remote sensing

Article

Bayesian Inference for Post-Processing of Remote-Sensing Image Classification

Gilberto Camara [1,*], Renato Assunção [2], Alexandre Carvalho [3], Rolf Simoes [4], Felipe Souza [1], Felipe Carlos [1], Anielli Souza [1], Ana Rorato [1] and Ana Paula Dal'Asta [1]

1 National Institute for Space Research (INPE), Avenida dos Astronautas, 1758, Jardim da Granja, Sao Jose dos Campos 12227-010, SP, Brazil; felipe.carvalho@inpe.br (F.S.); felipe@openmsc.org (F.C.); anielli.souza@inpe.br (A.S.); ana.rorato@inpe.br (A.R.); ana.dalasta@inpe.br (A.P.D.)
2 Computer Science Department, Federal University of Minas Gerais (UFMG), Av. Pres. Antônio Carlos, 6627, Belo Horizonte 31270-901, MG, Brazil; rassuncao@esri.com
3 National Institute for Applied Economics Research (IPEA), SBS, Quadra 1 Bloco J, Brasília 70076-900, DF, Brazil; alexandre.ywata@ipea.gov.br
4 Open Geo Hub (OGH), Waldeck Pyrmontlaan 14, 6865 HK Doorwerth, The Netherlands; rolf.simoes@opengeohub.org
* Correspondence: gilberto.camara@inpe.br

Abstract: A key component of remote-sensing image analysis is image classification, which aims to categorize images into different classes using machine-learning methods. In many applications, machine-learning classifiers assign class probabilities to each pixel. These class probabilities serve as input for post-processing techniques that aim to improve the results of machine-learning algorithms. This paper proposes a new post-processing algorithm based on an empirical Bayes approach. We employ non-isotropic neighborhood definitions to capture the impact of borders between land classes in the statistical model. By incorporating expert knowledge, the algorithm improves the consistency of the classified map. This technique has proven its efficacy for large-scale data processing using image time-series analysis. The proposed method is a key component of a time-first, space-based approach for big Earth-observation data processing. It is available as open source as part of the R package `sits`.

Keywords: Bayesian inference; post-processing; image classification; machine learning

check for updates

Citation: Camara, G.; Assunção, R.; Carvalho, A.; Simoes, R.; Souza, F.; Carlos, F.; Souza, A.; Rorato, A.; Dal'Asta, A.P. Bayesian Inference for Post-Processing of Remote-Sensing Image Classification. *Remote Sens.* **2024**, *16*, 4572. https://doi.org/10.3390/rs16234572

Academic Editors: Won-Ho Nam and Jingxiong Zhang

Received: 1 September 2024
Revised: 26 November 2024
Accepted: 26 November 2024
Published: 6 December 2024

1. Introduction

Classifying land use and land cover in remote-sensing images primarily relies on machine-learning techniques employing algorithms like random forests [1] and deep learning [2,3]. Many of these algorithms are pixel-based, using spectral and temporal information associated to each individual location. Training such models requires samples built by choosing "pure" pixels corresponding to the desired output classes. Their output is a set of class probabilities, one for each pixel. Post-processing algorithms use these probability maps to improve the initial results.

The diversity of responses in ground targets presents a significant challenge for image classification. Many pixels in remote-sensing images include mixed spectral signatures from multiple land-cover types, making it difficult to classify them accurately. Ground samples that train classification algorithms often fail to capture the full range of natural variation within different land classes. As a result, machine-learning methods frequently produce inaccurate classifications. To address this, post-processing techniques are crucial for refining initial classification results [4,5]. By reducing outliers and noise, post-processing improves the final output's accuracy and interpretability. These techniques enhance pixel-based classifications, producing more accurate and uniform maps [6].

Image classification post-processing techniques include methods such as Gaussian, semi-global, bilateral, and edge-aware filtering [4], modal filters [7], and co-occurrence matrices [6]. Border pixels pose challenges to all these techniques. Gaussian, bilateral, and semi-global smoothing methods rely on the assumption of constant local variance for each class. These methods presume that class probabilities change gradually around pixels and that the spatial distribution of these probabilities is isotropic within a pixel's vicinity. In contrast, modal filters operate directly on classified maps rather than classifier probabilities. Despite their differences, we argue that these methods fail to adequately capture the spatial variability inherent in land-cover classes, particularly in scenarios involving mixed pixels and border discontinuities. This highlights the need for post-processing techniques that can better address these complexities.

This paper introduces a novel post-processing algorithm based on an empirical Bayes approach to enhance image classification outcomes from machine-learning methods. The proposed method is particularly applicable to satellite image time series analysis. Calibrated time series provide consistent measurements of Earth locations over time, enabling the detection of significant land use and cover changes [8]. Large image collections enable measuring subtle ecosystem changes [9,10]. Time series analysis thus offers innovative solutions to global challenges, including climate change, biodiversity conservation, and sustainable agriculture [11].

Our approach incorporates anisotropic neighborhood definitions to account for discontinuities between land classes. Additionally, expert knowledge is used to enhance the consistency of the generated maps. The paper provides a comparative evaluation of Bayesian smoothing against established methods such as Gaussian smoothing [4] and bilateral smoothing [12]. This algorithm has demonstrated its utility in recent studies applying image time series to land use classification [13–17]. These studies employed a version of the Bayesian smoothing algorithm which is available in the R package `sits`, developed by the authors [18]. The practical application of this algorithm in solving real-world problems has been instrumental in refining its methodology. Despite its previous usage, this paper is the first to provide a comprehensive technical description and a comparative evaluation of the algorithm.

2. Methods

2.1. The Land Classification Problem

Land classification produces a categorical map where each category represents a distinct land type, such as forest, grassland, wetland, or cropland. Thus, land classification aims to partition space into discrete regions, each corresponding to a specific land use or cover type. Borders between different land classes have sharp transitions which are hard for algorithms to capture properly. Post-processing techniques address these challenges by refining the output of machine-learning models; they update each pixel's probabilities to better capture its likelihood of belonging to the target classes.

We assume a set of n spatial locations or pixels $S = \{\mathbf{s}_1, \ldots, \mathbf{s}_n\}$, each with an associated d-dimensional feature vector $\mathbf{z}_1, \ldots, \mathbf{z}_n$ with values in $\mathcal{Z} \subset \mathbb{R}^d$ and a set of m land classes $K = \{1, \ldots, m\}$. For image classification, the set of spatial locations S will correspond to a 2-D set of adjacent pixels. The d-dimensional vector includes attributes such as bands and indexes; for image time series, it also includes the temporal values of these attributes [19]. For example, consider a data cube consisting of a Sentinel-2 tile at a 10-m resolution of 10,980 $\times$ 10,980 pixels, with 24-time instances and ten bands, where we expect to identify ten classes. Set S will have 120 million pixels, each associated with a 240-dimensional feature vector, and the size of m is ten.

For each pixel $s_i \in S$, machine-learning algorithms produce a m-dimensional vector $\mathbf{p}_i = (p_{i,1}, \ldots, p_{i,m})$ with $0 \leq p_{i,k} \leq 1$ and $\sum_k p_{i,k} = 1 \; \forall k = 1, \ldots, m$. The value $p_{i,k}$ is the empirically estimated probability that the i-th pixel belongs to the k-th class. The probability $p_{i,k}$ is subject to noise, outliers, and classification errors. To get a classified map, the highest probability among the available options determines the class assigned to the pixel. Hence, the land classification function can be expressed as the composition $f \circ g$ of two functions, f going from $(\mathbf{s}_i, \mathbf{z}_i)$ to the output vector $\mathbf{p}_i$ provided by the machine-learning method and then g from this output probability vector to a class k_i in $K = \{1, \ldots, m\}$

$$\begin{aligned} f\colon S \times \mathcal{X} &\longrightarrow [0,1]^m & g\colon [0,1]^m &\longrightarrow K \\ (\mathbf{s}_i, \mathbf{z}_i) &\longmapsto \mathbf{p}_i & \mathbf{p}_i &\longmapsto k_i \end{aligned} \tag{1}$$

For pixel-based classification methods discussed in this paper, the spatial dimension is not explicitly used by the classification function f. The reason for this is that f is obtained from a training dataset $T = \{\mathbf{s}_{t_1}, \ldots, \mathbf{s}_{t_s}\} \subset S$ with $s \ll n$. Each element $\mathbf{s}_{t_i} \in T$ is coupled with one single label $k_{t_i} \in K$ representing the class assigned by an expert, even when the pixel has mixed classes on the ground. This training dataset comprises non-contiguous locations; therefore, no spatial context is available to train f. The only resource to fit f is the assigned label k and the features $\mathbf{z}$ measured at each location in the training dataset T.

2.2. Conversion from Probabilities to Logits

The class probability values $p_{i,k}$, which are the outputs of the machine-learning classifier, are converted to log-odds values to make the inference more tractable. The logit function converts probability values between 0 to 1 to values from $-\infty$ to ∞. The conversion from probabilities to logit values is expressed by:

$$x_{i,k} = \ln\left(\frac{p_{i,k}}{1 - p_{i,k}}\right) \tag{2}$$

Conversion from probabilities to logits helps to support our assumption of a normal distribution for our data. Confidence in pixel classification increases with logit. However, there are situations, such as border or mixed pixels, where the logit of different classes is similar in magnitude. These are cases of low confidence in the classification result. The post-classification smoothing method borrows strength from the neighbors to assess and correct these cases.

2.3. Bayesian Estimation

Our post-processing method is based on the Bayesian approach to data analysis, which interprets probability as a measure of belief in an event. This approach begins with an initial set of beliefs, known as the *prior probability*, derived from previous experiments or expert knowledge. Bayesian statistics then combines this prior knowledge with new data to update and refine these beliefs. This is carried out using Bayes' theorem, a mathematical framework for revising probabilities as new evidence becomes available.

Unlike classical (frequentist) methods, which treat parameters as fixed but unknown values, the Bayesian approach considers parameters as random variables with associated probability distributions. It generates a *posterior distribution* by integrating prior distributions with observed data through a *likelihood function*. This posterior distribution represents the updated beliefs about the parameters, offering a comprehensive framework for inference and decision-making.

We assume that the class probability $p_{i,k}$ estimated by the pixel-based machine-learning method is a noisy version of underlying probability $\pi_{i,k} > 0$ we want to learn. If the machine-learning method had perfect accuracy, we should have $p_{i,k} = \pi_{i,k}$. In practice, they are not equal due to estimation errors inherent in any machine-learning method.

Using Bayesian estimation for the post-processing takes three steps. First, define the corresponding logit for the true and unknown $\pi_{i,k}$ probabilities: $\mu_{i,k} = \ln(\pi_{i,k}/(1 - \pi_{i,k}))$. Then, assume the noisy data version $x_{i,k}$ to be randomly distributed around $\mu_{i,k}$ according to a Gaussian distribution:

$$(x_{i,k}|\mu_{i,k}) \sim N(\mu_{i,k}, \sigma_k^2) \tag{3}$$

where σ_k^2 is the variance associated with the k-th class. The distribution in (3) is called the *likelihood* function. Finally, suppose that the *prior probability* of the uncertainty around $\mu_{i,k}$ is also a Gaussian distribution:

$$\mu_{i,k} \sim N(m_{i,k}, s_{i,k}^2)\,. \tag{4}$$

Applying Bayes theorem [20] in the case of Gaussian distributions for the likelihood in (3) and the prior distribution in (4) results in a Gaussian *posterior distribution* of the unknown $\mu_{i,k}$ after observing $x_{i,k}$:

$$(\mu_{i,k}|x_{i,k}) \sim \sum N\left(\frac{m_{i,k}\sigma_k^2 + x_{i,k}s_{i,k}^2}{\sigma_k^2 + s_{i,k}^2}, \left(\frac{1}{\sigma_k^2} + \frac{1}{s_{i,k}^2}\right)^{-1}\right) \tag{5}$$

The posterior distribution represents the updated uncertainty about the actual probability $\pi_{i,k}$ of the pixel i belonging to class k after we have observed the noisy $p_{i,k}$. The posterior mean is a weighted average between the observed logit value $x_{i,k}$ and the prior logit mean $m_{i,k}$. When the prior logit variance $s_{i,k}^2$ is high, the algorithm assigns more weight to the observed value $x_{i,k}$. Conversely, as the variance σ_k^2 increases, the update assigns more weight to the prior mean $m_{i,k}$.

In the above equation, the value of $x_{i,k}$ is known. To complete the calculation, one needs to work out the values of the likelihood variance σ_k^2 and the mean $m_{i,k}$ and variance $s_{i,k}^2$ of the prior distribution. However, in the case of large data sets, it is hard to determine adequate values for these parameters. These challenges led the authors to adopt an empirical Bayes (EB) estimate, as described below.

2.4. Empirical Bayes Method with Anisotropic Neighborhoods

In most cases, no prior information is available to specify $m_{i,k}$ and $s_{i,k}^2$. To address this, we adopt an empirical Bayes (EB) approach to estimate these parameters using the pixel's spatial neighborhood. While the traditional Bayesian approach relies on a predetermined prior distribution, the empirical Bayes approach allows the data to inform the prior before proceeding with analysis [21]. Specifically, the EB method estimates the prior distribution directly from the observed data, eliminating the need for external information or subjective beliefs. Once the prior is derived, it is combined with the observed data—similar to the standard Bayesian approach—to produce the posterior distribution.

In our method, the pixel's local neighborhood serves as the data source for the EB estimation. However, using a standard symmetrical adjacency metric defined solely by the distance between pixels is insufficient to solve border errors. In such cases, adjacent pixels often belong to different classes, leading to biased or misleading estimates. To overcome this, we employ a non-isotropic neighborhood definition that selectively considers pixels belonging to the same spatial process, ensuring more accurate prior estimates for class probabilities.

For example, consider a boundary pixel located between forest and grassland. Within a dense forest patch, class probability values have strong spatial autocorrelation. This correlation does not extend across the boundary into other land classes. To estimate the prior probability of the pixel being classified as a forest, it is crucial to focus exclusively on nearby pixels inside the forest. Including pixels from the grassland side of the border would misrepresent the underlying spatial process and lead to inaccurate priors. Our method uses a non-isotropic definition to address this challenge. This choice produces more reliable priors and improves classification accuracy in boundary regions.

Therefore, the EB estimates $m_{i,k}$ and $s_{i,k}^2$ use a specific neighborhood $\mathcal{N}_{i,k}$ for each class k and pixel i. We use an L-statistic to estimate $m_{i,k}$ and $s_{i,k}^2$ in our EB approach. Let $\alpha \in (0,1)$ and W_i be the set of w nearest neighbors of pixel i (excluding the i-th pixel itself). Also, let $\mathbb{F}_{i,k}$ be the empirical distribution of the $x_{j,k}$ for $j \in W_i$. Then, we take

$$\hat{m}_{i,k} = \frac{1}{1-\alpha}\int_{\alpha}^{\infty} \mathbb{F}_{i,k}^{-1}(s)ds\,, \tag{6}$$

to be the average of the largest $(1-\alpha)$-th fraction order statistics of the $p_{i,k}$ logit transformed observations [22]. Likewise, based on $(1-\alpha)$-th subset, we obtain an empirical estimate $\hat{s}^2_{i,k}$. The values of $\hat{m}_{i,k}$ and $\hat{s}^2_{i,k}$ are used in the Bayesian updating (cf Equation (5)). In this way, the estimated value for each class probability can be expressed as a weighted mean

$$E[\mu_{i,k}|x_{i,k}] = \left[\frac{\hat{s}^2_{i,k}}{\sigma^2_k + \hat{s}^2_{i,k}}\right] \times x_{i,k} + \left[\frac{\sigma^2_k}{\sigma^2_k + \hat{s}^2_{i,k}}\right] \times \hat{m}_{i,k}, \tag{7}$$

where:

- $x_{i,k}$ is the logit value for pixel i and class k;
- $\hat{m}_{i,k}$ is the average of logits for pixels of class k in the selected neighborhood of pixel i;
- $\hat{s}^2_{i,k}$ is the variance of logits for pixels of class k in the selected neighborhood of pixel i;
- σ^2_k is a user-derived hyperparameter which estimates the variance for class k, expressed in logits.

Several methods can be used to estimate the optimal anisotropic neighborhood $\mathcal{N}_{i,k}$ for each class k and pixel i. One approach involves selecting the probability values for class k that best conform to a Gaussian distribution. However, after extensive testing, we identified a faster and equally effective alternative: selecting the top half of the neighbors with the highest logits. This selection facilitates an efficient empirical estimation of the prior distribution. Specifically, probabilities indicating the presence of land-cover class k are retained, while those representing its absence are excluded from the prior distribution calculation. We recommend a sliding window of size 9×9 as the default choice. Considering the highest 50% logit values, estimates of the prior distribution will then use 41 values, which is a reasonable number for a Gaussian model.

2.5. Effect of the Hyperparameter

The parameter σ^2_k in Equation (7) controls the level of smoothness. It expresses confidence in the inherent variability of the distribution of class values k. If σ^2_k is zero, the value $E[\mu_{i,k}|x_{i,k}]$ will be equal to the pixel value $x_{i,k}$. Small values of σ^2_k relative to $\hat{s}^2_{i,k}$ indicate trust in the classifier's estimated probabilities for class k. Conversely, higher values of σ^2_k relative to $\hat{s}^2_{i,k}$ indicate low reliance on the pixel values and more confidence in the local averages.

Consider the following two-class example. Take a pixel with probability 0.4 (logit $x_{i,1} = -0.4054$) for class A and probability 0.6 (logit $x_{i,2} = 0.4054$) for class B. Without post-processing, the pixel will be labeled as class B. Take the local average to be 0.6 (logit $m_{i,1} = 0.4054$) for class A and 0.4 (logit $m_{i,2} = -0.4054$) for class B. This indicates a possible outlier classified as class B amid a set of pixels of class A. Suppose the logits' local variance is $\hat{s}^2_{i,1} = 5$ for class A and $\hat{s}^2_{i,2} = 10$ for class B. This difference is expected if class A's local variability is smaller than class B.

To complete the estimate, we need to set the parameter σ^2_k, representing our confidence in the intrinsic variability of the probabilities for each class. The intuitive sense is that classes with high variability are likelier to produce outliers than classes with low variability. Lower values of σ^2_k indicate higher confidence in the classifier outputs for class k. Conversely, if we consider that probabilities of class k can vary significantly, we will choose higher values of σ^2_k.

Should we consider the local variability high, we take σ^2_1 for class A and σ^2_2 for class B to be 10.0. In this case, the Bayesian estimated probability for class A is 0.52, and for class B is 0.48, and the pixel will be relabeled as class A. Assuming low local variability, we use smaller values setting σ^2 to 5.0 for classes A and B. The Bayesian probability estimate will be 0.48 for class A and 0.52 for class B. The original class will be kept. Thus, the result is sensitive to the subjective choice of the hyperparameter. In what follows, we show how to use the local logit variance to set the appropriate values of σ^2.

2.6. Alternatives to Bayesian Smoothing

To better understand the benefits of Bayesian smoothing, we compare it with two alternatives: Gaussian smoothing and bilateral smoothing. Gaussian smoothing is a low-pass filter in which the weights are based on the normal distribution [4]. Pixels near the center of the kernel have a higher weight, and the weight decreases for pixels further away from the center. This creates a blurring effect that is stronger at the center and weaker at the edges. Gaussian smoothing is expressed as

$$I'(x,y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} \frac{1}{2\pi\sigma^2} e^{-\frac{(x-i)^2+(y-j)^2}{2\sigma^2}} \cdot I(x-i, y-j) \quad (8)$$

In this equation, $I'(x,y)$ represents the smoothed image, $I(x,y)$ is the original image, (i,j) are the indices running over the kernel size, and where σ is the standard deviation of a Gaussian distribution.

Bilateral smoothing aims to reduce noise while preserving edges [12]. Bilateral filtering combines a spatial Gaussian filter with a range filter, which considers the similarity in intensity values between pixels. The bilateral filter has two main parameters: (a) the spatial parameter σ controls the spatial extent of the kernel; (b) the range parameter τ controls how much a pixel must differ in intensity for it to be considered different. Its mathematical expression is

$$I'(x,y) = \frac{1}{W(x,y)} \sum_{i=-k}^{k} \sum_{j=-k}^{k} I(x-i, y-j) \cdot G_\sigma(i,j) \cdot G_\tau(I(x-i, y-j) - I(x,y)) \quad (9)$$

where $I'(x,y)$ is the filtered image and $I(x,y)$ is the original image. $G_\sigma(i,j)$ is the spatial Gaussian function, which decreases with distance from the central pixel, controlled by the spatial standard deviation σ. $G_\tau(I(x+i, y+j) - I(x,y))$ is the range Gaussian function, which decreases with the intensity difference between the neighboring pixel and the central pixel controlled by the range standard deviation τ. $W(x,y)$ is a normalization factor defined as

$$W(x,y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} G_{\sigma_s}(i,j) \cdot G_{\sigma_r}(I(x-i, y-j) - I(x,y)) \quad (10)$$

3. Case Study

To present a realistic case study for the proposed algorithm's performance, we chose an application of land use and land-cover classification using image time series. We built a data cube covering the Sentinel-2 tile "20LMR", which covers a part of the state of Rondonia in Brazil. Rondonia is the most deforested state in the Brazilian Amazonia in terms of its area. As of 2024, 6.7 Mha (millions of hectares) of the original forests in Rondonia have been completely cleared.

For our case study, the data cube was built with 16-day intervals and ten spectral bands from data retrieved from Microsoft Planetary Computer, combining images from Sentinel-2A and Sentinel-2B satellites from 5 January 2022 to 23 December 2022. The spatial resolution is 20 m. The data cube size is 10 GB. Figure 1 shows a color composite of bands B02 in blue, B8A in green, and B11 in red for tile "20LMR" on 16 July 2022. Porto Velho (capital of the Rondonia state) is visible on the mid-left border of the image. The Madeira River crosses the image diagonally, separating an area with much deforestation from two protected areas (Cuniã Ecological Station and Cuniã Lake Extractivist Reserve).

The data cube follows the definition proposed by Simoes et al. [18]. It is built using images that have been geometrically and radiometrically calibrated to be comparable in time; these kinds of images are called "analysis-ready data" [23]. The essential properties of an EO data cube are: (a) it uses an image time series of analysis-ready data; (b) the spatial support is georeferenced; (c) temporal continuity is assured; (d) all spatiotemporal locations share the same set of attributes; and (e) there are no gaps or missing values in the spatiotemporal extent. Data cubes support using the available archives of Earth-observation data for times series machine learning classification [24,25].

Image time-series analysis explores the information available in data cubes. By observing the same location multiple times, satellites provide data on environmental changes and survey areas that are difficult to observe from the ground. Using time series, experts improve their understanding of ecological patterns and processes. Instead of selecting individual images from specific dates and comparing them, researchers track change continuously [8].

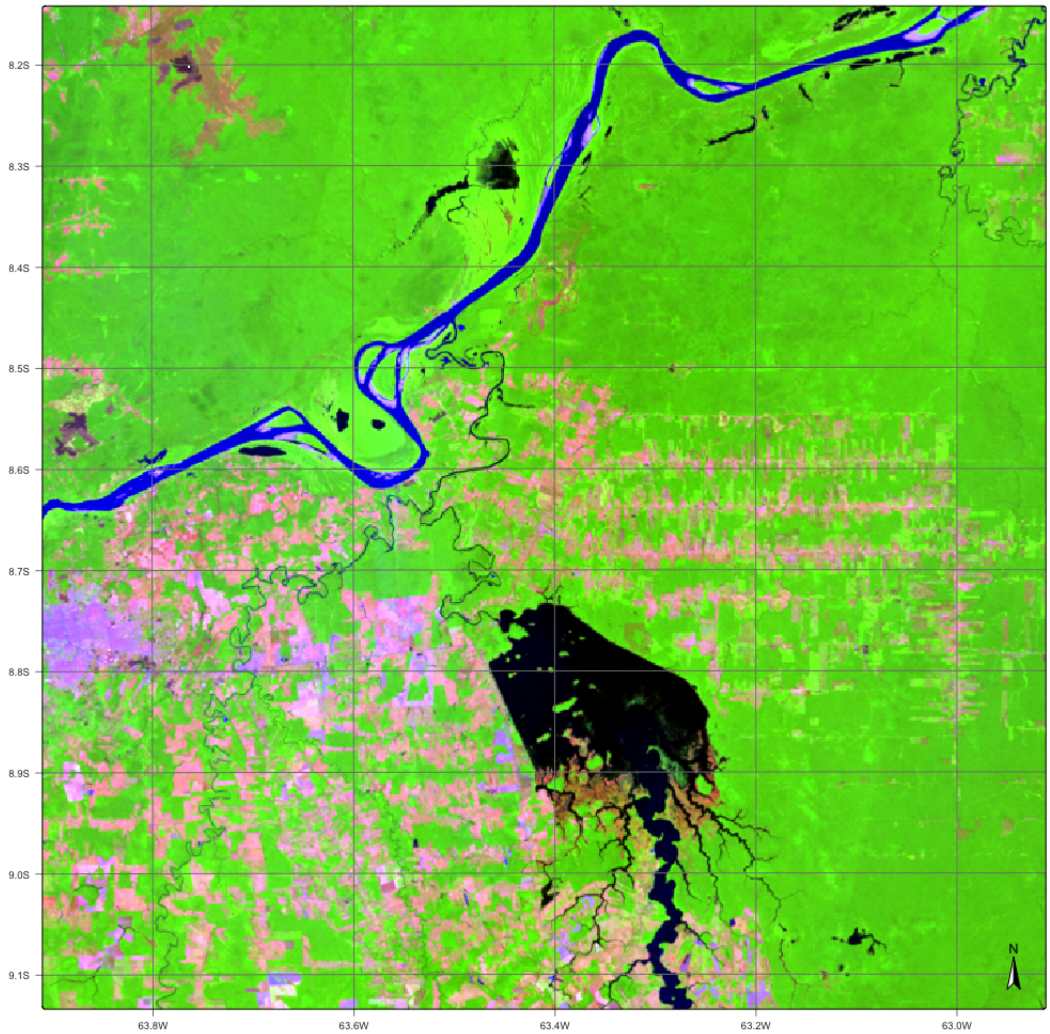

Figure 1. RGB color composite of tile 20MLR of Sentinel-2 data for date 16 July 2022 (source: authors).

The training data consists of 5796 samples for nine classes: (a) *Forest* for the stable natural tropical forest; (b) *Riparian Forest* for woody areas adjacent to rivers; (c) *Seasonally Flooded* for woody areas with are covered by water during the rainy season; (d) *Water* for lakes and rivers; (e) *Wetland* areas where water covers the soil all year or for varying periods during the year; (f) *Clear Cut with Burned Area* for pixels where fires cleared the land after tree removal; (g) *Clear Cut with Bare Soil* where the forest has been completely removed; (h) *Clear Cut with Vegetation* where some vegetation remains after most trees have been removed.

Using these samples, we built a classification model using the temporal convolution neural network (TempCNN) algorithm [26]. TempCNN is a neural network architecture designed to process image time series. It has three 1D convolutional layers and a final SoftMax layer for classification. TempCNN applies one-dimensional convolutions on the input sequence to capture temporal dependencies, allowing the network to learn long-term dependencies in the input sequence. Each layer of the model captures temporal dependencies at a different scale. Due to its multi-scale approach, TempCNN can capture complex temporal patterns in the data and produce accurate predictions. The algorithm has been evaluated favorably for land classification [17,18,27] and is a benchmark for comparison with proposed advances in deep-learning classification [11].

After tuning the hyperparameters of TempCNN, we found that an adequate configuration would be to use three 1D CNN layers and a SoftMax as proposed by Pelletier et al. [26] with 1D kernel sizes of 64 neurons, a temporal filter size of 5 timesteps per layer, and a learning rate of 2.46×10^{-3}.

The choice of the TempCNN algorithm is intended as one of many possible examples of image time-series algorithms. Alternatives would be machine-learning approaches such as random forest, support vector machines, extreme gradient boosting, or deep-learning methods such as temporal attention encoders [28]. These and similar methods can produce a set of probability matrices as output, as required by our method (cf Equation (1)).

4. Results and Discussion

4.1. Classifier Output Without Post-Processing

The output of the time-series machine-learning classifier is a set of probability maps, one for each class. Figure 2 shows a detail of the probability maps for classes *Forest* and *Clear Cut with Bare Soil*, *Water* and *Seasonally Flooded*. The probability map for *Forest* shows high values associated with compact patches. *Clear Cut with Bare Soil* areas are mostly dense patches of high probability whose geometrical boundaries result from forest cuts. Regions of *Seasonally Flooded* occur in areas bounded by the Madeira River in the top left corner of the image and areas close to smaller rivers. By contrast, areas of class *Water* include homogeneous areas of high probability. Since classes have different behaviors, the post-processing procedure should enable users to control how to handle outliers and border pixels of each class.

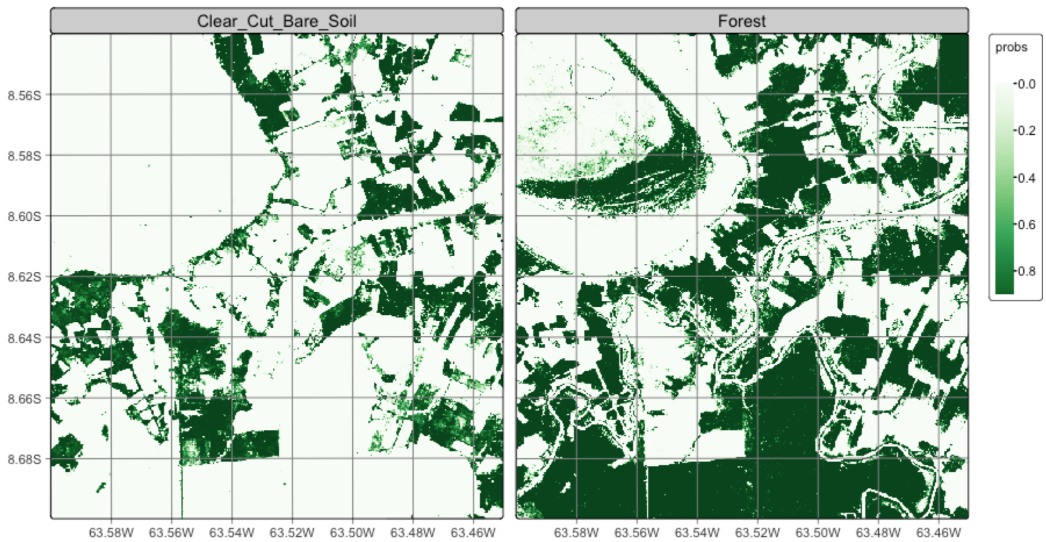

Figure 2. *Cont.*

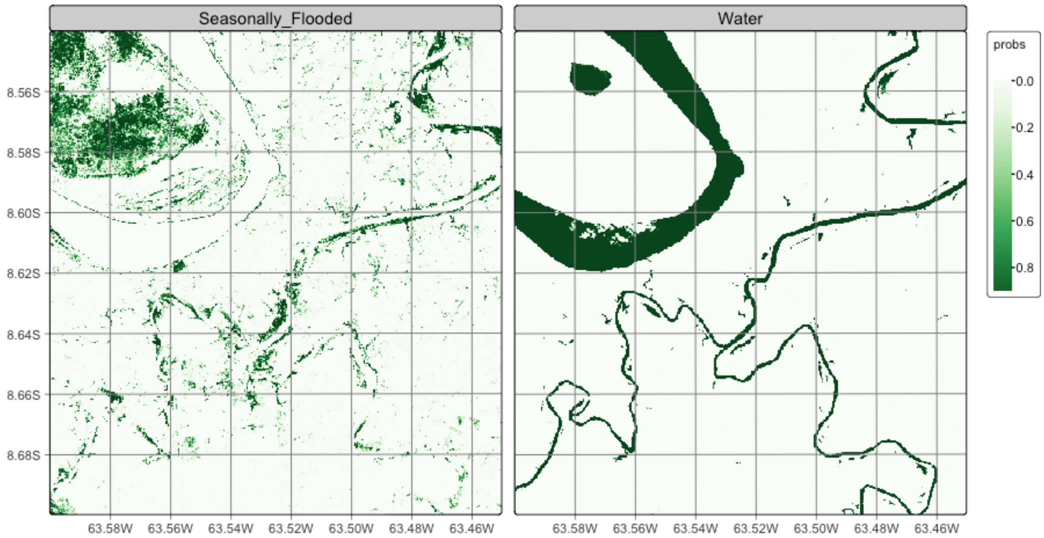

Figure 2. Probability maps for classes Forest, Clear Cut with Bare Soil, Seasonally Flooded, and Water (detail of full map).

Figure 3 shows the unsmoothed classification map and the one obtained by Bayesian smoothing. The unsmoothed map is obtained by taking the class of higher probability to each pixel without considering the spatial context. Many places have the so-called "salt-and-pepper" effect resulting from misclassified pixels. A closer examination of the non-smoothed map reveals different outliers and noisy classification cases. One example is the presence of small isolated or linear patches of transitional classes *Clear Cut with Vegetation* and *Clear Cut with Burned Area* inside larger areas of more stable classes such as *Clear Cut with Bare Soil* and *Forest*. Consider a situation where the removal of native forest area is captured as part of the time series. The machine-learning classifier may consider that the pixel is transitioning from forest to bare soil and thus label it as an intermediate class. Depending on the extent of the change, the result may be an isolated pixel or a compact area. Post-processing intends to remove outliers while keeping the continuous areas.

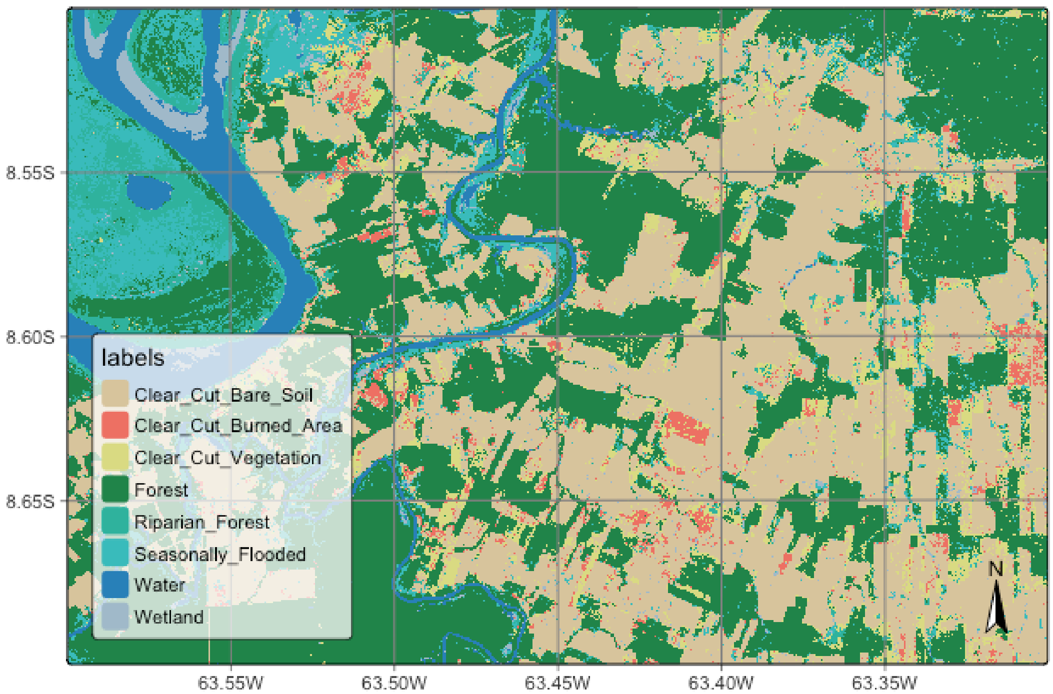

Figure 3. *Cont.*

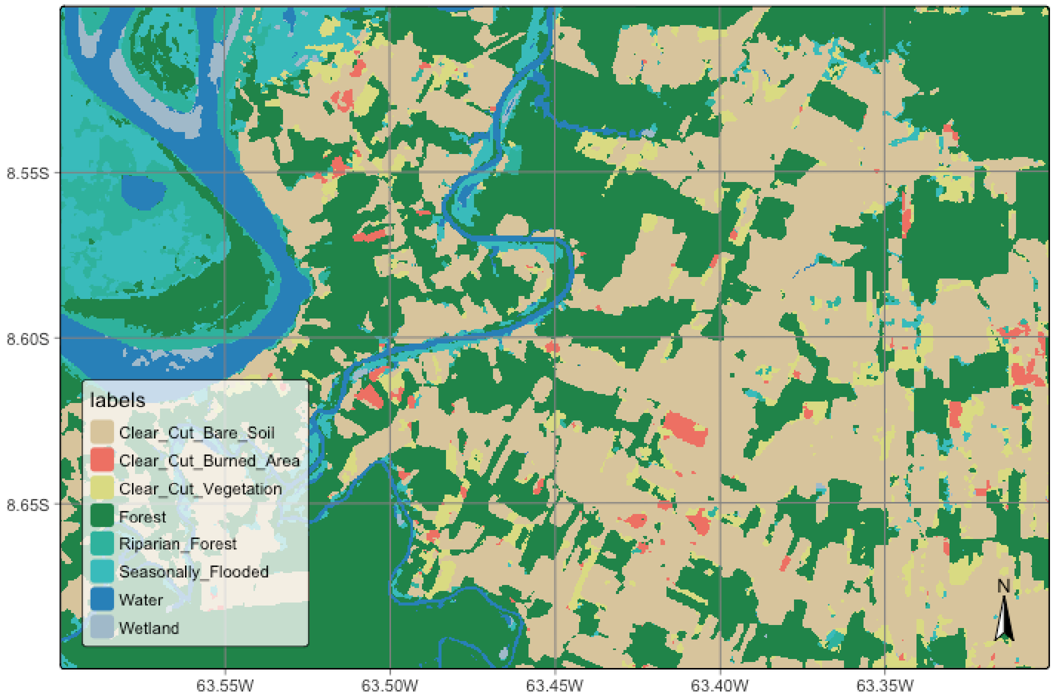

Figure 3. Detail of classified map without smoothing (**top**) and with Bayesian smoothing (**bottom**) (detail of full map).

A similar situation occurs in the case of classes *Forest*, *Seasonally Flooded*, *Riparian Forest* and *Wetlands*. The boundaries between these classes are fuzzy and location-dependent. Consider the case of small streams running through the forest. In some cases, these water bodies are not visible in the image, but linear patches of riparian forests will be detected. In cases where these patches are not big enough to form a continuous shape, they will appear as outliers inside larger forest areas. Thus, the boundaries between land-cover classes are usually blurry, making it necessary to run post-processing methods to improve the results.

4.2. Using Local Logit Variances to Estimate the Hyperparameter

The local logit variances correspond to the $\hat{s}^2_{i,k}$ parameter in the Bayesian inference shown in Equation (7). To obtain the local logit variances, we took a 9×9 sliding window and used the 50% upper probability values, according to Equation (6); in this way, 41 logit values are used to estimate $\hat{m}_{i,k}$ and $\hat{s}^2_{i,k}$ for each pixel i and class k. We computed the quantiles using the **R** `stats` package. The summary statistics are shown in Table 1 for values greater than the 75% quantile. There is a significant increase from the 95% quantile to the maximum. This interval is where the local logit variances have the highest variability and correspond to border and outlier pixels.

Table 1. Summary statistics of local logit variances for classes *Clear Cut with Bare Soil* (Bare Soil), *Clear Cut With Burned Area* (Burn), *Clear Cut with Vegetation* (Veg), *Forest* (For), *Riparian Forest* (RipFor), *Seasonally Flooded* (Flood), *Water* and *Wetlands* (Wet).

	Bare Soil	Burn	Veg	For	RipFor	Flood	Water	Wet
75%	1.23	1.31	2.90	1.19	2.67	3.52	0.08	1.28
80%	1.71	1.71	3.41	2.10	3.20	4.15	0.19	1.76
85%	2.56	2.32	4.16	3.79	3.86	5.01	0.40	2.44
90%	4.46	3.37	5.30	7.12	4.91	6.44	0.85	3.63
95%	10.24	5.23	7.55	15.77	7.16	9.13	2.62	6.22
100%	51.46	30.76	39.59	56.12	56.81	55.65	64.73	51.31

We use the variance data to support an adequate assignment of σ_k^2 value in Equation (7). The Bayesian model for post-processing expressed in Equation (5) assumes that a Gaussian distribution can describe the probability for each pixel for each class. In general, the mean and variance of this distribution will be different for each pixel since it represents the natural variability of each land-cover class. The empirical Bayes approximation (cf Equation (7)) takes a single value for the class variance σ_k^2. This requires some additional assumptions about how land-cover classes behave. At this point, experts use their knowledge of landscape patterns and the expected classification output to choose suitable hyperparameter values σ_k^2.

When selecting values of σ_k^2 for each class, one should consider that post-processing aims to remove outliers. In general, the class of most pixels should not be changed since they represent areas of low uncertainty. In these areas, we expect the local logit variance to be low. The empirical Bayes update (cf. Equation (7)) will not change the value of the logits of pixels significantly. The main concern for post-processing is places where the local logit variance is high, with values in the 95% to 100% range. In these cases, setting σ_k^2 to low values (say in the 75% range) will not change the pixel values. Outliers will not be removed. Thus, measuring the local variances is relevant for setting the values of the hyperparameter.

We chose values of σ_k^2 that reflect our prior expectation of the spatial patterns of each class. Classes *Clear Cut with Vegetation*, *Clear Cut with Burned Area*, and *Clear Cut with Bare Soil* have high spatial variability since they represent different actions leading to deforestation. In this case, we want to produce denser spatial clusters and remove "salt-and-pepper" outliers. For this reason, we take σ_k^2 values close to the 100% range of the local variances. Classes *Riparian Forest*, *Seasonally Flooded*, and *Wetlands* have time-varying spectral responses, and it is not simple to distinguish between them. For this reason, their border pixels have high variances, and thus, we choose high σ_k^2 values. For more stable classes *Forest* and *Water*, preserving their original spatial shapes is beneficial. For this reason, we set smaller σ_k^2 values in the 95% range for these classes. The resulting hyperparameter values, shown in Table 2, were used in the EB estimate.

Table 2. Chosen hyperparameter values for Bayesian smoothing.

	σ_k^2	Quantile
Clear Cut Bare Soil	50.0	100%
Clear Cut Burned Area	35.0	100%
Clear Cut Vegetation	40.0	100%
Forest	14.0	95%
Riparian Forest	56.0	100%
Seasonally Flooded	40.0	100%
Water	4.0	95%
Wetland	54.0	100%

4.3. Bayesian Smoothed Map and Comparison with Alternatives

The hyperparameter values of Table 2 were used to run the empirical Bayes estimate. The probabilities $x_{i,k}$ generated by the machine-learning classifier are updated according to Equation (7). To obtain $\hat{m}_{i,k}$ and $\hat{s}_{i,k}^2$ we used the top 50% values of a 9 × 9 sliding window. Figure 3 shows the smoothed classification map obtained by taking the class of higher probability to each pixel after post-processing. Areas of *Clear Cut with Bare Soil* and *Forest* increased slightly. The class *Clear Cut with Burned Area* now appears as dense patches as befits areas affected by forest fire, while class *Clear Cut with Vegetation* has preserved most of its areas while removing outliers. Compared to the non-smoothed map, the result is more compact and more consistent.

We compared Bayesian smoothing with the established methods of Gaussian and bilateral smoothing. Figure 4 visually compares a small area of the Sentinel-2 tile that highlights the differences between the methods. Arguably, Bayesian smoothing does a better job of removing outliers and keeping the original boundaries, while the Gaussian and bilateral methods keep many outliers. The Gaussian and bilateral methods also tend to produce more disc-shaped areas because of their isotropic kernel. While rounded shapes may be more pleasing to the viewer, such forms may not reflect the actual data. Thus, Bayesian smoothing appears to be a suitable compromise between spatial consistency and outlier removal.

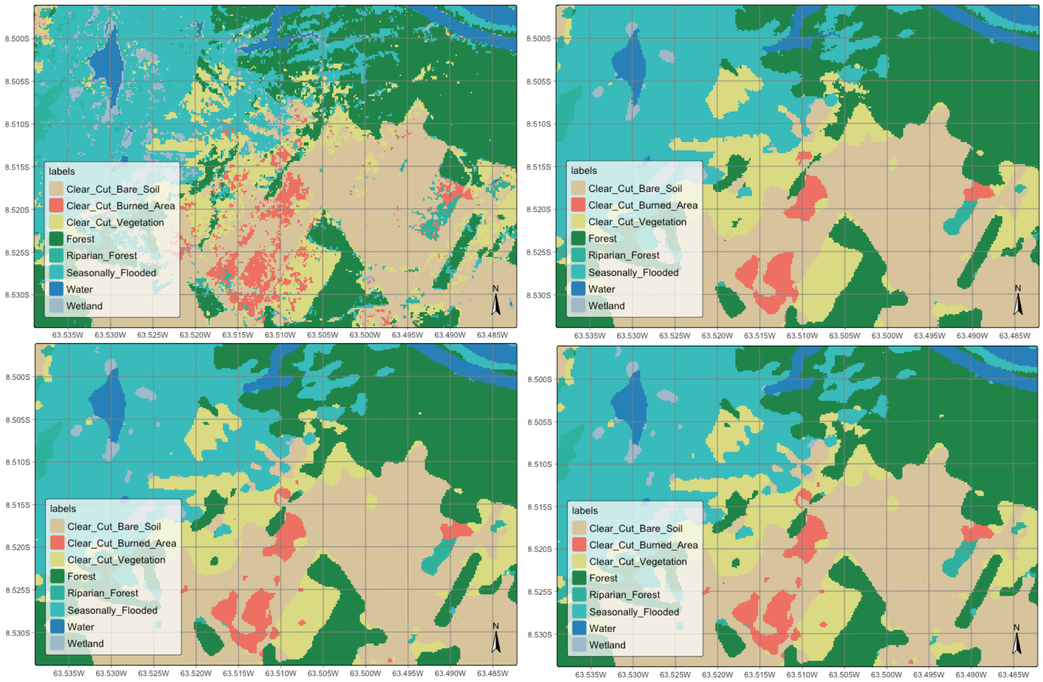

Figure 4. Detail of small area of classified map without smoothing (**top left**) and with Bayesian (**top right**), Gaussian (**bottom right**), and bilateral smoothing.

To run a quantitative evaluation, we used the probability map generated by the TempCNN algorithm to determine points of high epistemic uncertainty. For each point, we calculated its entropy. Entropy is a measure of uncertainty used by Claude Shannon in his classic work "A Mathematical Theory of Communication" [29]. It is related to the amount of variability in the probabilities associated with a pixel. The lower the variability, the lower the entropy. Let $x_{i,k}$ be the probability of class k for pixel i. The entropy is calculated as

$$\theta_E = \frac{-\sum_{k=1}^{K} x_{i,k} * log_2(x_{i,k})}{log_2 n} \tag{11}$$

After calculating the entropy map, we selected 10,000 points with the highest entropy values as representative of possible places of disagreement between the smoothing methods. These points comprise 0.03% of the total area and are thus likely to be linked to outliers and borders. Out of them, there were 5024 points showing no agreement between the original non-smoothed classified map, the one generated by Bayesian smoothing, and those generated by Gaussian or bilateral smoothing. We selected 300 points for manual labeling from the set of conflicting results. We compared the labels with the original Sentinel-2 images for each of them, considering different dates during the year. The overall accuracy for the manually labeled points is shown in Table 3.

The above results indicate that, for points with higher uncertainty, which are presumably outliers, Bayesian smoothing is better at predicting the correct class. The result is not the final map's overall accuracy since only a small fraction of all points were evaluated. The comparison shows that Bayesian smoothing is the post-processing method that best improves the accuracy of the final classification relative to the original probability map, compared to Gaussian and bilateral smoothing. These positive outcomes are expected since Bayesian smoothing benefits from expert knowledge.

Table 3. Overall accuracy for points with high entropy values for their estimated probabilities.

Method	Accuracy
Original map	18.8%
Bayesian smoothing	68.3%
Gaussian smoothing	16.2%
Bilateral smoothing	19.1%

5. Conclusions

The increasing use of satellite image time series for land classification with big Earth-observation data introduces new challenges. A significant issue is refining the outputs of machine-learning and deep-learning classifiers, which often contain noise and outliers. The proposed Bayesian smoothing method is particularly effective for classifying Earth-observation data using a time-first, space-later approach. This method prioritizes temporal analysis before spatial analysis, first classifying temporal data for each pixel and then integrating spatial information to remove outliers [30].

Focusing on temporal analysis first provides a clearer perspective on landscape changes. It allows the identification of seasonal and long-term patterns, and anomalies such as wildfires, floods, or droughts. In this framework, each pixel in a data cube is treated as a time series, leveraging information from its temporal instances. The time-series classification generates a set of class probability maps, which then serve as input for the spatial analysis. Bayesian smoothing refines the results of this time-first classification by accounting for spatial relationships between pixels. The resulting maps seamlessly combine temporal and spatial information, enhancing the accuracy of land classification. Recent research on deep learning for image time-series classification underscores the importance of spatial smoothing as a post-classification step [27], highlighting the relevance of Bayesian smoothing in time-first, space-later EO data analysis.

Post-processing is an essential step in any classification workflow. Bayesian smoothing improves the delineation of object boundaries created by classification and removes outliers resulting from pixel-based processing. It is robust and reliable, guided by objective metrics of class probability variance. Validated for large-scale data analysis, this method fits well in applications requiring a time-first, space-based approach. Moreover, it represents a compelling alternative to other post-processing techniques in the literature, offering a balance of reliability and precision for refining land classification results.

Author Contributions: Conceptualization: G.C., R.A. and A.C.; Data curation: F.S., F.C., A.S., A.R. and A.P.D.; Formal analysis: G.C., R.A. and A.C.; Funding acquisition: G.C.; Investigation, A.S., A.R. and A.P.D.; Methodology: G.C., R.A., A.C. and R.S.; Project administration: G.C.; Software: R.S., F.S. and F.C.; Resources: G.C.; Validation: A.S., A.R. and A.P.D.; Visualization: R.S., F.S. and F.C.; Writing—original draft: G.C. and R.A.; Writing—review and editing: G.C. and R.A. All authors have read and agreed to the published version of the manuscript.

Funding: The authors acknowledge the support of the following institutions: (a) Amazon Fund, established by Brazil with financial contribution from Norway, through contract 17.2.0536.1.; (b) International Climate Initiative of the German government (IKI) under grant 17-III-084-Global-A-RESTORE+ ("RESTORE+: Addressing Landscape Restoration on Degraded Land in Indonesia and Brazil"); (c) Microsoft AI for Good initiative under Azure Sponsorship grant d8072905-1d68-4b9e-add1-11da1a2e5908; (d) Instituto Clima e Sociedade under grants G-22-01193 and G-23-01643 ("Modernization of PRODES and DETER Amazon monitoring systems"); (e) Open-Earth-Monitor Cyberinfrastructure project, under European Commission grant agreement No. 101059548; (f) FAO-EOSTAT initiative to use big Earth-observation data for agricultural statistics under contract RLC/TCP/RLA/3902 number 3079474; (g) Brazilian Research Council (CNPq) through grant 303589/2023-0.

Data Availability Statement: All of the data and scripts used to reproduce the results of this paper are openly available in the repository https://github.com/e-sensing/rondonia20LMR, available since 15 October 2024. Please see instructions on how to reuse this work on the above landing page. The Bayesian smoothing algorithm is part of the **R** package `sits`, an end-to-end toolkit for land use and land-cover classification using big Earth-observation data developed by the authors, which is available on the standard **R** repository CRAN. Full documentation on using `sits` is available at https://e-sensing.github.io/sitsbook/, available since 19 June 2023.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Belgiu, M.; Dragut, L. Random Forest in Remote Sensing: A Review of Applications and Future Directions. *ISPRS J. Photogramm. Remote Sens.* **2016**, *114*, 24–31. [CrossRef]
2. Ma, L.; Liu, Y.; Zhang, X.; Ye, Y.; Yin, G.; Johnson, B.A. Deep Learning in Remote Sensing Applications: A Meta-Analysis and Review. *ISPRS J. Photogramm. Remote Sens.* **2019**, *152*, 166–177. [CrossRef]
3. Zhu, X.X.; Tuia, D.; Mou, L.; Xia, G.S.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 8–36. [CrossRef]
4. Schindler, K. An Overview and Comparison of Smooth Labeling Methods for Land-Cover Classification. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 4534–4545. [CrossRef]
5. Wang, L.; Shi, C.; Diao, C.; Ji, W.; Yin, D. A Survey of Methods Incorporating Spatial Information in Image Classification and Spectral Unmixing. *Int. J. Remote Sens.* **2016**, *37*, 3870–3910. [CrossRef]
6. Huang, X.; Lu, Q.; Zhang, L.; Plaza, A. New Postprocessing Methods for Remote Sensing Image Classification: A Systematic Study. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 7140–7159. [CrossRef]
7. Ghimire, B.; Rogan, J.; Miller, J. Contextual Land-Cover Classification: Incorporating Spatial Dependence in Land-Cover Classification Models Using Random Forests and the Getis Statistic. *Remote Sens. Lett.* **2010**, *1*, 45–54. [CrossRef]
8. Woodcock, C.E.; Loveland, T.R.; Herold, M.; Bauer, M.E. Transitioning from Change Detection to Monitoring with Remote Sensing: A Paradigm Shift. *Remote Sens. Environ.* **2020**, *238*, 111558. [CrossRef]
9. Verbesselt, J.; Hyndman, R.; Newnham, G.; Culvenor, D. Detecting Trend and Seasonal Changes in Satellite Image Time Series. *Remote Sens. Environ.* **2010**, *114*, 106–115. [CrossRef]
10. Pasquarella, V.J.; Holden, C.E.; Kaufman, L.; Woodcock, C.E. From Imagery to Ecology: Leveraging Time Series of All Available LANDSAT Observations to Map and Monitor Ecosystem State and Dynamics. *Remote Sens. Ecol. Conserv.* **2016**, *2*, 152–170. [CrossRef]
11. Miller, L.; Pelletier, C.; Webb, G.I. Deep Learning for Satellite Image Time Series Analysis: A Review. *arXiv* **2024**, arXiv:2404.03936. [CrossRef]
12. Tomasi, C.; Manduchi, R. Bilateral Filtering for Gray and Color Images. In Proceedings of the Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271), Bombay, India, 7 January 1998; pp. 839–846. [CrossRef]
13. Picoli, M.; Camara, G.; Sanches, I.; Simoes, R.; Carvalho, A.; Maciel, A.; Coutinho, A.; Esquerdo, J.; Antunes, J.; Begotti, R.A.; et al. Big Earth Observation Time Series Analysis for Monitoring Brazilian Agriculture. *ISPRS J. Photogramm. Remote Sens.* **2018**, *145*, 328–339. [CrossRef]
14. Simoes, R.; Picoli, M.C.A.; Camara, G.; Maciel, A.; Santos, L.; Andrade, P.R.; Sánchez, A.; Ferreira, K.; Carvalho, A. Land Use and Cover Maps for Mato Grosso State in Brazil from 2001 to 2017. *Sci. Data* **2020**, *7*, 34. [CrossRef] [PubMed]
15. Hadi, F.; Sabri, L.M.; Prasetyo, Y.; Sudarsono, B. Leveraging Time-Series Imageries and Open Source Tools for Enhanced Land Cover Classification. In *Proceedings of the IOP Conference Series: Earth and Environmental Science*; IOP Publishing: Bristol, UK, 2023; Volume 1276, p. 012035. [CrossRef]
16. Giuliani, G. Time-First Approach for Land Cover Mapping Using Big Earth Observation Data Time-Series in a Data Cube—A Case Study from the Lake Geneva Region (Switzerland). *Big Earth Data* **2024**, *8*, 435–466. [CrossRef]
17. Werner, J.P.S.; Belgiu, M.; Bueno, I.T.; Dos Reis, A.A.; Toro, A.P.S.G.D.; Antunes, J.F.G.; Stein, A.; Lamparelli, R.A.C.; Magalhães, P.S.G.; Coutinho, A.C.; et al. Mapping Integrated Crop–Livestock Systems Using Fused Sentinel-2 and PlanetScope Time Series and Deep Learning. *Remote Sens.* **2024**, *16*, 1421. [CrossRef]
18. Simoes, R.; Camara, G.; Queiroz, G.; Souza, F.; Andrade, P.R.; Santos, L.; Carvalho, A.; Ferreira, K. Satellite Image Time Series Analysis for Big Earth Observation Data. *Remote Sens.* **2021**, *13*, 2428. [CrossRef]
19. Appel, M.; Pebesma, E. On-Demand Processing of Data Cubes from Satellite Image Collections with the Gdalcubes Library. *Data* **2019**, *4*, 92. [CrossRef]
20. Gelman, A.; Carlin, J.B.; Stern, H.S.; Dunson, D.B.; Vehtari, A.; Rubin, D.B. *Bayesian Data Analysis*, 3rd ed.; CRC Press: Boca Raton, FL, USA, 2014.
21. Carlin, B.; Louis, T. *Bayes and Empirical Bayes Methods for Data Analysis*; CRC Press: Boca Raton, FL, USA, 2000.
22. Van der Vaart, A. *Asymptotic Statistics*; Cambridge University Press: Cambridge, UK, 2000.
23. Dwyer, J.L.; Roy, D.P.; Sauer, B.; Jenkerson, C.B.; Zhang, H.K.; Lymburner, L. Analysis Ready Data: Enabling Analysis of the Landsat Archive. *Remote Sens.* **2018**, *10*, 1363. [CrossRef]

24. Ferreira, K.R.; Queiroz, G.R.; Vinhas, L.; Marujo, R.F.B.; Simoes, R.E.O.; Picoli, M.C.A.; Camara, G.; Cartaxo, R.; Gomes, V.C.F.; Santos, L.A.; et al. Earth Observation Data Cubes for Brazil: Requirements, Methodology and Products. *Remote Sens.* **2020**, *12*, 4033. [CrossRef]
25. Giuliani, G.; Chatenoux, B.; De Bono, A.; Rodila, D.; Richard, J.P.; Allenbach, K.; Dao, H.; Peduzzi, P. Building an Earth Observations Data Cube: Lessons Learned from the Swiss Data Cube (SDC) on Generating Analysis Ready Data (ARD). *Big Earth Data* **2017**, *1*, 100–117. [CrossRef]
26. Pelletier, C.; Webb, G.I.; Petitjean, F. Temporal Convolutional Neural Network for the Classification of Satellite Image Time Series. *Remote Sens.* **2019**, *11*, 523. [CrossRef]
27. Perbet, P.; Guindon, L.; Côté, J.F.; Béland, M. Evaluating Deep Learning Methods Applied to Landsat Time Series Subsequences to Detect and Classify Boreal Forest Disturbances Events: The Challenge of Partial and Progressive Disturbances. *Remote Sens. Environ.* **2024**, *306*, 114107. [CrossRef]
28. Garnot, V.S.F.; Landrieu, L. Lightweight Temporal Self-attention for Classifying Satellite Images Time Series. In *Advanced Analytics and Learning on Temporal Data, Proceedings of the 5th ECML PKDD Workshop, AALTD 2020, Ghent, Belgium, 18 September 2020*; Lemaire, V., Malinowski, S., Bagnall, A., Guyet, T., Tavenard, R., Ifrim, G., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2020; pp. 171–181.
29. Shannon, C.E. A Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [CrossRef]
30. Camara, G.; Picoli, M.; Simoes, R.; Maciel, A.; Carvalho, A.; Coutinho, A.; Esquerdo, J.; Antunes, J.; Begotti, R.; Arvor, D. Land Cover Change Maps for Mato Grosso State in Brazil: 2001–2016. *PANGAEA* **2019**. [CrossRef]