LOW-LEVEL PROCESSING OF POLSAR IMAGES WITH BINARY PARTITION TREES

Philippe Salembier[1*], *Samuel Foucher*[2], *Carlos López-Martínez*[1]

[1] Technical University of Catalonia, Barcelona, Spain
[2] Computer Research Institute of Montreal, Vision Team, Montreal, Canada
{philippe.salembier@upc.edu, sfoucher@crim.ca, carlos.lopez@tsc.upc.edu}

ABSTRACT

This paper discusses the interest of Binary Partition Trees (BPTs) and the usefulness of graph cuts for low-level processing of PolSAR images. BPTs group pixels to form homogeneous regions, which are hierarchically structured by inclusion in a tree. They provide multiple resolutions of description and easy access to subsets of regions. Once constructed, BPTs can be used for many applications including filtering, segmentation, classification and object detection. Many processing strategies consist in populating the tree with a specific feature and in applying a graph-cut called *pruning*. Different graph-cuts are discussed and analyzed in the context of PolSAR images for speckle filtering and segmentation.

Index Terms— Binary Partition Tree, PolSAR, graph-cut, speckle noise, segmentation

1. INTRODUCTION

The interest of Binary Partition Trees (BPTs) [12] has been recently investigated for remote sensing applications such as Polarimetric SAR (PolSAR) [2] and hyperspectral images [13]. BPTs are region-based representations in which pixels are grouped by similarity. They provide multiple resolutions of description and easy access to subsets of regions. Their construction is often based on an iterative region-merging algorithm: starting from an initial partition, the pair of most similar neighboring regions is iteratively merged until one region representing the entire image support is obtained. The BPT essentially stores the complete merging sequence in a tree structure. Once constructed, BPTs can be used for a large number of tasks including image filtering, object detection or classification [3].

In this paper, we discuss low level PolSAR image processing tasks. We study the interest of specific graph cut called *pruning* in this context. We show how partitions can be extracted from the tree and be used for speckle noise filtering or segmentation. The main paper contributions compared to [2, 3] is the proposal of new pruning strategies for PolSAR images as well as the objective evaluation of the performances thanks to a set of realistic simulated PolSAR images where the underlying ground truth is available [6]. Furthermore, we tackle the computational complexity issue and the use of super-pixel [1] to drastically reduce the time required to create BPTs.

The paper is organized as follows: Sec. 2 discusses the BPT creation and their processing with graph cut. A possible way to evaluate the quality of a BPT is presented in Sec. 3 and used in Sec. 4 to study the influence of the initial partition on the BPT construction and its robustness to noise. Sec. 5 presents and analyzes four pruning techniques for segmentation. Finally, conclusions are reported in Sec. 6.

[*] This work was performed while the first author was on leave at the Computer Research Institute of Montreal.

2. BINARY PARTITION TREE CREATION AND PROCESSING THROUGH GRAPH CUT

The BPT creation starts by the definition of an initial partition which can be composed of individual pixels as in [2, 3]. While this strategy guarantees a high precision as starting point of the merging process, it also implies high computational and memory costs as many regions have to be handled. As an alternative, the initial partition may correspond to an over-segmentation as a super-pixel partition [1]. Once the initial partition is defined, the BPT construction is done by iteratively merging the pair of most similar neighboring regions.

In the PolSAR case, the information carried by pixels of an image I corresponds to the one-look covariance matrix $\mathbf{Z}_{ij}^I$ of the scattering vector: $\mathbf{k} = [S_{hh}, \sqrt{2}S_{hv}, S_{vv}]^T$ measured on the resolution cell at location (i, j). The subindices h and v indicate the horizontal and vertical polarization states and $S_{pq\in\{h,v\}}$ represents the complex SAR data where p and q define respectively the reception and transmission polarizations. To construct the BPT, regions R can be modeled as in [2] by their mean covariance matrix $\mathbf{Z}_R = \frac{1}{|R|}\sum_{i,j\in R}\mathbf{Z}_{ij}^I$, where $|R|$ is the region number of pixels. The distance between neighboring regions defining the merging order can be measured as in [3] by the geodesic similarity [4]:

$$S(R_1, R_2) = \|log\left(\mathbf{Z}_{R_1}^{-1/2}\mathbf{Z}_{R_2}\mathbf{Z}_{R_1}^{-1/2}\right)\|.ln\left(\frac{2|R_1||R_2|}{|R_1|+|R_2|}\right) \quad (1)$$

where $log(.)$ is the matrix logarithm and $ln(.)$ the natural logarithm. Using this similarity measure, regions are iteratively merged until a unique region representing the entire image support is obtained. After each merging, a new region is created, its mean covariance matrix is computed and its similarity with its neighbors is updated. The regions belonging to the initial partition form the leaves of the BPT. During the merging process, the BPT is constructed by creating a parent node for each pair of merged regions.

Once the BPT has been constructed, it can be used for a wide range of applications including filtering, segmentation or classification. In many cases, the application relies on the extraction of a partition from the BPT. This process can be seen as a particular graph cut called *pruning* that can be formally defined as follows: Assume the tree root is connected to a *source* node and that all the tree leaves are connected to a *sink* node. A *pruning* is a graph cut that separates the tree into two connected components, one connected to the source and the other to the sink, in such a way that any pair of siblings falls in the same connected component. The connected component that includes the root node is itself a BPT and its leaves define a partition of the space. In the sequel, we discuss several examples of pruning in the context of PolSAR images. The first one will allow us to discuss the quality of a BPT itself through the evaluation of its robustness to speckle noise.

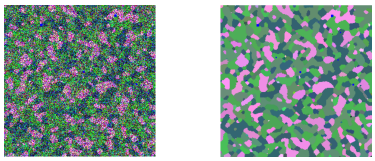

Fig. 1. Example of original PolSAR image (Left) and its corresponding ground-truth (Right). RGB-pauli color coding.

3. ROBUSTNESS OF BPT TO SPECKLE NOISE

One of the major issues in PolSAR is the speckle noise that results from the coherent integration of the electromagnetic waves. Speckle filtering aims at reducing noise within homogeneous extended targets while preserving spatial details [6]. Most filters are based on adaptive strategies using a sliding square window of fixed size [8].

In the context of PolSAR images and speckle noise, we want to evaluate the quality of a BPT and assess the influence of specific choices related to its construction (for example the initial partition as in Sec. 4). This is not a trivial task since many partitions can be extracted from a given BPT. As we are concerned by low-level processing and by removing the speckle noise as much as possible to allow a precise estimation of the polarimetric parameters, we rely on a dataset of PolSAR images on which the ground-truth polarimetric information is available. More precisely, we use the set of simulated PolSAR images [6] where the underlying ground-truth, i.e. the class regions, is modeled by Markov Random Fields. A set of typical polarimetric responses has been extracted from an AIRSAR image (L-band) so that they represent the 8 classes found in the $H/\overline{\alpha}$ plane and randomly assigned to each class. Finally, single look complex images have been generated from the polarimetric responses using a Cholesky decomposition [7]. An image example and its corresponding ground-truth are presented in Fig. 1. Thanks to this dataset involving ground-truth, we can measure the quality of a BPT.

Let us define *the quality of a BPT as the quality of the best image, according to a given error measure E, that can be extracted from it*. Extracting an image from the BPT consists in selecting a set of nodes forming a partition of the image and in assigning the mean covariance matrix of the region to its pixels. So the question is to identify the *ideal* partition that can be extracted from the tree.

The error measure between an image $I(i,j)$ and the ground-truth image $I_{GT}(i,j)$ we use is defined by [2]:

$$E(I, I_{GT}) = \frac{1}{N} \sum_{i,j} \|\mathbf{Z}_{ij}^{I} - \mathbf{Z}_{ij}^{I_{GT}}\| / \|\mathbf{Z}_{ij}^{I_{GT}}\| \qquad (2)$$

where N is the image number of pixels, $\mathbf{Z}_{ij}^{I}$ ($\mathbf{Z}_{ij}^{I_{GT}}$) the pixel value of image I (I_{GT}) at location (i,j) and $\|.\|$ the matrix norm. This measure is based on the average inverse signal to noise ratio.

As previously mentioned, the extraction of a partition from the BPT is defined by a pruning. To define the *ideal* pruning, let us use the following ideal criterion $C_{\text{ideal}} = \sum_R \phi_R$

$$\phi_R = \sum_{i,j \in R} \|\mathbf{Z}_R - \mathbf{Z}_{ij}^{I_{GT}}\| / \|\mathbf{Z}_{ij}^{I_{GT}}\|, \text{ s.t. } \{R\} \text{ is a partition} \qquad (3)$$

derived from Eq. 2 by noting that all pixels belonging to the same region R have the same covariance matrix Z_R. This criterion is ideal because it uses the ground-truth $Z^{I_{GT}}$ which is unknown in practice. However, it is very useful to quantify the BPT quality and to define an upperbound on the performances of possible pruning strategies.

Initial Partition	No filter (dB)	σ-Lee (dB), [9]	NL-SAR (dB), [5]	Normalized time
Pixel partition	-12,82	-16.39	-15.88	1
SLIC (size=2)	-12.23	-16.25	-15.78	1/15
SLIC (size=3)	-12.43	-16.22	-15.82	1/56
SLIC (size=4)	-12.72	-16.12	-15.88	1/121
SLIC (size=5)	-11.91	-16.11	-15,87	1/212

Table 1. Influence of the initial partition on the BPT quality: The three central columns show the quality (Eq. 2 measured in dB) of the ideal image extracted from the BPT as a function of the preprocessing filter and the size parameter of the super-pixel partition. The last column gives the BPT computation time normalized with respect to the case where the initial partition is composed of individual pixels. Results have been averaged over the entire dataset.

This criterion can be efficiently minimized using an dynamic programing algorithm originally proposed in [12] for global optimization. The solution consists in propagating local decisions in a bottom-up fashion. The BPT leaves are initially assumed to belong to the optimum partition. Then, one checks if it is better to represent the area covered by two sibling nodes as two independent regions $\{R_1, R_2\}$ or as a single region R (the common parent node of R_1 and R_2). The selection of the best choice is done by comparing the criterion ϕ_R evaluated on R with the sum of the costs ϕ_{R_1} and ϕ_{R_2}:

$$\text{If } \phi_R \leq \phi_{R_1} + \phi_{R_2} \begin{cases} \text{then} & \text{select } R \\ \text{else} & \text{select } R_1 \text{ and } R_2 \end{cases} \qquad (4)$$

The best choice (either "R" or "R_1 plus R_2") is stored in the node representing R with the corresponding cost value (ϕ_R or $\phi_{R_1} + \phi_{R_2}$). The procedure is iterated up to the root and defines the best partition. This algorithm finds the global optimum of the criterion on the tree and the selected regions are represented by their mean covariance matrix to create the filtered image. Finally, this image is used to compute the BPT quality with Eq. 2. Based on this strategy to evaluate a BPT quality, we can now investigate the impact of specific choices related to the BPT construction. As an example, we analyze the influence of the initial partitions in the following section.

4. SUPER-PIXEL INITIAL PARTITION AND ITS INFLUENCE ON THE BPT QUALITY

The initial partition used in [2, 3] was composed of regions involving only one pixels. The main drawback of this choice is the high number of initial regions and the corresponding cost in memory and computation. To see whether the number of initial regions can be drastically reduced while preserving quality, super-pixels created with the SLIC algorithm [1] have been tested. Only the diagonal elements of the covariance matrices have been used to generate the super-pixels. Furthermore, we have analyzed the interest of using a denoising filter before the computation of the super-pixel partition.

Thanks to the strategy presented in Sec. 3, we can compare the influence of various initial partitions on the BPT construction by extracting the ideal partition and measuring $E(I, I_{GT})$. The results are given in Table. 1. As can be seen, the best BPT is obtained with the use of a denoising filter prior the computation of super-pixel. Moreover, the use of the SLIC algorithm almost preserves the BPT quality but drastically reduces the computational complexity. It is therefore a very good alternative and, in the sequel, we use the σ-Lee filter (window size: 7x7) with the SLIC (size 4) super-pixels to create the initial partition.

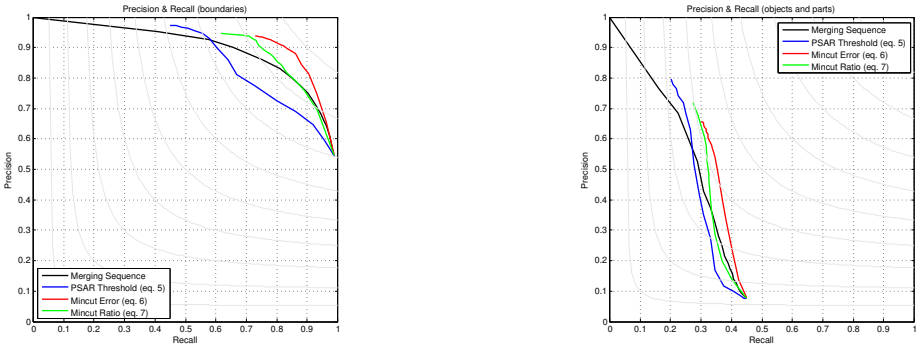

Fig. 2. Precision and Recall performances of the four pruning techniques (average over the entire dataset). Left: Precision and Recall for boundaries, Right: Precision and Recall for objects and parts.

5. SEGMENTATION AND LOW LEVEL PROCESSING THROUGH BPT PRUNING

This section discusses pruning techniques for low-level processing and grouping of PolSAR data. The main goal is to segment the images so that a precise estimation of the polarimetric parameters can be done. In the previous section, we have used an ideal pruning technique to assess the quality of the BPT but it cannot be used in practice as it relies on the ground-truth data. Here we study the interest of four pruning techniques useful in practical situations.

The first one simply consists in following the merging sequence and in stopping the iterative merging process when a predefined number N_R of regions is obtained [2]. Note that this can be viewed as a pruning of the BPT, but actually, there is no need to fully construct the BPT to compute the resulting partition.

The second pruning [2] consists in populating the tree nodes with a feature measuring the region homogeneity (difference between the pixel values and the region mean) given by:

$$\phi_R = \frac{1}{|R|} \sum_{i,j \in R} \|\mathbf{Z}_{ij}^I - \mathbf{Z}_R\| / \|\mathbf{Z}_R\| \quad (5)$$

Once the tree has been populated, the feature value of each node is compared to a predefined threshold. Note that the feature value is expected to be rather high for large regions and low for small regions. In the extreme case of single pixel regions, $\mathbf{Z}^R$ coincides with $\mathbf{Z}_{ij}^I$ and therefore $\phi_R = 0$. However, the feature of a parent node is not always larger or equal to the features of its siblings. To define the pruning, we have used the so-called *Max* rule [3] which consists in selecting on each branch the closest node to the root for which the homogeneity criterion is below the threshold.

We introduce now two new pruning strategies based on the minimization of a global criterion as in Sec. 3. The initial idea is to use $C = \sum_R \phi_R$ with ϕ_R being the homogeneity criterion $\phi_R = \sum_{i,j \in R} \|\mathbf{Z}_{ij}^I - \mathbf{Z}_R\| / \|\mathbf{Z}_R\|$. Note that this criterion is the same as the one defined by Eq. 5 without the averaging parameter $|R|$. However, on its own, this criterion is useless because a partition made of single pixel regions sets it to 0. Following classical approaches in functional optimization, ϕ_R can be interpreted as a data fidelity term and combined with a data regularization term which encourages the optimization to find partitions with a reduced number of regions. As simple data regularization, we use a constant value λ that penalizes the region presence. Therefore, the final homogeneity-based criterion to be minimized is given by $\phi_R^{Hom.}$:

$$\phi_R^{Hom.} = \sum_{i,j \in R} \|\mathbf{Z}_{ij}^I - \mathbf{Z}_R\| / \|\mathbf{Z}_R\| + \lambda, \text{ s.t. } \{R\} \text{ is a partition} \quad (6)$$

Finally, the last pruning is also based on a graph cut minimizing a global criterion but here the idea relies on ratio filters: if the ideal image structure is known (here represented by Z_R), then the ratio of the matrices diagonal values $\mathbf{Z}_{ij}^I(k,k)/\mathbf{Z}_R(k,k)$ should only contain noise of variance 1 and no structure information. If the structure information is absent, the energy of the ratio should be minimum. This reasoning leads to the following minimization criterion involving as before a data fidelity term and a data regularization term:

$$\phi_R^{Ratio} = \sum_{i,j \in R} \sum_{k=1,2,3} \| \frac{\mathbf{Z}_{ij}^I(k,k)}{\mathbf{Z}_R(k,k)} \| + \lambda, \text{ s.t. } \{R\} \text{ is a partition} \quad (7)$$

Fig. 2 shows the evaluation of the segmentation results as classically done in the supervised case through Precision and Recall curves. On the left side, the so called *Precision and Recall for boundaries* [10] is presented. In this case, each partition is evaluated by considering all pairs of neighboring pixels and by classifying them in either boundary or interior segments. The Precision and the Recall values of this classification are evaluated by comparison with the classification resulting from the ground-truth partition. In addition to this boundary-oriented evaluation, a region-oriented evaluation known as the *Precision and Recall for objects and parts* [11] is presented on the right side of Fig. 2. In this context, regions of the partition are considered as potential candidates to form regions of the ground-truth partition, and are classified as correct or not. In both cases, the curves are formed by modifying the pruning parameter allowing to have coarser or finer partitions. This pruning parameter is the number of region, the threshold on ϕ_R or the λ value, respectively for the pruning following the merging sequence, the one thresholding the homogeneity criterion (Eq. 5) or the two techniques involving global optimization (Eqs. 6 and 7). The ideal system has Precision and Recall values equal to one. So, the closest curve to the upper-right corner of the plots identifies the best system.

As can be seen in Fig. 2, the region-oriented evaluation is more severe than the boundary-oriented evaluation (see [11] for details). However, the conclusions on both plots are the same: the best pruning technique is the one based on the global optimization of the homogeneity (Eq. 6). The one defined by Eq. 7 provides good results for high precision (coarse partition). The pruning following the merging sequence does not lead to the best results. This observation highlights the interest of constructing the BPT to extract partitions that have not been observed during the merging process. Moreover, in practice, it is difficult to a priori define the appropriate number of regions. Finally, the pruning involving the thresholding on the homogeneity criterion provides interesting results for high precision.

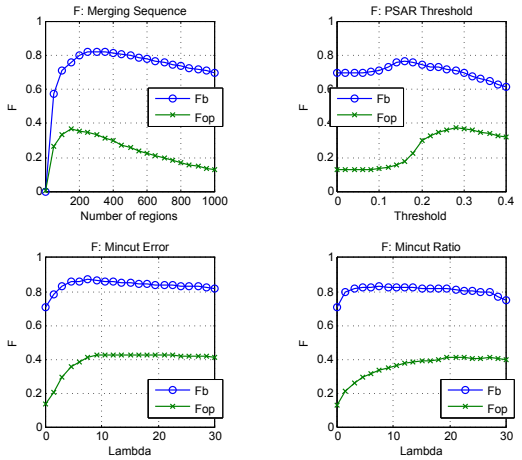

Fig. 3. F value as a function of the pruning parameter. F_b (F_{op})corresponds to the Precision and Recall for boundaries (object and parts) curves. Top: Merging sequence and threshold on homogeneity (Eq. 5). Bottom: mincut on homogeneity criterion (Eq. 6) and on ratio image (Eq. 7).

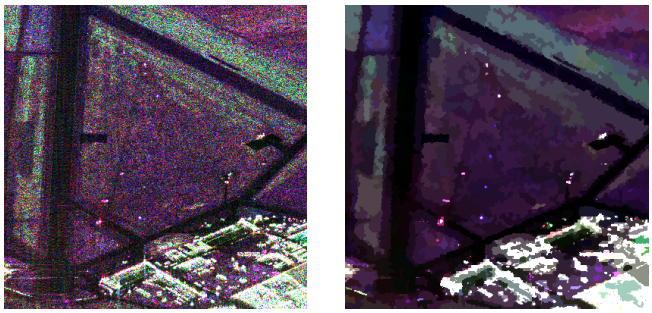

Fig. 4. Results on real images. Right: Original images (RGB Pauli composition). Left: Mincut on homogeneity criterion (Eq. 6).

Precision and Recall curves describe the system performances for the complete range of the pruning parameter values. However, they do not efficiently describe the system sensitivity to the parameter value. To this end, Fig. 3 presents for each pruning technique, the F value as a function of the pruning parameter. The F value is classically used to summarize the Precision P and Recall R trade-off. It is the harmonic mean of P and R: $F = 2PR/(P+R)$. Fig. 3 reveals that the most stable pruning involves the global minimization of the homogeneity (Eq. 6). It practice, this means that λ values between 5 and 20 will extract similar partitions from the tree and does not have to be fine tuned. We can also assess the robustness of this pruning to speckle noise by computing the error measure E (Eq. 2). With $\lambda = 5$, this error is equal to $-15,15$ dB. As shown in Table 1 (SLIC size=4, σ-Lee filter), there is therefore only 1dB difference between this pruning and the ideal one.

Finally, the pruning with global optimization of the homogeneity (Eq. 6) has been evaluated on a L-band fully polarimetric data set acquired in 2003 by the Deutsches Zentrum für Luftund Raumfahrt (DLR) ESAR system over the area of the Oberpfaffenhofen airport near Munich, Germany. The images are Single Look Complex with a pixel size of 1,5x1,5m. Results are shown in Fig. 4 together with the original image. They visually highlight the interest of the BPT to perform low-level processing of PolSAR images while preserving the spatial resolution of the content.

6. CONCLUSIONS

This paper has discussed the interest of Binary Partition Trees (BPTs) for PolSAR images and highlighted the usefulness of a particular type of graph cut called pruning to extract partitions from the BPT. Specific pruning techniques have been defined to evaluate the quality of BPT and to perform low-level grouping allowing a precise estimation of the polarimetric information to be done without losing in terms of spatial resolution. In this context, the pruning technique resulting from the global optimization of a criterion minimizing the region homogeneity has proved to be very efficient and robust.

7. REFERENCES

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. PAMI*, 34(11):2274 – 2282, 2012.

[2] A. Alonso-Gonzalez, C. Lopez-Martinez, and P. Salembier. Filtering and segmentation of polarimetric SAR data based on binary partition trees. *IEEE Trans. GRS*, 50(2):593–605, 2012.

[3] A. Alonso-Gonzalez, S. Valero, J. Chanussot, C. Lopez-Martinez, and P. Salembier. Processing multidimensional SAR and hyperspectral images with binary partition tree. *Proceedings of IEEE*, 101(3):723–747, 2013.

[4] F. Barbaresco. Interactions between symmetric cone and information geometries: Bruhat-tits and siegel spaces models for high resolution autoregressive doppler imagery. In *Emerging Trends in Visual Computing*, volume 5416, pages 124–163. LNCS, 2009.

[5] C.A. Deledalle, F. Tupin, and L. Denis. Polarimetric SAR estimation based on non-local means. In *IGARSS*, 2010.

[6] S. Foucher and C. Lopez-Martinez. Analysis, evaluation, and comparison of polarimetric sar speckle filtering techniques. *IEEE Trans. IP*, 23(4):1751–1764, 2014.

[7] J.-L. Lee, T.L. Ainsworth, J.P. Kelly, and C. López-Martínez. Evaluation and bias removal of multilook effect on entropy/alpha/anisotropy in polarimetric SAR decomposition. *IEEE Trans. GRS*, 46(10):3039–3051, 2008.

[8] J.S. Lee, M.R. Grunes, and G. De Grandi. Polarimetric SAR speckle filtering and its implication for classification. *IEEE Trans. GRS*, 37(5):2363–2373, 1999.

[9] J.S. Lee, J.H . Wen, T.L. Ainsworth, K.S. Chen, and A.J. Chen. Improved sigma filter for speckle filtering of SAR imagery. *IEEE Trans. GRS*, 47(1):202 – 213, 2009.

[10] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Trans. on PAMI*, 26(5):530–549, 2004.

[11] J. Pont-Tuset and F. Marques. Measures and meta-measures for the supervised evaluation of image segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2013.

[12] P. Salembier and L. Garrido. Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. *IEEE Trans. IP*, 9(4):561 –576, 2000.

[13] S. Valero, P. Salembier, and J. Chanussot. Hyperspectral image representation and processing with binary partition trees. *IEEE Trans. IP*, 22(4):1430 – 1443, 2013.