllllllllllllllllll$	
$\begin{array}{llllllllllllllllllllllllllllllllllll$	
$\begin{array}{llllllllllllllllllllllllllllllllllll$	
CTex [9]0.800 $\approx 85 \text{ s} [184 \times 184] [9]$ PRIF [21]0.800 $\approx 80 \text{ s} (320 \times 215)$ MIS <sub>[<math>\lambda = 50</math>]</sub> [28]0.798 $\approx 2.9 \text{ s} (320 \times 215) [28]$ SCKM <sub>[<math>\kappa = 9 \xi = 0.37] [29]0796<math>\approx 20 \text{ s} (320 \times 215)</math>FCR<sub>[<math>\kappa_1 = 13 \kappa_2 = 6 \kappa = 0.135] [8]0.788<math>\approx 60 \text{ s} (320 \times 215)</math></math></sub></math></sub>	
PRIF [21]0.800 $\approx 80 \text{ s} (320 \times 215)$ $MIS_{[\lambda = 50]}$ [28]0.798 $\approx 2.9 \text{ s} (320 \times 215)$ [28] $SCKM_{[\kappa=9]\xi=0.37]}$ [29]0796 $\approx 20 \text{ s} (320 \times 215)$ $FCR_{[\kappa_1=13]\kappa_2=6]\kappa=0.135]}$ [8]0.788 $\approx 60 \text{ s} (320 \times 215)$	
$ \begin{array}{llllllllllllllllllllllllllllllllllll$	
SCKM [ $\kappa=9 \xi=0.37]$ [29]0796 $\approx 20 \text{ s} (320 \times 215)$ FCR [ $\kappa_1=13 \kappa_2=6 \kappa=0.135]$ 0.788 $\approx 60 \text{ s} (320 \times 215)$	]
FCR <sub>[<math>\kappa_1 = 13</math>]<math>\kappa_2 = 6</math>[<math>\kappa = 0.135</math>][8] 0.788 <math>\approx 60 \text{ s} (320 \times 215)</math></sub>	
MD2S [ $\kappa=11 \xi=0.4$ ] [30] 0.784 $\approx 60 \text{ s} (320 \times 215)$	
FH [5] in [5] $0.784 \approx 1 \text{ s} (320 \times 215)$ [5]	
HMC [31] 0.783 $\approx 80 \text{ s} (320 \times 215) [31]$	
JSEG [11] (in [9]) 0.770 $\approx 6 \text{ s} (184 \times 184)$ [11]	
The first value is the $CTM_{\eta=0.15}$ [10, 32] 0.762 $\approx 180 \text{ s} (320 \times 215)$ [32	2]
entire database and values $St-SVGMM_{[\kappa=15]}[33]$ 0.759 N/A	
between square brackets Mean-Shift [6] (in [10]) $0.755$ $\approx 20 \text{ s} (320 \times 215)$	
correspond to performances on the train and test image sets NCuts [4] (in [10]) $0.722$ $\approx 120 \text{ s} (100 \times 120)$ [4]	

# 4.3 Results and discussion

Table 1 shows the obtained results for different values of  $\beta^{\star}$  and  $\xi$ . A comparison with the best existing state-of-theart segmentation algorithms, in terms of PRI<sup>5</sup> score and computing time is also given in this Table. In terms of PRI measure, we observe that the discussed fusion strategy gives competitive results over the set of images of the Berkeley image database. We can notice that a good value of  $\xi$  allows to significantly improve the segmentation result. Consequently, it is really necessary to help the optimization process to succeed in finding an optimal solution. To this end, the hard constraint enforcing the spatial continuity is useful in this very difficult energybased segmentation method. For the four different sets of soft boundary maps used in our tests (i.e., for our SFSBM algorithm, see Table 1) and for the three different sets of contour maps given in Table 2, we have noticed that  $\xi$ should be ideally within the range  $\xi \in [0, ..., 30]$ .  $\xi = 0$ means that the hard constraint (enforcing the spatial continuity) is not considered and consequently, in this case, the segmentation result is often over-segmented. On the other hand, a high value of  $\xi$  induces a useless under-segmentation (occurring when real regions are erroneously fused). In order to find the optimal value of  $\xi$ , a strategy consists in **Table 2** PRI score obtained on several sets of soft boundary map estimated by the boundary detection algorithms [35–37] (test set of the Berkeley database)

PRI
0.8754
test set ⊳ 0.79415
test set ⊳ 0.77082
test set ⊳ 0.77711

gradually increasing this parameter until the set of segmentation results are correctly segmented in terms of average number of regions, a priori defined as acceptable (for a given application).

We recall that, to decrease the computational load of our segmentation procedure, these segmentation results are obtained by not considering a *complete* graph in the second upper level of our pyramidal structure; each node is, in fact, totally connected with all pixels contained in a reduced squared search window of size  $N_s = 80$ . By considering a *complete* graph, i.e., a *search window* of size  $N_s = 160$  on this upper coarser level (allowing all the pairwise connections to be taken into account for an input image on the second upper level of our pyramidal structure), the PRI score is essentially the same, i.e., PRI = 0.785, for a computational time approximately ten times higher. It is also worth pointing out that our multi-resolution procedure is mainly used to decrease the complexity and the computational load of our optimization/segmentation procedure (because most of the labeling procedure is correctly

 $<sup>\</sup>frac{1}{5}$  We have used the Matlab source code, proposed by Yang to estimate the PRI measure presented in the following section. This code is kindly available on-line at http://www.eecs.berkeley. edu/~yang/software/lossy\_segmentation/.



Fig. 4 Distribution of the PRI performance measure over the 300 images of the Berkeley database (for SFSBM<sub>[ $\beta^*=2,3|\zeta=30|$ </sub>)

estimated at this resolution level [see Fig. 3]). Consequently, two levels of resolution allow us to decrease the computing time of our segmentation procedure by approximately a factor 4. Nevertheless, and it deserves to be mentioned, a lower resolution level of our multi-resolution structure should not be too coarse, to estimate a reliable soft boundary map which will be able to exhibit the consistent parts of small objects that we want to detect (and thus to get a reliable estimation of the pairwise potentials  $p_{ii}$  computed over this decimated soft edge map). Consequently, more levels of resolution will not improve the result since the boundary information of small objects could unexpectedly disappear with a too coarse level of resolution. In other words, the more we use a coarse level of the pyramid, the more we select (a priori) only the large consistent parts of objects to be segmented.

Figure 4 shows the distribution of the PRI measure over the 300 images of the Berkeley image database (for the algorithm SFSBM<sub>[ $\beta^{\star}=2.25 | \xi=30]$ </sub>). Figure 5 displays some examples of segmentations obtained by our algorithm.

We have also shown, in Fig. 6, the four worst segmentations (in the PRI sense), obtained by our segmentation method. Experiments show that our method tends to oversegment some images containing regions with large texture elements or with spatially variant and/or progressively variant and/or illumination variant textures or sometimes to merge a part of an animal (with its texture camouflage) with its natural environment (i.e., a background with an almost identical texture). In these latter cases, this oversegmentations are directly due to the small size of the  $N_w$ squared window (used to compute the Canny-based and Kmeans based binary edge maps) which, in fact, sets the size of the texture elements which are then segmented. Nevertheless, it is worth pointing out that a larger  $N_w$  window size does not improve the average PRI score because it **Fig. 5** Example of segmentations obtained by our algorithm **SFSBM**<sub> $[\beta^{\star}=2.3|\xi=30]$ </sub> (and the soft edge map given in Sect. 3) on several images of the Berkeley image database (see also Table 1 for quantitative performance measures and at our website for the segmentation results on the entire database)

induces a loss of accuracy on the detected boundaries between each textured region. There is thus a compromise, via the value of  $N_w$ , between good classification (of possibly large texture elements) and contour accuracy between the segmented regions. It seems that this is the main drawback of our method used to compute our soft boundary map. Nevertheless, this drawback does not call into question our MRF reconstruction model for which the average PRI score will be all the more competitive than the soft boundary map will be correctly approximated (see Table 2).

The results for the entire database and the list of PRI scores obtained for each segmented image by our algorithm are available on-line.

We have also validated our algorithm on a set of soft boundary maps (all of whose edges are one-pixel-thick but whose boundary map does not exhibit, for a given threshold, a set of closed curves), estimated on the test image base of the Berkeley image database and publicly available at http address [34]. More precisely, we have tested our algorithm on the soft boundary maps estimated by the contour detection algorithms proposed by [35–37]. For each one, we have selected the parameter vector [ $\beta^{\star} \xi$ ] which ensures the better PRI score. Table 2 displays some examples of performance measures obtained by our algorithm.

#### 4.4 Algorithm

The segmentation procedure takes, on average (per image), <30 s (for an i7-930 Intel CPU, 2.8 GHz, 5611 bogomips and non-optimized C++ code running on Linux) for the estimation of our soft edge map and <30 s for the segmentation procedure from this soft edge map (for a  $320 \times 214$  image). Source code (in C++ language) of our algorithm (with the set of segmented images) is publicly available at our website to make possible eventual comparisons with future segmentation algorithms or different performance measures.

#### 5 Conclusion

In this paper, we have presented a new and efficient segmentation procedure from a soft (or possibly probabilistic) boundary representation. This procedure relies on a





**Fig. 6** The four Berkeley images associated with the four worst PRI scores obtained by our segmentation method SFSBM<sub> $[\beta^{\star}=2.30|\xi=30]$ </sub>. From *left to right*, image number 15088 (PRI = 0.394), 130034 (PRI = 0.392), 210088 (PRI = 0.347), 86068 (PRI = 0.283)

non-stationary MRF model with long-range pairwise interactions whose potentials are estimated from the likelihood of the presence of an edge at each considered pair of pixels along with a multi-resolution optimization procedure. A contribution of this paper is also to show that an interesting alternative strategy to region-based segmentation models consists in estimating or averaging several (possibly quickly estimated) soft contour maps and to exploit our segmentation model to finally achieve a reliable segmentation map into regions.

Acknowledgments The author would like to thank the anonymous reviewers for their many valuable comments and suggestions that helped to improve both the technical content and the presentation quality of this paper.

#### References

- Geman S, Geman D (1984) Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Trans Pattern Anal Mach Intell 6(6):721–741
- Besag J (1986) On the statistical analysis of dirty pictures. J R Stat Soc B-48:259–302
- Destrempes F, Mignotte M, Angers J-F (2005) A stochastic method for Bayesian estimation of hidden Markov models with application to a color model. IEEE Trans Image Process 14(8):1096–1108
- 4. Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22:888–905
- 5. Felzenszwalb P, Huttenlocher D (2004) Efficient graph-based image segmentation. Int J Comput Vis 59:167–181
- Comaniciu D, Meer P (2002) Mean shift: a robust approach toward feature space analysis. IEEE Trans Pattern Anal Mach Intell 24(5):603–619
- Lloyd SP (1982) Least squares quantization in PCM. IEEE Trans Inform Theory 28(2):129–136
- Mignotte M (2008) Segmentation by fusion of histogram-based K-means clusters in different color spaces. IEEE Trans Image Process 17:780–787
- Ilea DE, Whelan PF (2008) Ctex- an adaptive unsupervised segmentation algorithm on color-texture coherence. IEEE Trans Image Process 17(10):1926–1939
- Yang AY, Wright J, Sastry S, Ma Y (May 2008) Unsupervised segmentation of natural images via lossy data compression. Comput Vis Image Underst 110(2):212–225

- Deng Y, Manjunath BS (2001) Unsupervised segmentation of color-texture regions in images and video. IEEE Trans Pattern Anal Mach Intell 23(8):800–810
- Dawoud A, Netchaev A (2011) Preserving objects in Markov random fields region growing image segmentation. Pattern Anal Appl, pp 1–7
- Salah MB, Mitiche A, Ayed IB (2010) Effective level set image segmentation with a kernel induced data term. IEEE Trans Image Process 19(1):220–232
- Zhu S, Yuille A (1996) Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. IEEE Trans Pattern Anal Mach Intell 18:884–900
- 15. Canny J (1986) A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 8(6):679–698
- Martin D, Fowlkes C, Malik J (May 2004) Learning to detect natural image boundaries using local brightness, color and texture cues. IEEE Trans Pattern Anal Mach Intell 26:530–549
- Guy G, Medioni G (1996) Inferring global perceptual contours from local features. Int J Comput Vis 20(1–2):113–133
- Papari G, Petkov N (2008) Adaptive pseudo-dilatation for geistalt edge grouping and contour detection. IEEE Trans Image Process 17(10):1950–1962
- Papari G, Petkov N (2011) Edge and line oriented contour detection: state of the art. Image Vis Comput 29(2–3):79–103
- Arbelaez P (2006) Boundary extraction in natural images using ultrametric contour maps. In: 5th IEEE workshop on perceptual organization in computer vision (POCV'06), New York, USA
- Mignotte M (2010) A label field fusion bayesian model and its penalized maximum rand estimator for image segmentation. IEEE Trans Image Process 19(6):1610–1624
- Pérez P, Hue C, Vermaak J, Gangnet M (2002) Color-based probabilistic tracking. In: Eur conf on computer vision, ECCV'2002. LNCS 2350. Copenhaguen, Denmark, pp 661–675
- Martinkauppi JB, Soriano MN, Laaksonen MH (2001) Behavior of skin color under varying illumination seen by different cameras at different color spaces. In: Proc SPIE, machine vision applications in industrial inspection IX. San Jose, California, pp 102–113
- 24. Banks S (1990) Signal processing, image processing and pattern recognition. Prentice Hall, Englewood Cliffs
- Braquelaire J-P, Brun L (1997) Comparison and optimization of methods of color image quantization. IEEE Trans Image Process 6:1048–1952
- 26. Martin D, Fowlkes C, Tal D, Malik J (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc 8th int'l conf computer vision, vol 2, July 2001, pp 416–423
- 27. Unnikrishnan R, Pantofaru C, Hebert M (2005) A measure for objective evaluation of image segmentation algorithms. In: Proc

of the 2005 IEEE conference on computer vision and pattern recognition (CVPR '05). Workshop on empirical evaluation methods in computer vision, vol 3, June 2005, pp 34–41

- Krinidis M, Pitas I (2009) Color texture segmentation based on the modal energy of deformable surfaces. IEEE Trans Image Process 7(18):1613–1622
- Mignotte M (January 2011) A de-texturing and spatially constrained K-means approach for image segmentation. Pattern Recognit Lett 32(2):359–367
- Mignotte M (March 2011) MDS-based multi-resolution nonlinear dimensionality reduction model for color image segmentation. IEEE Trans Neural Netw 22(3):447–460
- Hedjam R, Mignotte M (2009) A hierarchical graph-based Markovian clustering approach for the unsupervised segmentation of textured color images. In: Proc of the IEEE international conference on image processing (submitted)
- 32. Ma Y, Derksen H, Hong W, Wright J (2007) Segmentation of multivariate mixed data via lossy coding and compression. IEEE Trans Pattern Anal Mach Intell 29(9):1546–1562

- 33. Sfikas G, Nikou C, Galatsanos N (2008) Edge preserving spatially varying mixtures for image segmentation. In: Proc of the IEEE int conf comput vis pattern recognit, vol 1. Anchorage, AK (USA), June 2008, pp 1–7
- 34. Martin D, Fowlkes C The Berkeley segmentation database and benchmark. image database and source code publicly. http:// www.cs.berkeley.edu/projects/vision/grouping/segbench/
- Ren X (2008) Multi-scale improves boundary detection in natural images. In: Proc of the 10th European conference on computer vision (ECCV'08), 2008
- Dollar P, Tu Z, Belongie S (2006) Supervised learning of edges and object boundaries. In: Proc of the computer vision and pattern recognition (CVPR'06), 2006
- Maire M, Arbelaez P, Fowlkes C, Malik J (2008) Using contours to detect and localize junctions in natural images. In: Proc of the 2008 IEEE conference on computer vision and pattern recognition (CVPR '08), 2008