

# Bayesian Oil Spill Segmentation of SAR Images via Graph Cuts<sup>1</sup>

Sónia Pelizzari and José M. Bioucas-Dias

*Instituto de Telecomunicações, I.S.T., TULisbon, Lisboa, Portugal*  
*soniap@lx.it.pt, bioucas@lx.it.pt*

**Abstract.** This paper extends and generalizes the Bayesian semi-supervised segmentation algorithm [1] for oil spill detection using SAR images. In the base algorithm on which we build on, the data term is modeled by a finite mixture of Gamma distributions. The prior is an M-level logistic Markov Random Field enforcing local continuity in a statistical sense. The methodology proposed in [1] assumes two classes and known smoothness parameter. The present work removes these restrictions. The smoothness parameter controlling the degree of homogeneity imposed on the scene is automatically estimated and the number of used classes is optional. Semi-automatic estimation of the class parameters is also implemented. The maximum a posteriori (MAP) segmentation is efficiently computed via the  $\alpha$ -expansion algorithm [2], a recent graph-cut technique. The effectiveness of the proposed approach is illustrated with simulated (Gaussian or Gamma data term and M-level logistic classes) and real ERS data.

## 1 Introduction

Segmentation of dark patches in SAR images is an important step in any oil spill detection system and many different approaches to the problem have been proposed so far. These approaches are built on off-the-shelf segmentation algorithms such as ‘Adaptive Image Thresholding’, ‘Hysteresis Thresholding’, ‘Edge Detection’ (see [3] and references therein), and entropy based methods like the ‘Maximum Descriptive Length’ technique [4].

Work [1] introduces a Bayesian segmentation algorithm where the observed data (oil and water) data is modeled by a finite Gamma mixture, with a given predefined number of components. To estimate the parameters of the class conditional densities, an expectation maximization (EM) algorithm was developed. The used prior is a second order Markov Random Field (MRF), more specifically an isotropic Ising Model. To estimate the labels, the posterior distribution is maximized (MAP) via graph-cut techniques [5].

Notwithstanding the promising results provided by the above described segmentation method, it has restrictions that the present work overcomes. The first restriction

---

<sup>1</sup> The work was supported in part by the Portuguese “Fundação para a Ciência e Tecnologia” (FCT) under the grant PDCTE/CPS/49967/2003 and by the European Space Agency (ESA) under the grant ESA/C1:2422.

concerns the number of classes that is limited to two. The second restriction concerns the smoothness parameter that has to be manually tuned. Furthermore, the class parameters estimation process is completely supervised, requiring an interaction with the user in order to manually select a region containing oil pixels and a region containing water pixels.

In the present work we generalize [1] by: (1) extending the number of segmented classes to a predefined optional number  $c$ , (2) automatically estimating the homogeneity parameter  $\beta$  in the MRF, and (3) automatically estimating the class parameters.

To extend [1] to an optional number of classes, the so-called  $\alpha$ -expansion algorithm [2] is implemented. In order to estimate the smoothness parameter, two different techniques are tested, namely the Least Squares (LS) Fit and the Coding Method (CD) [6]. A first attempt is carried out to implement unsupervised segmentation using a semi-supervised initialization.

To evaluate the accuracy of the algorithm, different simulations are carried out. The simulations address both the Gamma and the Gaussian data model. For the real images, the Gamma mixture data model proposed in [1] is adopted to model the observed SAR intensity values.

The article is organized as follows: Section 2 gives a short overview of the original algorithm that builds the base to this work; Section 3 describes, in pseudo-code, the main steps of the proposed segmentation methods; Section 4 presents simulation and real results, and finally Section 5 contains concluding and future work remarks.

## 2 Overview of Base Algorithm

The algorithm proposed in [1] addresses the problem of finding an estimation  $\hat{f}$  of a labeling for a set of  $N$  pixels  $P := \{1, 2, \dots, N\}$ . When  $c$  possible classes are available, a labeling  $f := \{f_1, f_2, \dots, f_N\}$  is a mapping from  $P$  to  $L$ , where  $L := \{l_1, l_2, \dots, l_c\}$  is the set of discrete values that the pixels may take. The vector  $y := \{y_1, y_2, \dots, y_N\}$  stands for the observed data, corresponding to the image intensity measurements at the pixels.

In order to infer  $\hat{f}$ , we adopt the MAP criterion. This amounts to maximize the posterior density of the labeling given the observed data. As described in [1] in detail, this is equivalent to minimizing the objective function

$$E(f_1, \dots, f_N) = \sum_{p=1}^N E^p(f_p) + \sum_{p < j} E^{p,j}(f_p, f_j), \quad (1)$$

where  $p, j \in P$  are pixel locations,  $E^p$  is the negative likelihood given by

$$E^p(f_p) = -\log(p(y_p | f_p)), \quad (2)$$

where  $p(y_p | f_p)$  is the conditional density of  $y_p$  given  $f_p$ , called data model or sensor function, and  $E^{p,j}$  is the prior clique potential associated with the the clique  $\{p, j\}$

containing the pair of neighboring pixels  $p$  and  $j$  [6]. Since we have adopted an MLL, we have

$$E^{p,j}(f_p, f_j) = -\beta \delta(f_p, f_j), \quad (3)$$

where  $\delta$  is the discrete delta function and  $\beta$  controls the degree of homogeneity we wish to impose on the scene. Note that

$$\sum_{p < j} E^{p,j}(f_p, f_j) = -\beta \text{nrNeighbours}(f), \quad (4)$$

with

$$\text{nrNeighbours}(f) = \sum_{i=1}^N n(f_i), \quad (5)$$

where  $n(f_i)$  is the number of neighbors in neighborhood  $N_i$  having the same label as pixel  $i$ .

As demonstrated in [1],  $E(f_1, \dots, f_N)$  is graph representable for  $c = 2$  and in these circumstances, the global minimum of the objective function may be computed by applying the graph-cut algorithm described in [5].

### 3 Proposed Segmentation Methods

In the next Sections we propose supervised and unsupervised approaches to the segmentation. The first approach assumes known class parameters, whereas the second does not. In both methods, the smoothness parameter is assumed unknown.

#### 3.1 Supervised Segmentation with Beta Unknown

In the first segmentation method, we adopt iterative labeling-estimation, with the two steps being performed alternately, inspired by the EM algorithm [6]. The initial values for the labeling and the parameter estimator are optional and don't seem to have a relevant influence on the final performance. Since the class parameters are assumed known, they are omitted from the pseudo-code.

Algorithm-1:

1. Start with an arbitrary initial labeling  $f_0$  and arbitrary parameter  $\hat{\beta} = \beta_0$
2. While  $|\Delta\hat{\beta}| \leq \delta$  or nrIterations < ItMaxNr do
  - 2.1 Find  $\hat{f} = \alpha\_Expansion(f_0, \hat{\beta})$
  - 2.2 Find  $\hat{\beta} = LS\_Estimation(\hat{f})$  or CodingMethod( $\hat{f}$ )
3. Return ( $\hat{f}, \hat{\beta}$ )

### 3.2 Unsupervised Segmentation with Semi-Supervised Initialization

In this second method, we have also adopted iterative labeling-estimation as in ‘Algorithm-1’, but now the class parameters are also iteratively estimated. The initialization of the class parameters is performed in a semi-automatic way: the user provides a region of pixels corresponding to one (for example the most frequent) of the classes (class1). This region is then used to estimate the ML (Maximum Likelihood) parameters of the class1 distribution. In a second step, pixels are clustered in two sets, class1 and not-class1, by applying a simple threshold to the estimated distribution. Then, the parameters of the remaining classes are initialized by applying an EM mixture estimation procedure to the pixels clustered in the set not-class1.

Algorithm-2:

1. Start with an arbitrary parameter  $\hat{\beta}=\beta_0$   
and arbitrary initial labeling  $f_0$
2. Provide initial class parameter estimations  $\hat{\theta}=\theta_0$
3. Provide initial  $\hat{f} = \alpha\_Expansion(f_0, \beta_0, \theta_0)$
4. While  $|\Delta\hat{\beta}| \leq \delta$  or nrIterations < ItMaxNr do
  - 4.1 Find  $\hat{\theta} = ML\_Estimation(\hat{f})$
  - 4.2 Find  $\hat{f} = \alpha\_Expansion(f_0, \hat{\beta}, \hat{\theta})$
  - 4.3 Find  $\hat{\beta} = LS\_Estimation(\hat{f}, \hat{\theta})$  or CodingMethod( $\hat{f}, \hat{\theta}$ )
5. Return ( $\hat{f}, \hat{\beta}, \hat{\theta}$ )

## 4 Results: Simulated and Real Images

This section presents results for simulated and for real SAR images. In the first case, different test scenarios are provided, corresponding to Gaussian and Gamma data terms. Although simulations have been restricted to one Gamma mode per class, the developed procedure also works with Gamma mixtures as developed in [1].

### 4.1 Simulated Images

Three different test scenarios have been adopted:

Scenario 1: the simulated image contains three classes generated by an MLL Markov-Gibs distribution corrupted with Gaussian noise. Segmentation is performed applying “Algorithm-1”, described in Section 3.1. The parameter estimation is performed using the LS method and the class parameters are known (same values as used for the simulation). The test is performed for five different images, corresponding to an increasing difficulty grade of segmentation. Each test is run three times and the mean

values of the overall accuracies (OA) corresponding the percentage of correct label are computed. For comparison, the estimation of  $\beta$  is performed in a supervised way, applying the LS method to the ‘ground-truth’ image and running the  $\alpha$ -expansion algorithm once with the estimated  $\beta$ .

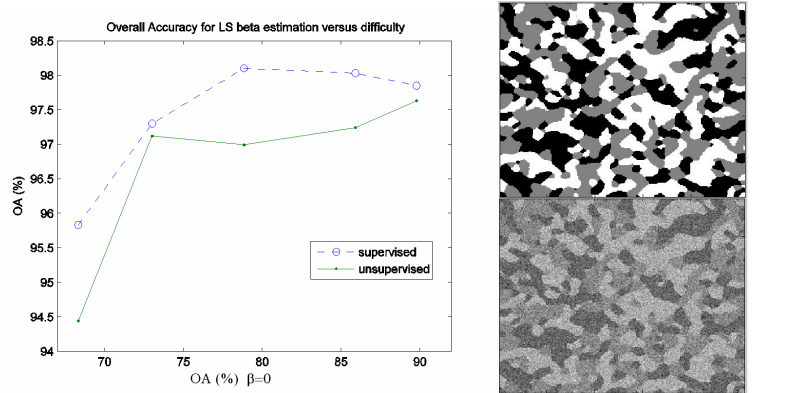
**Scenario 2:** here a simulated image of three classes corrupted by Gamma noise is used. The ground-truth is ‘hand-made’ and contains structures resembling those that may be found in oil-spill scenarios. The same algorithm as in ‘Scenario 1’ is used, both with the LS and the CD estimation methods. The unsupervised segmentation is compared with the results given by the best achievable segmentation using  $\alpha$ -expansion, corresponding to tuning the  $\beta$  parameter manually.

**Scenario 3:** here, for the same simulated image used in scenario 1, the class parameters estimation is also incorporated in the algorithm, by applying ‘Algorithm-2’, described in Section 3.2. Initial class parameters estimation is provided by performing a one-class supervised estimation based on one-class clustering.

**Scenario 1:** To assess the segmentation performance, we compare the OA with that obtained without the MRF prior, *i.e.*,  $\beta=0$ . For Gaussian classes with equal standard deviation  $\sigma$ , means equally D-spaced, we have

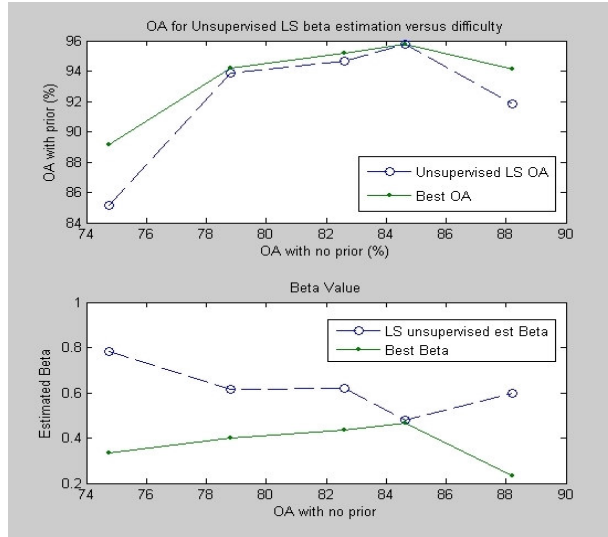
$$OA(\beta=0) = \left[ 1 - \left( \frac{c-1}{c} \operatorname{erfc} \left( \frac{D/2}{\sqrt{2} \times \sigma} \right) \right) \right] 100, \quad (9)$$

where  $\operatorname{erfc}(\cdot)$  is the complementary error function and  $c$  is the number of classes. Figure 1 shows the OA’s obtained by segmenting the image using ‘Algorithm-1’ (legend: unsupervised) and using the supervised estimated beta value (legend: supervised) against the values provided by (9).

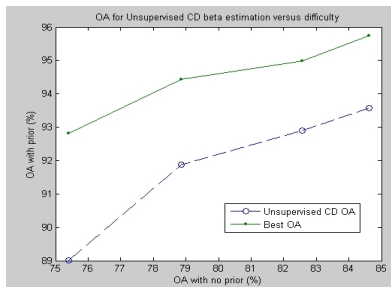


**Fig. 1.** On the left: overall accuracies against  $OA_{\text{MAP}}(\beta=0)$ . On the right: upper image is the MLL ground-truth with 3 classes; lower image are the simulated intensity values.

**Scenario 2:** In Figure 2 and 3 as in Scenario 1 but the  $OA(\beta=0)$  is now estimated by running the algorithm with  $\beta=0$ , since there is no close expression for it.



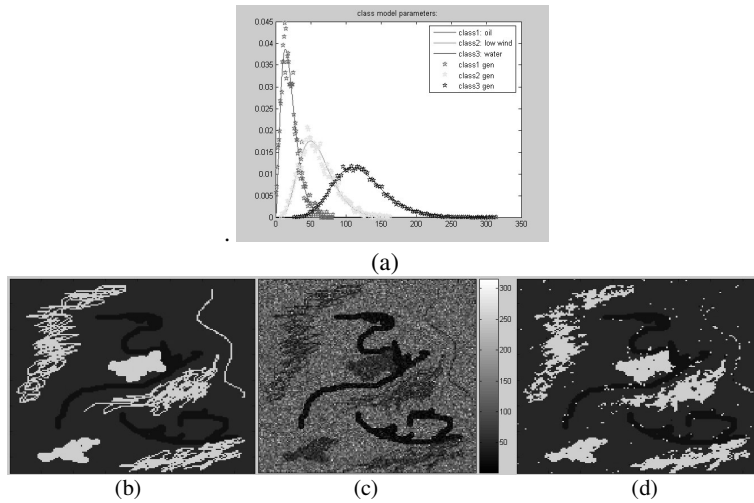
**Fig. 2.** Upper image: OA's obtained by unsupervised segmentation using LS and best achievable results with manually tuned  $\beta$ . Lower image: LS estimated  $\beta$  and best  $\beta$ .



**Fig. 3.** Overall accuracies obtained by unsupervised segmentation using CD and best achievable results with manually tuned  $\alpha$ -expansion.

Figure 2 shows the results obtained with the LS method and Figure 3 the results obtained with the CD method. Figure 4 displays an example of simulated image and corresponding segmentation results.

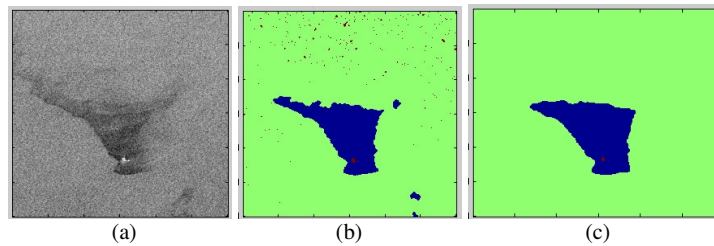
**Scenario 3:** By applying 'Algorithm-2' to an MLL image like the one adopted in Scenario 1, for a  $OA(\beta=0)$  of 85.9% given by expression (9), the achieved OA value is 94.5% using LS estimation. The Best achievable OA is 99.1%.



**Fig. 4.** (a) Density functions used to generate the simulated image with superimposed histogram of generated data set. (b) Ground-truth (c) Simulated image (d) Segmentation result using ‘Algorithm-1’: unsupervised LS estimated  $\beta = 0.4213$ , OA = 95.3%. The best achievable OA for this image was determined to be 95.4% . The OA for  $\beta = 0$  is 83.2%.

#### 4.2 Real Images

The ‘Algorithm-1’ has been applied to a real ERS-1 SAR image fragment. The scene (frame 2367, orbit 17211) containing the fragment has been acquired on 30 October 1994, and covers several oil platforms in the Norwegian and British sector of the North Sea. The image has been radiometric calibrated and corrected for the incidence angle effect. We have assigned a class to ‘oil’, a class to ‘water’ and a class to ‘platform’ and learned the class parameters using the supervised method described in [1]. Figure 6 displays the obtained results after applying ‘Algorithm-1’.



**Fig. 6.** (a) ERS image: intensity values (b) Segmentation with LS (c) Segmentation with CD.

## 6 Conclusions

The first results of applying the proposed methodology to simulated images with Gaussian and Gamma data models and to real ERS SAR data are promising. With 'Algorithm-1' higher OA accuracies have been achieved. The analysis of the resulting OA plots for Gamma data exhibits a maximum around circa  $OA_{no\ prior} = 85\%$ . This value corresponds to a value for the estimated  $\beta$  equal to the best  $\beta$ . At this point the add-on value provided by introducing a prior into the segmentation starts to decrease. Regarding 'Algorithm-2', the adopted methodology seems to be adequate but needs further assessment. In the example given in Scenario 3, the inclusion of the parameter estimation into the segmentation procedure only reduces the OA from 97,3% to 94,5%. By applying 'Algorithm-1' to a real ERS image, we have been able to successfully segment a platform of reduced size, the water and the oil. Hereby, the CD estimation method seems to provide a better segmentation than the LS method, contrarily to what happened for simulated images, where the LS method provided slightly better results.

These are preliminary results and more tests, with more trials per test, are required to fully determine the accuracy of the proposed methods..

**Acknowledgments.** The authors acknowledge Vladimir Kolmogorov for the max-flow/min-cut C++ code.

## References

1. S. Pelizzari, José M. B. Dias: Bayesian Adaptive Oil Spill Segmentation of SAR Images via Graph Cuts. Proceedings of the SeaSAR 2006 (2006), Frascati, Italy.
2. Y. Boykov, O. Veksler and R. Zabih: Fast Approximate Energy Minimization via Graph Cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No.11 (2001).
3. A. Montali, G. Giacinto *et al*: Supervised Pattern Classification Techniques for Oil Spill Classification in SAR Images: Preliminary Results, Proceedings of the SeaSAR 2006 (2006).
4. F. Galland: Synthetic Aperture Radar oil spill segmentation by stochastic complexity minimization. IEEE Geoscience and Remote Sensing Letters, Vol. 1, issue 4 (2004).
5. V. Kolmogorov and R. Zabih: What Energy Functions Can Be Minimized via Graph Cuts? IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 26, No.2, (2004).
6. S.Z.Li: Markov Random Field Modeling. Computer Vision, Computer Science Workbench, Springer (1995).