Texture and color segmentation based on the combined use of the structure tensor and the image components

Rodrigo de Luis-García, Rachid Deriche, Carlos Alberola-López

Abstract

In this paper, we propose a novel segmentation scheme for textured gray-level and color images based on the combined use of the local structure tensor and the original image components. The structure tensor is a well-established tool for image segmentation and has been successfully employed for unsupervised segmentation of textured gray-level and color images. The original image components can also provide very useful information. Therefore, a combined segmentation approach has been designed that combines both elements within a common energy minimization framework. Besides, an original method is proposed to dynamically adapt the relative weight of these two pieces of information. Quantitative experimental results on a large number of gray-level and color images show the improved performance of the proposed approach, in comparison to several related approaches in recent studies. Experiments have also been carried out on real world images in order to validate the proposed method.

1. Introduction

Image segmentation is a key step for image interpretation applications and it may be of high interest for some image processing applications, such as image coding. Generally speaking, the segmentation process relies on the extraction of appropriate features from the image that bear high discriminative power, the more discriminative the more likely the eventual segmentation will be successful.

The present work introduces a novel segmentation method for textured images, a well-known problem for which a golden solution has not been reported. Many different approaches have been employed in the literature for texture segmentation and texture classification, which is a closely related problem. These include those based on Markov Random Fields [1–4], Gabor filters [5–9], multiple resolution techniques [10,11] or neural network...
classification methods [12,13], just to mention a few. Most of these approaches provide segmentation results with a spotty appearance, as they are pixelwise classification methods. As opposed to them, contour-oriented segmentation has gained much relevance in the last years.

In this paper, we propose a contour-oriented segmentation approach for textured images. As for implementation, level-set methods [14–18] are employed for curve evolution. They have gained much relevance lately due to their good properties: as opposed to parametric active contours [19–21], the curve is represented implicitly, so topological changes can be handled naturally; it can be directly extended to higher dimensions and efficient schemes for numerical solutions have been developed [22]. Moreover, different types of information can be used in a common level set framework, including boundary information [15,23,24], region information [25–30] and even shape prior information [31–34].

In our approach, the texture features will be employed as region information in a geodesic active regions (GAR) framework. This model was proposed by Paragios and Deriche [28] and constitutes a common framework for contour-oriented image segmentation problems, where different sources of information can be incorporated within a general Bayesian formulation.

With regard to texture feature extraction, we propose an adaptive mixed approach based on the combined use of the local structure tensor (LST) and the image components. The LST was proposed for orientation estimation [35–38] and it is widely accepted as a powerful feature extractor for textured images. Based on the LST, Rousson et al. proposed in [39] a segmentation method for textured images that applies the GAR model to a vector-valued image where the channels are the components of the LST. This method showed interesting results. However, the advantages of extracting the texture features with the LST are partially lost because of the vector processing of that information. Based on this work, in [40] tensor processing was applied to the LST for texture segmentation. In that work, intrinsic tensor dissimilarity measures on the structure tensor drew a performance improvement over the vector processing of these features. Even though the use of the LST information for texture segmentation has shown considerably good results, it suffers from a major drawback, namely, the LST does not include any intensity information (or color information) from the image, which may constitute a significant information misuse or loss. This problem was partially solved in [40] with the introduction of several modified structure tensor architectures (i.e. extended and compact structure tensor) but again, two main issues remain unsolved. First one is adaptivity. A detailed analysis of each image characteristic shows that the relative importance of the texture and components information may vary from case to case. Second, although texture information may be appropriately encoded in a tensor architecture, the components information naturally fits a vector scheme (in fact, the convenience of a separate analysis of the texture and the color information was shown in [49]). Therefore, different dissimilarity measures should be employed for each case.

Different approaches for the combination of texture and intensity information for image segmentation have been proposed recently in the literature. In [50], a supervised classification method is designed for images containing both textured and non-textured parts, based on a texture-structure decomposition of the image. The global segmentation is performed by combining the segmentation results for the geometrical and textured components using a logical framework. However, it is necessary to know what the logical combination of the geometrical and textural patterns into the different classes should be, thus making this approach strongly supervised. In [51], a technique is devised to measure the local scale based on a property of the total variational flow. This novel feature is added to the intensity and the components of the structure tensor to perform the segmentation in a feature space of dimension 5. This method incorporates the local scale as a valuable new feature to describe textures, but does not balance the importance of the different kinds of information for each particular image. Other attempts more closely related to our approach will be later discussed, once our method is described.

The method presented in this work is intended for the segmentation of both gray-level and color textured images. With regard to the segmentation
of color textures, an extensive amount of work can be found in the literature \[52–56\]. In \[57\], the dichromatic reflection model was introduced that describes how photometric changes, such as shadows and specularities, influence the \textit{RGB}-values in an image. Based on this model, different algorithms have been proposed that are invariant to different photometric phenomena \[58–60\]. Recent approaches focus precisely on the characterization and segmentation of images using features that are independent from these photometric variations, such as the color invariants by Geusebroek et al. \[61\] or the photometric quasi-invariants by van de Weijer et al. \[62\]. Invariance to photometric variations has not been considered in this paper, as we intend to distinguish between different textures without considering photometric variations such as shadows and specularities. Therefore, the color structure tensor proposed by Di Zenzo \[63\] has been employed as the basis of the texture descriptors employed in this work.

In this paper, we propose an adaptive segmentation method that combines the LST with the image components into a common energy minimization framework. For the LST information, the Kullback–Leibler distance is employed as an intrinsic tensor dissimilarity measure, whereas the Euclidean distance is applied for the components information. This way, a natural metric is applied to each kind of feature. Both energy terms are balanced for minimization and their relative weights are dynamically estimated based on their respective discriminative powers in the current state of the segmentation process.

The main contributions of this work are as follows: First, not only texture and image components are employed as features in a common energy minimization framework, but also adequate representations and metrics are applied for each piece of information without altering the global scheme. In this sense, the proposed approach differs from that presented in \[40\], where modified tensor architectures jointly encode the texture and intensity information. Second, an adaptive weighing algorithm has been added that measures the discriminative power of the two different types of features and consequently adjusts the relative importance of the different energy terms. Finally, extensive experiments have been carried out in order to test the proposed method and to compare it with other approaches. A major effort has been made on obtaining quantitative results from large data sets, which allow an objective comparison. Those sort of studies are considerably rare to find in the literature. All in all, comparisons indicate that the novel segmentation method described in this paper yields satisfactory results and indeed improves the state of the art.

Our approach is mostly unsupervised in nature, as it does not need any prior training step; however, strictly speaking, some parameters related to the scale of the problem (scale of the LST and the smoothing term in the level set evolution) need to be tuned.

The method described in this paper is intended for a two-class segmentation problem. However, the energy formulation can be naturally extended to multiple class scenarios. Multiple level sets can be used for the representation of the different regions and their boundaries \[28\], although problems of vacuum and overlap appear and need to be solved by imposing additional constraints. This drawback was elegantly solved in \[64\], also reducing the number of level sets needed to represent the same number of regions. The energy term defined to perform the image segmentation can be even minimized with respect to the number of regions, if it is not known a priori \[65\].

The paper is organized as follows: the next section describes the LST for texture extraction. Then, we explain the segmentation method, which is posed as an energy minimization problem and solved by means of a level set framework. In Section 4, we focus on the feature extraction process. The nonlinear LST is first considered, and then its modified variants, which were presented in \[40\], are briefly described as will be afterwards considered for experimental comparison with the algorithms proposed in this paper. Then, the original image components are considered. Section 5 introduces the adaptive segmentation method proposed. In Section 6 we describe the extensive experiments carried out to test and validate the method, followed by a discussion of the results. Finally, a brief summary is presented.

2. Structure tensor for texture extraction

Local orientation is indeed a major component of textures. Although this feature is not easy to estimate or even to represent, the LST \[35–38\] is widely accepted to provide a compact representation of orientation. Moreover, tensor algebra is a
solid mathematical body that supports further analysis in the tensor domain.

Estimating the LST is not a trivial task. Two major approaches have been proposed: gradient-based methods [35–37] and local energy-based methods [38]. Although the second method may have advantages over the first method, a gradient-based approach has been followed in this work mainly because of simplicity reasons, as the LST estimation itself is out of the scope of this paper.

For a scalar image $I$, the LST is defined as follows [35–37,66–68]:

$$J_\rho = K_\rho \ast (\nabla I \nabla I^T) = \begin{pmatrix} K_\rho \ast I_x^2 & K_\rho \ast I_x I_y \\ K_\rho \ast I_x I_y & K_\rho \ast I_y^2 \end{pmatrix},$$

(1)

where $K_\rho$ is a Gaussian kernel with standard deviation $\rho$, and subscripts denote partial derivatives. The tensor yields three feature channels: $K_\rho \ast I_x^2$, $K_\rho \ast I_y^2$ and $K_\rho \ast I_x I_y$. In the case of a vector-valued image, all channels are taken into account by summing the tensor products of the particular channels [63]:

$$J_\rho = K_\rho \ast \left( \sum_{i=1}^{N} \nabla I_i \nabla I_i^T \right),$$

(2)

where $N$ is the total number of vector channels. We adopt this simple metric for vector valued images since the paper does not focus on optimal processing regarding in color spaces. Better choices could be made for a particular choice of a color space. The smoothing with a Gaussian kernel may make the structure tensor suffer from the dislocation of edges, leading to inaccurate segmentation results near region boundaries. To solve this problem, Brox and Weickert [51,69,70] proposed to replace the Gaussian smoothing by nonlinear diffusion. In [39] it was demonstrated that this (nonlinear) structure tensor performs very well for texture discrimination.

Nonlinear diffusion was introduced by Perona and Malik [71], and aims at reducing the smoothing in the presence of edges. The resulting diffusion equation is:

$$\partial_t u = \text{div}(g(|\nabla u|)\nabla u)$$

(3)

with $u(t = 0)$ being the image $I$ and $g(\cdot)$ a decreasing function. The choice of the diffusivity function $g$ is a very important and critical issue [72], although out of the scope of this paper. In this work, $g(s) = (s + \epsilon)^{-1}$ was employed [73].

This diffusion equation is only suitable for scalar-valued data. For vector-valued data a new version of nonlinear diffusion was introduced in [74]:

$$\partial_t u_i = \text{div} \left( g \left( \sum_{k=1}^{N} |\nabla u_k|^2 \right) \nabla u_i \right) \quad \forall i,$$

(4)

where $u_i$ is an evolving vector channel. For implementation, the AOS (additive operator splitting) scheme, proposed in [75], allows for a much more efficient and faster computation than a straightforward implementation scheme.

Finally, the structure tensor is obtained, for a scalar image, by applying Eq. (4) with initial conditions $u = [I_x^2, I_y^2, I_x I_y]^T$. For all the experiments carried out in this paper, the step size for the diffusion process was chosen to be $10^5$, and the number of steps was equal to 10. Parameter $\epsilon$ of the diffusivity function $g$ was chosen to be $\epsilon = 0.001$.

3. Segmentation method

Throughout this work, a segmentation scheme based on the GAR model is employed [28]. The segmentation is performed by means of the maximum a posteriori frame partition. Specifically, let $\mathcal{P}(\Omega)$ denote the partition of the image $I$ into a set of classes. Then, given the observed image $I$, the a posteriori frame partition probability can be expressed, using Bayes rule, as

$$p(\mathcal{P}(\Omega)|I) = \frac{p(I|\mathcal{P}(\Omega))}{p(I)} p(\mathcal{P}(\Omega)),$$

(5)

where $p(I)$ is the probability of image $I$ and $p(\mathcal{P}(\Omega))$ is the probability of the partition $\mathcal{P}(\Omega)$. The term $p(I|\mathcal{P}(\Omega))$ is the likelihood function of image $I$, given the partition $\mathcal{P}(\Omega)$. If we assume that all partitions are equally probable, then $p(\mathcal{P}(\Omega))$ can be ignored, and therefore we have

$$p(\mathcal{P}(\Omega)|I) \propto p(I|\mathcal{P}(\Omega)).$$

(6)

For the two-class case let $\mathcal{P}(\Omega) = \{\Omega_1, \Omega_2\}$ denote the partition of the image domain $\Omega$. Assuming conditional independence of the image given the two class labellings we can write

$$p(\mathcal{P}(\Omega)|I) \propto p(I|\Omega_1) p(I|\Omega_2).$$

(7)

Finally, if the pixels within each region are conditionally independent, the maximization of the a posteriori segmentation probability is equivalent
to the minimization of the energy obtained after applying the negative logarithm:

\[ E(\Omega_1, \Omega_2) = -\int_{\Omega_1} \log p(I(x)|\Omega_1) \, dx - \int_{\Omega_2} \log p(I(x)|\Omega_2) \, dx. \]  

This energy is the basis of all the functionals considered in this paper, and has also been employed in a number of other works in the literature [28,39,30,72]. Now, let us consider the field \( F(x) \). Depending on the nature of the chosen feature to build the field, an appropriate distance measure should be adopted: Consider that the feature to build the field, an appropriate distance can be minimized with respect to that level set function. For mathematical convenience we assume that \( p_{d,1} \) and \( p_{d,2} \) are Gaussian of zero mean and variances \( \sigma_1^2 \) and \( \sigma_2^2 \); a right-sided functional will be used to account for the non-negativity of any distance, i.e.,

\[ p_{dd} = \frac{2}{\sqrt{2\pi\sigma_1^2}} e^{-d^2/2\sigma_1^2}, \quad d \geq 0. \]  

For the minimization of the defined energy, we introduce a level set function \( \phi : \Omega \to \mathbb{R} \), where \( \phi(x) = \mathcal{D}(x, \partial\Omega) \). \( \mathcal{D}(x, \partial\Omega) \) stands for the signed Euclidean distance between \( x \) and the boundary between regions \( \Omega_1 \) and \( \Omega_2 \). The energy term in Eq. (9) can be rewritten using this level set function, and can be minimized with respect to that level set function through a gradient descent. If we add a regularization constraint on the length of \( \partial\Omega \) to the corresponding Euler–Lagrange equation for \( \phi \), the following evolution equation is obtained (see [29] for details)

\[ \frac{\partial \phi}{\partial t}(x) = \delta \text{div} \left( \frac{\nabla \phi}{|\nabla \phi|} \right) - \frac{d^2(F(x), F_1)}{\sigma_1^2} + \frac{d^2(F(x), F_2)}{\sigma_2^2} - \log \frac{\sigma_1^2}{\sigma_2^2}. \]  

The Gaussian parameters are updated at each iteration using

\[ \sigma_i^2 = \frac{1}{|\Omega_i|} \int_{\Omega_i} d^2(F(x), F_i) \, dx. \]  

Parameter \( v \) in Eq. (11) weighs the importance of the regularity constraint on the contour length. It is related to the scale of the detail in the image, and needs to be empirically adjusted. For all the experiments made, it was set \( v = 1 \).

4. Feature extraction

4.1. Local structure tensor

Starting from the original image \( I \), let us consider an image containing at each pixel, instead of the original gray scalar or color vector value, the \( 2 \times 2 \) LST described in Section 2:

\[ T = \begin{pmatrix} \hat{I}_x^2 & \hat{I}_x \hat{I}_y \\ \hat{I}_x \hat{I}_y & \hat{I}_y^2 \end{pmatrix}, \]  

\[ T_c = \sum_i \begin{pmatrix} (\hat{I}_i)_x^2 & (\hat{I}_i)_x (\hat{I}_i)_y \\ (\hat{I}_i)_x (\hat{I}_i)_y & (\hat{I}_i)_y^2 \end{pmatrix}, \]  

for gray level or color images, respectively, where the “hatted” components denote the nonlinearly diffused components. Note that, according to [76], the coupling of the channels by a joint diffusivity in Eq. (4) ensures the preservation of positive semidefiniteness.

As the LST, the diffusion tensor that appears in DTI is a symmetric positive-semidefinite (SPD) tensor. This imaging modality produces a volumetric image containing, at each voxel, a \( 3 \times 3 \) SPD tensor. For the segmentation of anatomical structures from such tensor fields, Zhukov et al. [48]
defined an invariant anisotropy measure in order to isolate anisotropic regions of the brain. However, it draws a reduction of the full tensor to a single scalar, thus losing much discriminating power. Wiegel et al. [47] proposed to use the Frobenius norm of the difference of tensors together with a spatial coherence term in a k-means algorithm. However, this approach relies on a very restrictive hypothesis, which is rarely fulfilled. Jonasson et al. [42] employed a geometric measure of overlap (so-called divergence) between two tensors. Finally, in [45, 46], a SPD tensor is interpreted as a covariance matrix of a Gaussian distribution. Then, the natural distance between two tensors. It was shown in [45] that there is a closed form for the level set, as shown in Section 3, if we define the mean values \( T_1 \) and \( T_2 \) as follows:

\[
T_i = \text{argmin} \int_{\Omega_i} d(T_i, T(x)) \, dx.
\]

(17)

It was shown in [45] that there is a closed form for such a tensor field mean value, given by

\[
T_i = \sqrt{B^{-1}} \left[ \sqrt{BA} \sqrt{B} \right] \sqrt{B^{-1}},
\]

(18)

where \( A = \int_{\Omega_i} T(x) \, dx \) and \( B = \int_{\Omega_i} T^{-1}(x) \, dx \).

This KL-based approach has been successfully employed for Diffusion MRI segmentation in [43], and for texture segmentation in [40].

4.1.1. Advanced structure tensor architectures

In order to overcome the disadvantage of not directly using any gray level information (or color information, in the case of vector-valued images), new variants of the LST were proposed in [40] that incorporate this information without losing its nice properties. Here, we briefly describe these architectures, as experimental results will be given in Section 6 comparing this approach with the methods proposed in this paper, which consider the classic nonlinear LST and the image components separately, with their own appropriate metrics.

First, the extended structure tensor (EST) is defined for a scalar image as follows:

\[
T_E = D(vv^T) = \begin{pmatrix}
\tilde{I}_x^2 & \tilde{I}_x\tilde{I}_y & \tilde{I}_x^3
\end{pmatrix}.
\]

(19)

where \( v = [I_x, I_y, I]^T \). \( D(\cdot) \) denotes the diffusion process applied to the tensor components, such as the nonlinear diffusion described in Section 2, which is also denoted by the hat operator.

With respect to color images, the extended tensor is obtained as \( T_E = D(vw^T) \), where \( w = [I_x^T, I_y^T, I_R^T, I_G^T, I_B^T]^T \) and \( I' = I_R + I_G + I_B/3 \).

The EST contains a lot of useful information for the discrimination between different textures. However, the \( 3 \times 3 \) tensor (\( 5 \times 5 \) for color images) implies that the energy minimization \( (5 \times 3) \) is done in a higher dimensional space, which can be too difficult and result in multiple local minima. To overcome this disadvantage, the tensor size can be reduced without losing valuable information, using principal component analysis (PCA).

For scalar images, let us consider again the vector \( v = [I_x, I_y, I]^T \), used to construct the EST. Then we have a three-dimensional data matrix of size \( M \times N \times d \), where \( M \times N \) is the size of the image and \( d \) is the dimension of the vector \( v \) (\( d = 3 \). The same data can be arranged as a two-dimensional matrix \( X \) of size \( M \cdot N \times d \), where each row can be thought of as an observation. Using PCA, the linear transformation is given by

\[
Y = XH,
\]

(20)

where \( Y \) is the derived \( M \cdot N \times d' \) pattern matrix (\( d' \leq d \)), \( H \) is the \( d \times d' \) transformation matrix

(footnote continued)

manifold [40, 43]. However, for simplicity reasons, that discussion has not been included in this work.

\footnote{Instead of the KL distance, a geodesic can be defined that takes into account the Riemannian structure of the underlying manifold [40, 43]. However, for simplicity reasons, that discussion has not been included in this work.}
whose columns are the eigenvectors corresponding to the \(d\)' largest eigenvalues of the \(d \times d\) covariance matrix of the input data, \(X\).

Using this transformation, it is possible to obtain 
\[ v' = \text{PCA}(v) = [v_1' v_2']^T, \]
which is then used to construct the compact structure tensor (CST)
\[ T_C = D(v'(v')^T) = \begin{pmatrix} (v_1')^2 & v_1' v_2' \\ v_1' v_2' & (v_2')^2 \end{pmatrix}. \]
(21)

For color images, the same procedure can be used to reduce the \(5 \times 5\) extended structure tensor to the \(2 \times 2\) CST.

In some cases, however, significant information can be lost as the dimension reduction may be severe for the color case (\(5 \times 5\) to \(2 \times 2\)). This can be solved by applying a dimensionality reduction to a size that keeps all the eigenvectors responsible for a minimum percentage of the total variance, i.e. of the sum of the squared eigenvalues. Then, using this procedure a structure tensor of variable size is obtained, which is called adaptive compact structure tensor (ACST).

4.2. Image components

Starting from the original image \(I\), which can be scalar or vector valued, we can consider the Euclidean distances from the pixel values to the mean values \(\mu_1\) and \(\mu_2\) over their respective domains \(\Omega_1\) and \(\Omega_2:\)
\[ d_c^2(I(x), \mu_i) = \sum_{k=1}^{N} (I_k(x) - \mu_{i_k})^2, \]
(22)
where the subscript \(k\) denotes the image components and \(N\) is the number of channels.

The distributions of the Euclidean distances can be then modeled, for \(d_c > 0\), as right-sided Gaussians with zero means and variances \(\sigma_{i1}^2\) and \(\sigma_{i2}^2\), so that the level set evolution equation presented in Eq. (11) can be applied.

5. Combined adaptive segmentation

The segmentation method studied in Section 3 is designed to work on a distance measure derived from the extracted features. In order to take into account both the classic LST (Eqs. (13) and (14)) and the image components through the KL distance and the Euclidean distance, respectively, a hybrid energy term is proposed:
\[
E(\Omega_1, \Omega_2) = \frac{\beta_1}{\beta_1 + \beta_2} \left[ - \int_{\Omega_1} \log p_{t,d,1}(d_c(I(x), T_1)) \, dx \right.
+ \int_{\Omega_2} \log p_{t,d,2}(d_c(I(x), T_2)) \, dx
\]
\[
+ \frac{\beta_2}{\beta_1 + \beta_2} \left[ - \int_{\Omega_1} \log p_{c,d,1}(d_c(I(x), \mu_1)) \, dx \right.
- \int_{\Omega_2} \log p_{c,d,2}(d_c(I(x), \mu_2)) \, dx \right],
\]
(23)
where \(p_{t,d,1}\) and \(p_{t,d,2}\) are the probability density functions of the KL or geodesic information distances in the LST domain, as seen in Section 4.1, and \(p_{c,d,1}\) and \(p_{c,d,2}\) are the probability density functions of the Euclidean distances in the image components domain, as described in Section 4.2. \(\beta_1\) and \(\beta_2\) are the relative weights of the LST and components based terms in Eq. (23). The evolution equation to minimize this energy is
\[
\left\{ \begin{array}{l}
\sigma_{c1}^2 = \frac{1}{|\Omega_1|} \int_{\Omega_1} d_c^2(I(x), \mu_1) \, dx, \\
\sigma_{c2}^2 = \frac{1}{|\Omega_2|} \int_{\Omega_2} d_c^2(I(x), \mu_2) \, dx, \\
\frac{\partial \phi}{\partial t}(x) = \delta(\phi) \left( \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|} \right)
\end{array} \right.
\]
\[
- \frac{\beta_1}{\beta_1 + \beta_2} \left[ - \frac{d_c^2(T_1(x), T_1)}{\sigma_{i1}^2} + \frac{d_c^2(T_1(x), T_2)}{\sigma_{i2}^2} - \log \frac{\sigma_{i1}^2}{\sigma_{i2}^2} \right]
- \frac{\beta_2}{\beta_1 + \beta_2} \left[ - \frac{d_c^2(I(x), \mu_1)}{\sigma_{c1}^2} + \frac{d_c^2(I(x), \mu_2)}{\sigma_{c2}^2} - \log \frac{\sigma_{c1}^2}{\sigma_{c2}^2} \right]
\]
(24)

If \(\beta_1 = \beta_2\), the LST and the image components are equally weighed. However, it is also possible to adaptively adjust this parameter depending on the relative discriminative power of the LST and components terms.\(^4\) The estimation of the relative importance of both types of features is related to the problem of the structure-texture decomposition of images, which is an important issue in the literature \(\cite{[77,78]}\), especially for denoising purposes. Usually, image decomposition is performed via an energy minimization process. This kind of decomposition still needs an initial guess of the splitting parameter between the geometrical and textural components and is beyond the scope of this paper. Other efforts on measuring the discriminative power of the different

\(^4\)Strictly speaking, varying \(\beta_i, i = 1, 2\), make the objective function in Eq. (23) change. However, empirical evidence favours this approach as results in Section 6 indicate.
channels are those in [79,9]. In the first work, Cardelino et al. considered the PDFs of the different feature channels in both regions \( \Omega_1 \) and \( \Omega_2 \) and computed their Kullback–Leibler distance. A large value for that distance belonging to a particular channel means that the channel provides good discrimination power. However, in our case the pieces of information used are not directly comparable in terms of the same distance function (the Euclidean distance for the image components is a bounded quantity, since components themselves are bounded whereas the Kullback–Leibler distance for the LST information is not). Besides, in [79] the previous slice of the volume data is employed for this procedure as a correct distinction between both regions is needed, which is not available in our case. With regard to the work by Sandberg et al. [9], the variations on the different features between both regions are considered as a criterion to discriminate among the feature channels. Again, this approach needs all channels be commensurate, which, as previously stated, is not our case. Consequently, an alternative procedure has been designed for the estimation of \( \beta_1 \) and \( \beta_2 \). In order to do so, let us first consider the following expression:

\[
Q = \int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx,
\]

where \( p_1(x) \) and \( p_2(x) \) are the resulting probability density functions when considering two Gaussian distributions with zero mean and \( T_1 \) and \( T_2 \) as covariance matrices, respectively. Indeed, \( Q \) is a measure of the overlap between both distributions, and \( 0 \leq Q \leq 1 \). As \( \beta_1 \) must be a measure of the discriminative power of the LST-based term in the segmentation method, it can be chosen to be

\[
\beta_1 = 1 - Q = 1 - \int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx. \tag{26}
\]

This function constitutes a valid metric for the distributions used in this paper, as demonstrated in Appendix A. If the mean tensors over regions \( \Omega_1 \) and \( \Omega_2 \), \( T_1 \) and \( T_2 \), respectively, are substantially dissimilar, the overlapping between both distributions will be small, and therefore \( \beta_1 \) will be high, favouring the LST-based term.

With respect to \( \beta_2 \), and in order to reflect the discriminative power of the components term, the Euclidean distance between the mean values over regions \( \Omega_1 \) and \( \Omega_2 \), \( \mu_1 \) and \( \mu_2 \), respectively, can be employed:

\[
\beta_2 = d_{L}(\mu_1, \mu_2). \tag{27}
\]

If the image components are normalized to 1, for gray level images \( \beta_2 \) will also be naturally normalized (0 \( \leq \beta_2 \leq 1 \)). For the color case, a straightforward normalizing factor is applied.

The adaptive calculation of \( \beta_1 \) and \( \beta_2 \) implies a very small increase in the computation time of the segmentation method. This is due to the fact that they are computed from the mean tensors and image values, \( T_1, T_2, \mu_1 \) and \( \mu_2 \), which also need to be computed for the level set evolution (Eq. (24)). Finally, it is worth noticing that the combined adaptive segmentation proposed in this section can also take into account other texture extractors than the LST, as Gabor filters, for example [9]. In that case, however, an appropriate distance measure must be employed for the particular feature space, and a suitable choice for the balancing parameter \( \beta_1 \).

6. Experimental results

In this section, the performance of the proposed method is tested and validated. Furthermore, we compare our approach with several related as well as other non-related approaches described in the literature. As it is rare to find quantitative results and comparisons on different segmentation methods in the literature, an effort has been made in order to provide objective and quantitative results and extensive work has been done on synthetic compositions of gray-level and color images taken from public texture databases. Additionally, results will be shown on real images in order to validate our methods.

As for gray-level images, textures were taken from the Bonn’s University data set of Brodatz-like textures [80,81] (see Fig. 1(a), (b) for examples). Each test image was created by combining the textures present in two images into a single image with a mask, as can be seen in Fig. 1. This way, 100 images were created. An example of such test images can be seen in Fig. 1(e). For the color case, the Columbia-Utrecht reflectance and texture database (CUReT) was employed [82]. As in the previous case, a test set was constructed with 100 test images, using the same mask as before. A sample5 color test image is shown in Fig. 1(f). For all the experiments made, the smoothing term in the

---

5The whole synthetic data sets used are available at http://www.lpi.tel.uva.es/~rluigar/LSTtextures.
level set evolution equation (parameter \( n \) in Eq. (11)) has been set to \( n = 1 \).

Throughout the experimental work with synthetic images, the performance of the segmentation process was measured in terms of a success score, defined as follows:

\[
S = \frac{\text{number of pixels correctly classified}}{\text{total number of pixels}}. \tag{28}
\]

Clearly, \( 0 \leq S \leq 1 \). It is also worth mentioning that, for all the contour-oriented experiments made, small circular contours all over the image were taken as the initial contour (see Fig. 1(g), (h)).

The proposed adaptive combined segmentation algorithm (this approach will be referred to as AC) was first compared with some recent related methods which also make use of the LST in a level set framework. Specifically, the tensor processing of the LST, or its related architectures, introduced in [40] and revisited in Section 4, were used for comparison. The classical LST was employed for the gray-level images and the CLST was chosen for the color cases, as being the variants producing the best results for the employed data sets. In Fig. 2, we compare the results for these indicated approaches with those of the AC approach and the non-adaptive counterpart, which uses employs \( \beta_1 = \beta_2 \). In order to provide insight on both the robustness and accuracy of the different algorithms, results are presented in terms of the 25, 50, 75 and 95 percentiles of the success score, \( S \). In other words, we show how low parameter \( S \) must be so that each percentage of the test images gets a success score below this value. For example, for the 75 percentile, we indicate the value of \( S \) such that 75% of the images get a success score lower than \( S \). The perfect algorithm would have all these percentiles equal to one, and the lower these values, the worse.

The analysis of the obtained results allows us to discuss a number of interesting issues. First, results indicate that, for both the gray-level and the color data sets, employing an adaptive scheme for the selection of \( \beta_1 \) and \( \beta_2 \) leads to an improvement in the segmentation performance. This improvement is more important in the case of gray-level images, and mostly affects the robustness of the segmentation results (values of the 25 and 50 percentiles). It must also be noticed that, since a blind initialization (small circular contours all over the image) has been employed for these experiments, the first few iterations of the segmentation process are performed using a fixed value of \( \beta_1 = \beta_2 \), as the estimation of both weights are based on the present state of the segmentation process. Therefore, the advantages of the AC segmentation could probably be further exploited if a prior coarse segmentation step were used for initialization purposes.
Fig. 2 also shows that the combined segmentation approach presented in this paper outperforms the full tensor processing of the LST or its variants presented in [40]. For the gray-level images, the classical LST shows a remarkable performance, but is not as robust as the AC approach. Even though the difference is not as important for the color case, the adaptive approach shows a slightly better performance than the non-adaptive approach for all the percentile values. Probably, this better performance is due to the fact that the combined segmentation method employs a natural representation for the texture information, i.e. the structure tensor and its intrinsic distances, and so does with the components’ information, i.e. the Euclidean distance. Previous approaches either do not include any gray-level or color information, or denaturalize the information they make use of by merging into a single tensor representation heterogeneous pieces of information.

Besides the related contour-oriented approaches for the segmentation of textures based on the LST, a considerable number of other studies have been carried out in the literature [9–12,55,83,84]. However, most of them are not easily comparable with the approach presented in this paper, because they are either supervised methods, pixelwise classification approaches or have been designed for multiple classes, as opposed to being a two-class, unsupervised and contour-oriented method, as it is our case. Nevertheless, we have performed a quantitative comparison between the proposed method and some state-of-the-art algorithms for texture segmentation.

First, segmentation based on K-means and Fuzzy C-means clustering algorithms was performed, where the image components plus the LST components were taken as features (1/C2 feature vector for gray-level images and 1/C26 feature vector for color images). As the segmentation obtained with clustering algorithms will have a spotty appearance, these procedures are followed by a majority voting operation in order to achieve smoother results.

Second, we tested our method against the widely known texture segmentation method proposed by Bouman and Shapiro [83]. Their innovative segmentation approach is inherently supervised, as it requires a prior training step for the estimation of the model parameters. Therefore, and in order to make comparisons possible, we have recreated an unsupervised version, where a K-means clustering algorithm is applied first in order to construct the training set of pixels belonging to both classes (see Appendix B for details).

Finally, a comparison has been also made with the segmentation method by Hoang et al. presented in [55]. In their work, the authors perform the color texture measurement in the wavelength-Fourier domain. In practice, the color image is represented by means of the Gaussian color model, and then a set of Gabor filters is applied. The resulting feature image is, after been smoothed, clustered by means
of a K-means algorithm. In the original paper, the final number of segmented regions that are obtained is not fixed, and the K-means algorithm is followed by a region merging process in order to combine similar clusters. However, we have implemented a two-class segmentation method following this algorithm, and therefore no region merging is applied.

The results corresponding to the comparison of all these approaches are graphically shown in Fig. 3, for the gray-level and color data sets. As before, we provide the percentile values for the success score $S$.

Comparison with other well-known segmentation approaches in the literature also demonstrates that our method is fully competitive, as seen in Fig. 3.

K-means or Fuzzy C-means clustering on the LST components are not as robust as our method, although they can be extremely precise in the best-case scenarios, specially in the color case. With regard to Bouman’s segmentation method, results

![CUMULATIVE DISTRIBUTION FUNCTIONS OF S AND AREA UNDER THE CURVE](image)

Fig. 4. Cumulative distribution of $S$ and area under the curve (AUC).

<table>
<thead>
<tr>
<th>Table 1</th>
<th>AUC comparison for the different segmentation methods for gray images</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segmentation method</td>
<td>AUC</td>
</tr>
<tr>
<td>Adaptive $\beta_1$ and $\beta_2$</td>
<td>0.3230</td>
</tr>
<tr>
<td>Non-adaptive $\beta_1$ and $\beta_2$</td>
<td>0.3660</td>
</tr>
<tr>
<td>LST (Local structure tensor)</td>
<td>0.3240</td>
</tr>
<tr>
<td>Bouman’s method</td>
<td>0.3545</td>
</tr>
<tr>
<td>Hoang’s method</td>
<td>0.6610</td>
</tr>
<tr>
<td>K-Means</td>
<td>0.5335</td>
</tr>
<tr>
<td>Fuzzy C-Means</td>
<td>0.5315</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2</th>
<th>AUC comparison for the different segmentation methods for color images</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segmentation method</td>
<td>AUC</td>
</tr>
<tr>
<td>Adaptive $\beta_1$ and $\beta_2$</td>
<td>0.2100</td>
</tr>
<tr>
<td>Non-adaptive $\beta_1$ and $\beta_2$</td>
<td>0.2230</td>
</tr>
<tr>
<td>CLST (Compact local structure tensor)</td>
<td>0.3185</td>
</tr>
<tr>
<td>Bouman’s method</td>
<td>0.3410</td>
</tr>
<tr>
<td>Hoang’s method</td>
<td>0.3857</td>
</tr>
<tr>
<td>K-Means</td>
<td>0.2880</td>
</tr>
<tr>
<td>Fuzzy C-Means</td>
<td>0.2520</td>
</tr>
</tbody>
</table>
Fig. 5. Box-and-Whisker plots of the segmentation results for all tested methods, for the synthetic gray-level (a) and color (b) data sets. The boxes show the lower quartile, median and upper quartile values, whereas the whiskers (in black) show the extent of the rest of the data. Outliers are data with values beyond the ends of the whiskers.
show a remarkable accuracy in the best-case scenarios. The proposed AC segmentation method shows a slightly better overall performance for the gray-level case, and a much better robustness for the color case. Finally, it can be seen that Hoang’s method, based on the use of Gabor filters and the use of the Gaussian color model for the color case, is not able to obtain results comparable with the proposed approach, based on the use of the local structure tensor for the representation of the texture information. Probably, this remarkable difference is due to the fact that, for the employed data sets, the set of Gabor filters tends to highlight the boundaries between different textures instead of highlighting the difference between the textures themselves.

The different algorithms tested have been compared in terms of the percentiles of the success score, $S$. It is worth mentioning, however, that this quality parameter tends to produce a bias in contour-oriented algorithms, due to the smoothing applied to the contour. This smoothing effect can be appreciated, for example, in the well-known real image shown in Fig. 6. For this image, the AC method achieves a remarkable segmentation accuracy but, if we focus on the segmenting contour between the front or back legs, it can be noticed that the smoothing prevents the contour to be completely precise in this area, as opposed to Bouman’s segmentation method, for example, which is pixel-oriented. This negative bias helps explain why the AC method is not able to produce the extremely high values of $S$ in the best-case scenarios that pixel-oriented methods can get.

Having employed the different percentile values of parameter $S$ in order to quantitative analyze the segmentation results of the different approaches studied, it is helpful to derive an aggregate measure in order to get a final and fast overall comparison that helps corroborate our analysis. For this purpose, let us consider the cumulative distribution function of the success score over the data set, $F(S)$. Ideally, for a hypothetical perfect segmentation method this function would be a step function as shown in Fig. 5. For a real segmentation method, the closer $F(S)$ is to the ideal step function, the better the performance of the system will be (see Fig. 5, where Method 1 is the ideally perfect system, Method 2 performs better that Method 3 and so on). This similarity can be measured by means of the area under the curve (AUC) of the obtained $F(S)$ for each segmentation method. The lower the AUC, the better the performance of the segmentation method (in Fig. 4, the AUC is colored for Method 2).

We have employed the described aggregate parameter to compare the segmentation methods tested. In Tables 1 and 2, we show the corresponding results for gray and color images, respectively. As can be seen from the tables, the AC segmentation method proposed in this paper achieves a better overall performance in terms of the aggregated AUC indicator both for gray and color images, which confirms the previous analysis based on the percentile values of $S$.

Finally, and in order to provide a complete description of the comparison results for all the segmentation methods, in Fig. 5 we show Box-and-Whisker plots of the results obtained for all methods, both for gray-level and color images. These plots are consistent with the analysis of the results previously made from the percentile values of $S$ and the aggregated AUC indicator.

Fig. 6. Segmentation results with some real-world gray-level images.
Even though synthetic images allow us to precisely and quantitatively determine the goodness of a segmentation method and enable a direct comparison with related approaches, the use of real images can also provide insight about the segmentation performance. In Fig. 6, the segmentation results on two well known real-world gray-level images can be seen.

In Fig. 7, we show a comparison of the segmentation results using the ST, EST, CST and
AC approaches. As can be seen, the three first approaches have trouble with certain types of images because they do not make a proper use of the information present in the image components. In Fig. 8, we compare our result shown in Fig. 6 (upper image) with different results obtained using the K-means clustering method described before, as well as Bouman’s segmentation method. Results illustrate some limitations of these two approaches.

As for color real-world images, several examples of the segmentation results obtained with the AC approach are shown in Fig. 9, showing that our scheme is able to successfully deal with the unsupervised segmentation of very heterogeneous images.

7. Concluding remarks

This work has presented a combined approach for the segmentation of textured gray or color images. Instead of considering the texture or the components separately as discriminant features or merging them into a unique representation form (either vector or tensor), texture and image components are jointly considered respecting their most natural representation form, i.e. the structure tensor and a vector, respectively. Based on these representation schemes, an energy term was defined considering intrinsic distance measures for both, and minimized within a level set framework. The extensive experimental work showed that the proposed scheme is able to yield excellent results, even outperforming the ones obtained using a tensor processing of the structure tensor and its advanced architectures that were revisited here.

A few words about limitations of the algorithm are of importance; first, it must be noticed that, since the method is contour oriented, regularization constraints are needed in order to achieve smooth results. These constraints, however, prevent the segmentation from being completely accurate in the presence of sharp corners. This has been highlighted in some of the experiments described in the previous section. Second, we have assumed in the paper that the texture characterization by means of the LST is appropriate in terms of scale and our attention has been focused on finding the balance of the energy terms proposed. As future work, the ideas of Brox and Weickert concerning the notion of scale [85] will let us design a fully automated segmentation process.

Acknowledgments

The authors acknowledge the Imavis project numbered HPMT-CT-2000-00040 working within
the framework of the Marie Curie Fellowship Training Sites Programme, the Comisión Interministerial de Ciencia y Tecnología, Spain, for research Grant TEC2004-06647-C03-01, the Fondo de Investigaciones Sanitarias for Grant PI-041483 and the Junta de Castilla y León for grant VA075A05; the European Commission is also acknowledged for the funds associated to the Network of Excellence SIMILAR (FP6-507609).

We thank Charles A. Bouman for having his segmentation methods available to the scientific community.

Appendix A

In this Appendix we provide the demonstration of \( \beta_1 \), given by Eq. (26), to be a valid metric. Let us recall the expression:

\[
\beta_1(p_1, p_2) = 1 - Q = 1 - \int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx,
\]

(A.1)

where \( p_1(x), p_2(x) \) are the resulting probability density functions when considering two Gaussians distributions with zero mean and the describing tensors \( T_1 \) and \( T_2 \) as covariance matrices, respectively. In order to prove that \( \beta_1 \) is a valid metric, we must ensure that it satisfies the four necessary conditions:

- **Non-negativity.** To begin with, it is clear that

\[
\int_{-\infty}^{\infty} p_1(x) \, dx = \int_{-\infty}^{\infty} p_2(x) \, dx = 1.
\]

Additionally, \( \min(p_1(x), p_2(x)) \leq p_1(x) \) and \( \min(p_1(x), p_2(x)) \leq p_2(x) \). Therefore,

\[
\int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx \leq \int_{-\infty}^{\infty} p_1(x) \, dx = 1,
\]

\[
\int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx \leq \int_{-\infty}^{\infty} p_2(x) \, dx = 1,
\]

so,

\[
\int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx \leq 1
\]

which leads to

\[
\beta_1(p_1, p_2) = 1 - \int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx \geq 0.
\]

- \( \beta_1(p_1, p_2) = 0 \) if and only if \( p_1(x) = p_2(x) \). If \( p_1(x) = p_2(x) \), then

\[
\int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx = \int_{-\infty}^{\infty} p_1(x) \, dx = 1
\]

and so \( \beta_1(p_1, p_2) = 0 \). On the other hand, if \( p_1(x) \neq p_2(x) \), then

\[
\int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx < \int_{-\infty}^{\infty} p_1(x) \, dx = 1
\]

and so \( \beta_1(p_1, p_2) > 0 \). Notice that this may not be valid for distributions with finite support.

- **Symmetry.** As the \( \min() \) operator is symmetric, it is clear that

\[
\int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx = \int_{-\infty}^{\infty} \min(p_2(x), p_1(x)) \, dx
\]

and therefore

\[
\beta_1(p_1, p_2) = \beta_1(p_2, p_1).
\]

- **Triangle inequality.** We need to prove that\(^6\)

\[
\beta_1(p_1, p_3) \leq \beta_1(p_1, p_2) + \beta_1(p_2, p_3)
\]

which equals to

\[
1 - Q(p_1, p_3) \leq 1 - Q(p_1, p_2) + 1 - Q(p_2, p_3)
\]

and so

\[
Q(p_1, p_2) + Q(p_2, p_3) - Q(p_1, p_3) \leq 1.
\]

Let us name

\[
\Gamma = Q(p_1, p_2) + Q(p_2, p_3) - Q(p_1, p_3)
\]

\[
\Gamma = \int_{-\infty}^{\infty} \min(p_1(x), p_2(x)) \, dx + \int_{-\infty}^{\infty} \min(p_2(x), p_3(x)) \, dx
\]

\[
- \int_{-\infty}^{\infty} \min(p_1(x), p_3(x)) \, dx
\]

\[
= \int_{-\infty}^{\infty} \min[p_1(x), p_2(x)] + \min(p_2(x), p_3(x)) - \min(p_1(x), p_3(x)] \, dx.
\]

The domain of integration can be divided in 6 different regions, according to the situations that can be encountered if an ordering of the

\(^6\)We assume that the three distributions are different. If they are either pairwise equal or the three of them coincide, it is straightforward to see that the inequality holds.
distributions is carried out in terms of magnitude:

\[
\begin{array}{ccccccc}
\text{Region} & A_1 & A_2 & A_3 & A_4 & A_5 & A_6 \\
\hline
\text{Increasing value} & p_2(x) & p_2(x) & p_1(x) & p_1(x) & p_1(x) & p_1(x) \\
& p_2(x) & p_1(x) & p_1(x) & p_1(x) & p_1(x) & p_1(x) \\
& p_1(x) & p_2(x) & p_2(x) & p_2(x) & p_2(x) & p_2(x) \\
\end{array}
\]

\[A_1 \Rightarrow \min(p_1(x), p_2(x)) + \min(p_2(x), p_3(x)) \]
\[- \min(p_1(x), p_3(x))
\[= p_1(x) + p_2(x) - p_1(x) = p_2(x),
\]

\[A_2 \Rightarrow \min(p_1(x), p_2(x)) + \min(p_2(x), p_3(x)) \]
\[- \min(p_1(x), p_3(x))
\[= p_1(x) + p_3(x) - p_1(x) = p_3(x),
\]

\[A_3 \Rightarrow \min(p_1(x), p_2(x)) + \min(p_2(x), p_3(x)) \]
\[- \min(p_1(x), p_3(x))
\[= p_2(x) + p_2(x) - p_1(x) = 2p_2(x) - p_1(x),
\]

\[A_4 \Rightarrow \min(p_1(x), p_3(x)) + \min(p_2(x), p_3(x)) \]
\[- \min(p_1(x), p_3(x))
\[= p_1(x) + p_3(x) - p_3(x) = p_1(x),
\]

\[A_5 \Rightarrow \min(p_1(x), p_2(x)) + \min(p_2(x), p_3(x)) \]
\[- \min(p_1(x), p_3(x))
\[= p_2(x) + p_2(x) - p_3(x) = 2p_2(x) - p_3(x),
\]

\[A_6 \Rightarrow \min(p_1(x), p_2(x)) + \min(p_2(x), p_3(x)) \]
\[- \min(p_1(x), p_3(x))
\[= p_2(x) + p_3(x) - p_3(x) = p_2(x).
\]

We can now split the integral in the different parts:

\[\Gamma = \int_{A_1} p_2(x) \, dx + \int_{A_2} p_3(x) \, dx \]
\[+ \int_{A_3} [2p_2(x) - p_1(x)] \, dx + \int_{A_4} p_1(x) \, dx \]
\[+ \int_{A_5} [2p_2(x) - p_3(x)] \, dx + \int_{A_6} p_2(x) \, dx.
\]

Now, in \(A_2\), \(p_3(x) \leq p_2(x)\). In \(A_3\), \(2p_2(x) - p_1(x) = p_2(x) + [p_2(x) - p_1(x)]\), and since \(p_3(x) \leq p_1(x)\) then \(2p_2(x) - p_1(x) \leq p_2(x)\). The same reasoning applies for \(A_5\), so \(2p_2(x) - p_3(x) \leq p_2(x)\). Finally, in \(A_4\), \(p_1(x) \leq p_2(x)\). Therefore, we can write

\[
\Gamma \leq \int_{A_1} p_2(x) \, dx + \int_{A_2} p_3(x) \, dx + \int_{A_3} p_1(x) \, dx \\
+ \int_{A_4} p_2(x) \, dx + \int_{A_5} p_2(x) \, dx \\
+ \int_{A_6} p_2(x) \, dx = \int_{-\infty}^{\infty} p_2(x) \, dx = 1.
\]

**Appendix B**

Bouman’s multiscale segmentation method [83] is based on the estimation of the model parameters from a training set for each class. Therefore, it is inherently supervised. As the AC method proposed in this paper is an unsupervised segmentation approach, and in order to perform a fair comparison, we have constructed an unsupervised version of this segmentation method. To do so, representative regions for each class must be extracted from the image, in an unsupervised manner. We employ a K-means algorithm to this end as it has shown to provide competitive results (recall Section 6). This K-means algorithm, however, is applied on the diffused HSI image components instead of on the original image because of two reasons. First, smoother results are obtained and it will be possible to extract larger regions for the training step of Bouman’s method. Second, as the same nonlinear diffusion process as in our AC approach is applied, both methods employ exactly the same diffusion step, and consequently the comparison is as fair as possible.

Once the K-means algorithm has been applied and a two-class segmentation has been obtained, a fixed-size square is extracted from each class. If this is not possible because no connected region obtained is large enough so as to enclose such estimation window, the size of it is reduced. The estimation windows so located are used on the original components to carry out the training step for the estimation of the model parameters.\(^8\)

Finally, the SMAP segmentation algorithm is applied on the original images (HSI representation is employed) using the estimated parameters.

\(^8\)Note that, although the constructed version of Bouman’s algorithm is mostly unsupervised, as it was the case for the AC segmentation method proposed in this paper, some parameters still needed to be tuned, namely the amount of nonlinear diffusion applied and the size of the estimation window.
References


