remote sensing

MDPI

*Article*

# Next Generation Mapping: Combining Deep Learning, Cloud Computing, and Big Remote Sensing Data

**Leandro Parente [1,*], Evandro Taquary [1], Ana Paula Silva [1], Carlos Souza, Jr. [2] and Laerte Ferreira [1]**

[1] Image Processing and GIS Laboratory (LAPIG), Federal University of Goiás (UFG), Goiânia-GO 74001-970, Brazil; evandro.taquary@gmail.com (E.T.); annapaulamatos1@gmail.com (A.P.S.); laerte@ufg.br (L.F.)

[2] IMAZON – Amazon Institute of People and the Environment, Belém-PA 66055-200, Brazil; souzajr@imazon.org.br

[*] Correspondence: leal.parente@gmail.com; Tel.: +55-62-3521-1360

Received: 9 October 2019; Accepted: 26 November 2019; Published: 3 December 2019

**Abstract:** The rapid growth of satellites orbiting the planet is generating massive amounts of data for Earth science applications. Concurrently, state-of-the-art deep-learning-based algorithms and cloud computing infrastructure have become available with a great potential to revolutionize the image processing of satellite remote sensing. Within this context, this study evaluated, based on thousands of PlanetScope images obtained over a 12-month period, the performance of three machine learning approaches (random forest, long short-term memory-LSTM, and U-Net). We applied these approaches to mapped pasturelands in a Central Brazil region. The deep learning algorithms were implemented using TensorFlow, while the random forest utilized the Google Earth Engine platform. The accuracy assessment presented F1 scores for U-Net, LSTM, and random forest of, respectively, 96.94%, 98.83%, and 95.53% in the validation data, and 94.06%, 87.97%, and 82.57% in the test data, indicating a better classification efficiency using the deep learning approaches. Although the use of deep learning algorithms depends on a high investment in calibration samples and the generalization of these methods requires further investigations, our results suggest that the neural network architectures developed in this study can be used to map large geographic regions that consider a wide variety of satellite data (e.g., PlanetScope, Sentinel-2, Landsat-8).

**Keywords:** machine learning; deep learning; random forest; U-Net; LSTM; LULC classification; PlanetScope

## 1. Introduction

The science and technology of remote sensing has reached the era of big data [1]. This imposes a paradigm change in the way remote sensing data is processed to extract information for a variety of environmental and societal applications. The proliferation of constellations of compact satellites adds another level of challenge to processing remote sensing imagery [2]. The extraction of information from this type of data, regardless of the application category (e.g., global climate change, urban planning, land-use and land-cover (LULC) monitoring), can be classified as big remote sensing data, while simultaneously meeting volume, variety, and data growth rates [3]. The challenges involved in this process have led to the emergence of cloud-based platforms, specifically for remote sensing, that can perform planetary-scale analysis of massive amounts of data [4–7].

The combination of these factors, i.e., the high availability of satellite data and adequate analysis platforms, has enabled the emergence of terrestrial coverage mappings on global [8–10], continental [11–13], and national [14–16] scales, which are useful for supporting decision-making and broader policy objectives [17]. In general, these mappings were produced with supervised classifications using conventional machine learning algorithms (e.g., random forest, classification and regression tree-CART) implemented on platforms with thousands of processors capable of handling communication, parallelism, and distributed computing problems [18].

Although the operation of these algorithms for large geographic regions uses cloud-based platforms, the classification approaches that are implemented generally use a step of feature engineering to handle and reduce the dimensionality of the input satellite data. In addition to allowing faster and computationally more efficient execution, this step provides a better understanding of the features that are most relevant to the training process and that most often impacts the accuracy of the classification result [19,20]. Typically, feature engineering applied to remote sensing depends on human experience and domain knowledge, and is time-consuming as it needs to consider a lot of complex factors simultaneously (e.g., inter-class similarity, intra-class variability, atmospheric conditions [21]). Furthermore, as we enter a new era of big remote sensing data, this step becomes more arduous and challenging.

In this scenario, deep learning algorithms (e.g., convolutional neural networks, recurrent neural networks, semantic segmentation [22]) may be more appropriate as they are able to construct and select the most relevant features of raw data automatically, without human interference, which can result in better and more accurate mapping products [23]. In fact, the deep learning approaches have proven their effectiveness in many application domains, dramatically improving speech recognition, visual object recognition, health diagnosis, and others applications [24]. However, most studies that apply deep learning approaches to remote sensing data are restricted to the use of publicly available benchmark image datasets that involve pre-processed images organized under ideal conditions and have been previously labeled [25–27], such that research focuses on the comparative analysis of accuracy gains. Because of that, practical applications of deep learning algorithms for land-use and land-cover mapping, operationally and for large geographic regions, require further research investigations [22,28].

On the other hand, deep learning algorithms generally require substantially more training data than conventional machine learning supervised classifications [24]. Acquiring training data is challenging due to the high costs and difficulties involved in obtaining field data, especially for past years [29], as well as due to the subjectivity and time-consuming work involved in the visual interpretation of remote sensing images [30]. In this sense, the feature engineering exemption from deep learning is likely to be offset by a greater need for samples, a trade-off that should be considered when designing national, continental, and global mapping products.

Another trade-off that needs to be considered when using deep learning algorithms concerns the increased need for computational processing, which is usually met by using graphical processing units (GPUs) [24]. Although there are initiatives that seek to advance GPU satellite data processing [31,32], existing solutions are not as operational as cloud-based platforms that make exclusive use of CPUs, such as the Google Earth Engine [5]. From this perspective, we can state that the next generation of land surface mappings involves an adequate understanding of the benefits and trade-offs between conventional (e.g., random forest, CART, support vector machine-SVM) and cutting-edge (e.g., Google LeNet, U-Net, generative adversarial network-GAN) machine learning algorithms when applied to remote sensing, as well as a proper definition and evaluation of operational strategies that is capable of promoting better information extraction.

Aiming to contribute to the next generation of land-cover and land-use maps, we evaluated the performance of random forest, long short-term memory (LSTM), and U-Net algorithms, which were applied to a large dataset of PlanetScope satellite imagery, and implemented in the Google Earth Engine and in a local infrastructure using Tensorflow. Additionally, we present a framework regarding pre-processing, the preparation of training data, feature engineering, and accuracy assessment, and discuss the challenges faced in implementing such novel methods to mapping very large areas.

## 2. Materials and Methods

We focused the implementation of machine learning and deep learning algorithms on the mapping of pasture areas, one of the most complex LULC classes to map [33], using PlanetScope imagery obtained with the Planet Dove constellation. With over 130 commercial CubeSats in orbit, the Planet constellation enables the monitoring of the entire terrestrial surface with a daily frequency and a spatial resolution of 3–5m, producing a very large data volume, which fits into the realm of big remote sensing data, despite not presenting the same radiometric quality and calibration precision of government satellites (e.g., Sentinel 2 and Landsat 8) [2].

Specifically, we used thousands of PlanetScope images acquired over a one-year period to produce 12 monthly mosaics over a study area located in central Brazil [34]. Based on these mosaics, a specific training dataset for mapping pasture areas in the region was produced by considering aspects such as the local variability of LULC classes and potential spectral confusion. The monthly mosaics and training set were used to build supervised classification models using the random forest [35], LSTM [36], and U-Net [37] algorithms. In relation to the random forest, a conventional machine learning approach, a feature engineering step was performed, aiming to improve its performance in the detection of pasture areas. For LSTM and U-Net, both of which are deep learning approaches, the model building considered only raw data (i.e., the monthly mosaics). At the end of the process, the three resulting mappings were evaluated by considering the validation data, previously separated from the training data, and data obtained via fieldwork in the study area (considered as test data). The datasets and methodological steps used in this study are presented in Figure 1 and are detailed in the following sections.

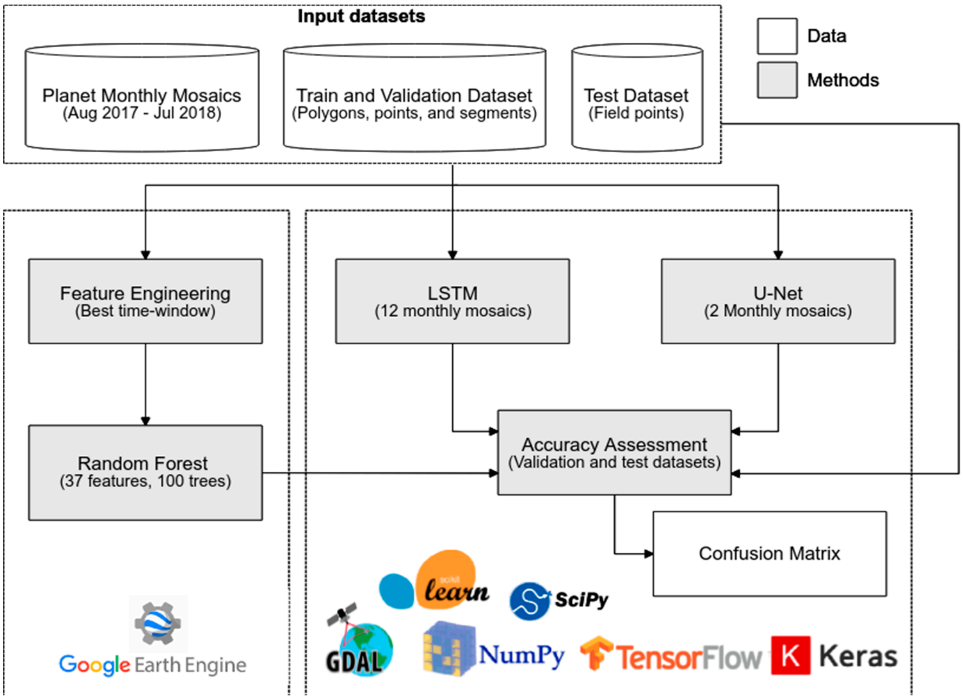
Input datasets
Planet Monthly Mosaics (Aug 2017 - Jul 2018)
Train and Validation Dataset (Polygons, points, and segments)
Test Dataset (Field points)
Data
Methods
Feature Engineering (Best time-window)
LSTM (12 monthly mosaics)
U-Net (2 Monthly mosaics)
Random Forest (37 features, 100 trees)
Accuracy Assessment (Validation and test datasets)
Confusion Matrix
Google Earth Engine
learn
SciPy
GDAL
NumPy
TensorFlow
Keras

**Figure 1.** Data and methods used in the generation of pasture mappings for the study area. The random forest classification approach was performed using the Google Earth Engine, while the deep learning approaches were implemented on a local server. LSTM: long short-term memory.

### 2.1. Pre-processing of PlanetScope Images

Our study area was a region of ≈18,000 km$^2$ in area, delimited by the topographic chart SD-22-ZC [38], and located in the central portion of the Cerrado biome in the Brazilian state of Goiás (Figure 2). This region was evaluated using 36,962 PlanetScope images (i.e., four-band PlanetScope Scenes), acquired between August 2017 and July 2018. We used this large imagery dataset to produce 12 monthly mosaics, which were screened for clouds and/or cloud-shadows contamination. Cloud and cloud-shadow pixels were identified using histogram slicing, which used empirically defined spectral thresholds, along with a frequency filter that prioritized pixels with low temporal stability, and consequently, had a greater chance of being associated with clouds and/or cloud shadows. The mosaicking process considered a top-of-atmosphere correction, based on Landsat reference values, and the normalization of the PlanetScope images with less than 20% cloud coverage in order to minimize the amount of noise in the final mosaic. The normalization process was based on a histogram adjustment, using an annual median mosaic (produced with only cloudless images, i.e., cloud-cover <1%, obtained over 12 months) as a reference. The histogram adjustments were implemented using statistical regressions (via a random forest) between the cumulative distributions of the PlanetScope images and the reference mosaic, band by band. Finally, the daily PlanetScope images were stacked and reduced using the median to generate the monthly mosaics. All of this was implemented on the Google Earth Engine platform, and the final mosaics, resampled to 4 m, can be accessed at https://code.earthengine.google.com/bf9fb20125dede14ab0457490a5a8f42 [34]

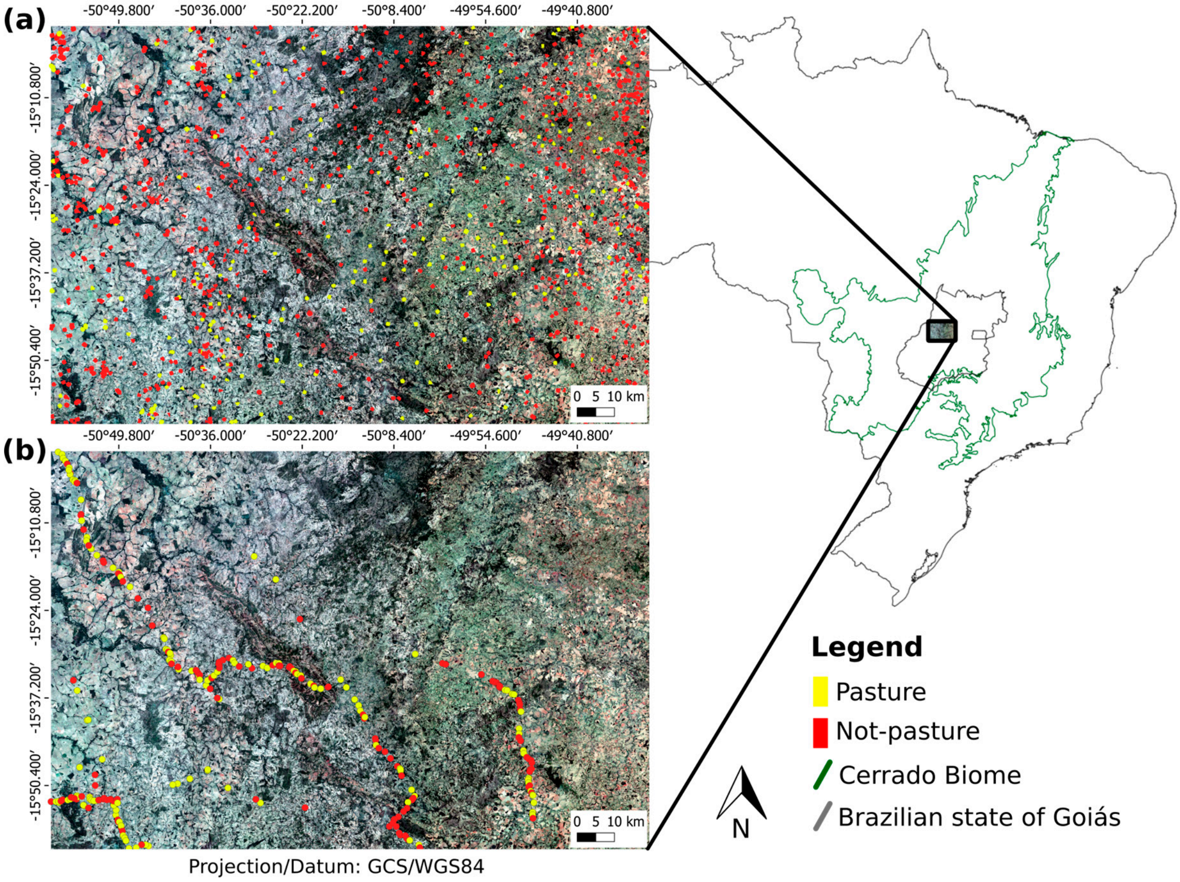

**Figure 2.** Study area, which is 18,000 km$^2$ in area and located in the state of Goiás, Brazil. (**a**) Spatial distribution of polygons collected via visual interpretation and (**b**) field samples in relation to the study area (the PlanetScope mosaic shown refers to November 2017).

### *2.2. Training, Validation, and Test Data*

Based on the PlanetScope of the study area, we collected pasture and non-pasture samples in order to build the training, validation, and test datasets. Considering the monthly mosaics, as well as having prior knowledge of the region, image interpretation experts collected 1326 regular and homogeneous polygons (i.e., with pixels of only one LULC class), well-distributed throughout the study area and covering, as much as possible, all the LULC class variability (Figure 2a). For the pasture class, we collected polygons of "regular pasture," "wet pasture," "pasture with bare soil," and "pasture with shrubs"; for the not-pasture class, we collected polygons of "crop," "deforestation," "forest formation," "other land-use," "planted forest," "sand bank," "savannah formation," "urban area," and "water." These polygons were used to generate point and segment samples, which were used for the classifier training and validation. For the test set, fieldwork data obtained from the study area were used.

### 2.2.1. Sampling Points

Aiming to minimize the spatial autocorrelation of the point samples, a minimum distance between samples located within the same polygon was established, which varied according to the LULC classes defined by the interpreters. These distances were calculated according to Equation (1) (defined using the relation between the number of pixels and the area of a perfect square polygon), which considered the same number of samples for pasture subclasses and non-pasture subclasses (Table 1):

$$d = \left(\sqrt{\frac{N}{n}} - 1\right) \times s \tag{1}$$

where for each LULC class: $d$ is the minimum class distance, $N$ is the total of pixels in the class, $n$ is the number of desired samples in a given class, and $s$ is the pixel size.

At the end of the process, 30,000 point samples were generated, with balanced amounts between pasture and non-pasture (Figure 3) that was representative of the study area and large enough to accommodate the increasing number of data dimensions in order to meet best training practices for supervised classifiers [19]. Considering the generated samples, 90% of the points were used for training and 10% were used for validation of the random forest and LSTM algorithms (the utilized datasets are available at: https://www.lapig.iesa.ufg.br/drive/index.php/s/DE3vjlUugyhEOwD).

**Table 1.** Number of polygons, collected by specialists via visual interpretation, with their respective classes and subclasses. Based on these polygons, we calculated the total amount of PlanetScope pixels, accounting for a 4-m spatial resolution, and the minimum distance necessary to generate 30,000 training samples according to Equation (1). LULC: land use and land cover.

| LULC Class | | Polygons | | Pixels (4 m) | | Min. Distance | Point Samples | |
|---|---|---|---|---|---|---|---|---|
| **Class** | **Subclass** | **Quantity** | **Percent** | **Quantity** | **Percent** | | **Subclass** | **Class** |
| Non-pasture | Sand Bank | 50 | 3.77% | 9764 | 1.04% | 6.205 | 1500 | 15,000 |
| | Planted Forest | 36 | 2.71% | 32,871 | 3.51% | 14.725 | 1500 | |
| | Water | 89 | 6.71% | 36,684 | 3.92% | 15.781 | 1500 | |
| | Urban Area | 143 | 10.78% | 55,839 | 5.97% | 20.405 | 1500 | |
| | Deforestation | 145 | 10.94% | 69,821 | 7.46% | 23.290 | 1500 | |
| | Others | 85 | 6.41% | 131,552 | 14.06% | 33.460 | 1500 | |
| | Forest Formation | 111 | 8.37% | 116,957 | 12.50% | 31.321 | 1500 | |
| | Savannah Formation | 140 | 10.56% | 131,422 | 14.04% | 33.441 | 1500 | |
| | Crop | 234 | 17.65% | 178,509 | 19.07% | 39.636 | 1500 | |
| Pasture | Regular Pasture | 42 | 3.17% | 42,213 | 4.51% | 9.420 | 3750 | 15,000 |
| | Wet Pasture | 87 | 6.56% | 52,951 | 5.66% | 11.031 | 3750 | |
| | Pasture with bare soil | 39 | 2.94% | 33,226 | 3.55% | 7.906 | 3750 | |
| | Pasture with shrubs | 125 | 9.43% | 44,034 | 4.71% | 9.707 | 3750 | |
| Total | | 1326 | 100.00% | 935,843 | 100.00% | | | 30,000 |

### 2.2.2. Sampling Segments

The generation of segment samples, necessary for the training of semantic segmentation algorithms (e.g., U-Net), considered a regular grid over the study area that was composed of ≈14,300 cells with a cell size defined by a function of the established input for the U-Net in this study (i.e., 286 × 286 pixels; Figure 3). Referring to this regular grid, all cells containing at least one polygon collected by the experts were selected, resulting in 1440 cells. The area delimited by these cells (i.e., ≈1800 km$^2$) was segmented via the mean shift algorithm and the Orfeo Toolbox/Monteverdi framework [39], and classified via visual interpretation in the Quantum GIS software [40], using the PlanetScope Mosaic of November 2017 as a reference.

In order to generate pasture segments capable of aggregating visible objects in very high-resolution images (e.g., isolated trees, patches of bare soil), the following parameters were defined for the mean shift algorithm: tile size of 286 × 286, spatial radius of 24, range radius of 128, and a minimum segment size of 4 (all in pixel units). The choice of the November 2017 mosaic was justified by the greater vegetative vigor of the pasture areas due to the beginning of the rainy season, and the presence of bare soil in the agricultural areas due to the beginning of the cropping cycle, factors that generally facilitate classification based on segmentation and visual interpretation. This process resulted in ≈937,000 segments, with 36% of the segments classified as pasture, which were converted to a raster format compatible with the PlanetScope mosaics (i.e., same pixel size and geographic coordinate system; Figure 3). Considering the total number of selected cells and their respective segments in the raster format, 90% of the cells were used for training and 10% were used for validation of the U-Net algorithm (the utilized datasets are available at https://www.lapig. iesa.ufg.br/drive/index.php/s/DE3vjlUugyhEOwD).

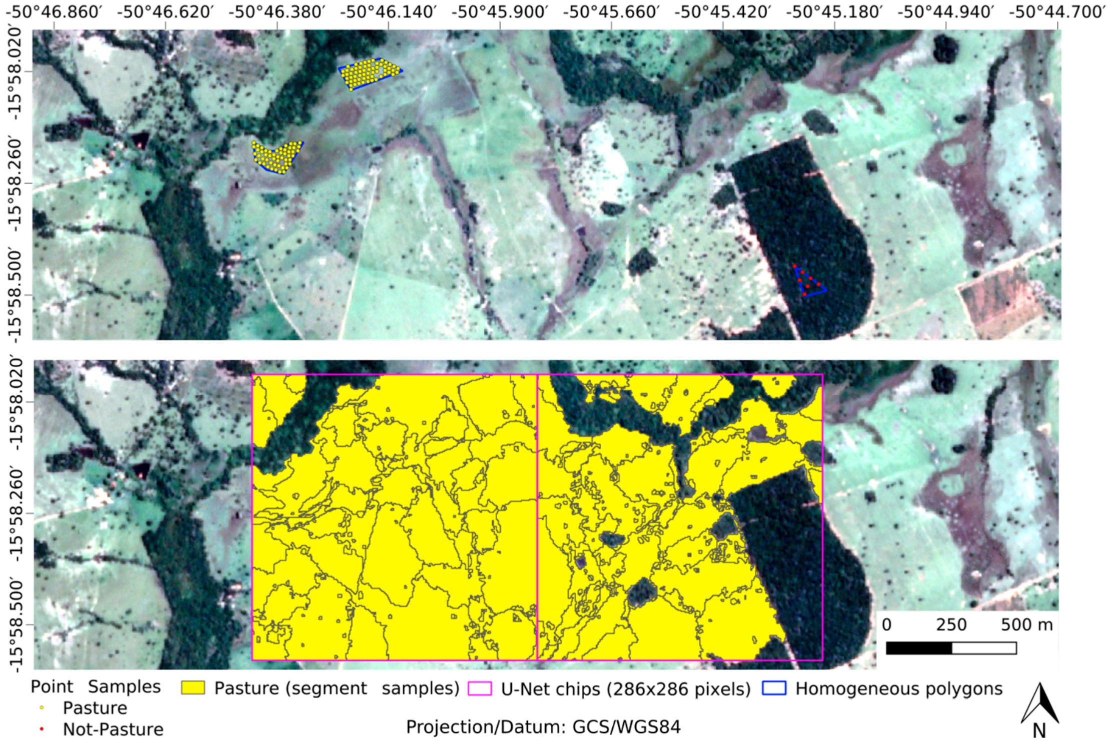

**Figure 3.** Examples of homogeneous polygons (i.e., with pixels of only one LULC class) and reference samples used in this study. It is noteworthy that point and segment samples have very different data structures.

### 2.2.3. Field Data

Field data were obtained for three different periods between 2017 and 2018, comprising pasture and non-pasture regions. This study considered a total of 262 samples containing geographic coordinates, LULC classes (i.e., "regular pasture," "well-managed pasture," "pasture with bare soil," "pasture with shrubs," "forest formation," "savannah formation," "crop," "sugarcane," "planted forests," "urban area," and "water"), area characterization information, and field pictures. These data, relatively concentrated in the southwestern portion of the study area (Figure 2b), were used as test data in the accuracy analysis of the produced mappings.

## *2.3. Random Forest Classification*

The random forest training utilized a set of texture-spectral and temporal metrics generated from the monthly mosaics and 27,000 point samples to build 100 statistical decision trees. The growth of each tree considered 50% of the training samples as a bag fraction, six random spectral metrics to be selected in the definition of the best split node (Mtry), and one sample as a minimum leaf population of a terminal node. Considering its low sensitivity to overfitting, the most common recommendation for a random forest is to set a large number of trees and the square root of the feature space size as the number of variables for splitting [19]. The set of metrics was defined using a feature engineering strategy, which evaluated different time windows and opted for the one that produced the best pasture mapping of the study area according to the validation samples. All of these processing steps were implemented on the Google Earth Engine platform and are available at: https://code. earthengine.google.com/2bf9aebba03eac0d1e55d79731847cff.

Feature Engineering

The feature engineering used all the spectral information available on the PlanetScope images (i.e., the blue, green, red, and near-infrared bands), as well as the normalized difference vegetation index (NDVI) [41], the photochemical reflectance index (PRI) [42], and the normalized difference water index (NDWI) [43] to generate a set of spectral-temporal metrics, an approach that is already consolidated in pasture operational mappings with Landsat [15] and MODIS [44] satellite data. The temporal aspect was incorporated via statistical operations that reduced multiple monthly mosaics to a single spectral value in order to capture the seasonal aspects of pasture and non-pasture pixels. This process used the median and three other statistical reducers derived from the standard deviation (i.e., $+2\sigma$ to obtain maximum values, $-2\sigma$ to obtain minimum values, and $4\sigma$ to obtain the amplitude of the spectral variation), yielding 28 spectral-temporal metrics overall.

Aiming to improve the classification efficiency of very high spatial resolution pixels, the feature engineering also considered texture metrics capable of capturing spatial and topological aspects. We applied these metrics on the green, red, and near-infrared spectral bands using the gray-level co-occurrence matrix (GLCM) method and a 3 × 3 pixel spatial window [45]. Based on the assessment of the GLCM result, we opted for the use of contrast, second angular momentum, and entropy metrics, which are capable of highlighting transition zones, homogeneous regions, and the degree of disorder in the land-use and land-cover classes in the landscape, respectively. The application of these metrics was done for multiple monthly mosaics, and at the end of the process, a median reducer was used in each spectral band to generate nine texture metrics.

The feature space, composed of these 37 variables (i.e., 28 spectral-temporal metrics and 9 texture metrics), was generated for 9 distinct time windows, considering only one month (i.e., November and March); every quarter (i.e., August–October, November–January, February–April, and May–July); two semesters (i.e., August–January and February–July); and 12 months (i.e., August–July) of the analyzed period. Specifically, for the one-month classifications, 16 metrics were considered (i.e., four spectral bands, three spectral indices, and nine texture metrics). This process generated nine feature spaces, and consequently, nine classification results, which were evaluated through the validation set derived from the point samples. The result that presented the highest F1 score, which was a metric capable of simultaneously considering omission and commission errors [46], was considered as the best mapping produced with the random forest, which was used in the accuracy analysis and in the comparison with the other pasture mappings (i.e., produced via deep learning approaches).

### *2.4. Long Short-Term Memory Classification*

The implementation of the LSTM [36] classification approach considered a neural network architecture that simultaneously analyzed the spectral, spatial, and temporal dimensions of all monthly PlanetScope mosaics (Figure 4). The LSTM bidirectional layer processed the spectral and temporal dimensions by analyzing the two-way time series (i.e., from beginning to end, and from end to beginning) in order to seek seasonal patterns capable of separating pasture and non-pasture pixels. The first 3D convolution layer used 256 kernels with a 3 × 3 × 5 size (i.e., a 3 × 3 spatial window over five months) to process the PlanetScope monthly mosaics, maintaining the same time-series size as the output. The second 3D convolution layer used a valid padding strategy to transform the 3D time-series into a 2D time-series, which was later concatenated with the output of the LSTM layer and processed using a set of fully-connected layers. Prior to concatenation, a batch normalization operations [47] was applied, and between the fully-connected layers, a dropout rate of 0.3 [48] was applied, which generally decreased the chance of overfitting during model training. Because it is a binary classification, the output layer used the sigmoid activation function, while the other layers used the ReLu function.

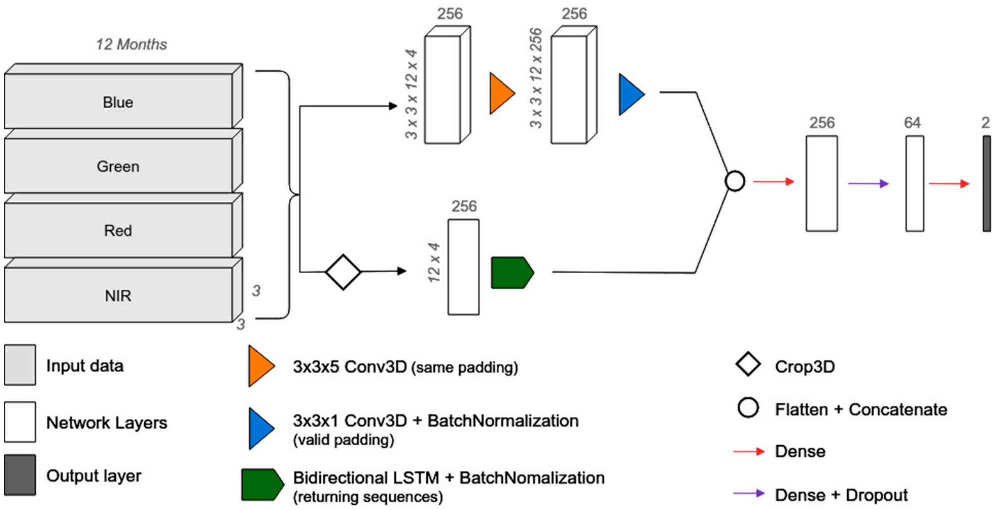

**Figure 4.** Neural network architecture used to classify pasture areas in the study area. The LSTM layer was responsible for analyzing the temporal dimension by considering all PlanetScope mosaics, while the convolutional layers analyzed the spatial dimension through a 3 × 3 pixel window; the spectral dimension (i.e., blue, green, red, and near-infrared) was analyzed by both layers.

The training of the neural network occurred for 100 epochs with a learning rate of $5 \times 10^{-5}$, the binary cross-entropy as a cost function, and the Nadam optimizer, which incorporates the nesterov momentum in the Adam optimizer [49]. In order to make possible the comparison between the results produced by this approach and the random forest, we used the same set of training and validation with, respectively, 27,000 and 3000 point samples, with a mini-batch of size 16. Finally, the trained model was used to classify all the pixels in the study area and produce a pasture map for the region.

All the processing steps were performed on a local server with an Intel Xeon E5-2620 v2 processor, 48 GB of memory, and two GPU cards (NVIDIA Titan-X and NVIDIA 1080ti). The neural network was developed in Keras using Tensorflow, with the LSTM implementation in NVIDIA cuDNN [50,51], while the other operations involving the PlanetScope mosaics (e.g., access and preparation of the time-series) made use ofGeospatial Data Abstraction Library-GDAL and NumPy libraries [52]. The source code used by this classification approach can be accessed at https://github.com/NexGenMap/dl-time-series.

### 2.5. U-Net Classification

We used the original U-Net architecture [37] with some modifications to consider the temporal dimension and a set of more recent hyperparameters (Figure 5). The incorporation of time occurred by stacking the spectral bands of the March and November monthly mosaics. The use of the November mosaic was expected as it was used in the generation of the segment samples, while the option for the March mosaic was justified by its relevance in the rainy season. In March, the pastures generally show a higher vegetative vigor and some agricultural areas are at the peak of cultivation, showing complementary aspects to those observed in November, which can therefore contribute to a better performance of the U-Net regarding pasture discrimination.

With regard to the new hyperparameters, all convolutional layers considered the batch normalization operation [47] and an L2 regularization of 0.5 [53], while specifically for the penultimate layer, a dropout rate of 0.5 was utilized [49]. The cost function used was the intersection-over-union, which is generally better suited to binary semantic segmentation problems [54], and the optimizer chosen was Nadam [50]. We chose this cost function to minimize the impact of the class imbalance in the segment samples, which is a common issue of binary semantic segmentation problems, with the aim of enabling a fair comparison with the other classification approaches (i.e., random forest and LSTM-based architecture) that used balanced point samples (Table 1). With a learning rate of $5 \times 10^{-5}$ and a 32-size mini-batch, the neural network training occurred for 100 epochs and considered all segment samples in raster format to generate the input chips (i.e., a set of pixels with a regular squared size), which were submitted to two traditional data augmentation strategies, three rotations (i.e., 90°, 180°, and 270°), and a flip operation (i.e., left–right [55]).

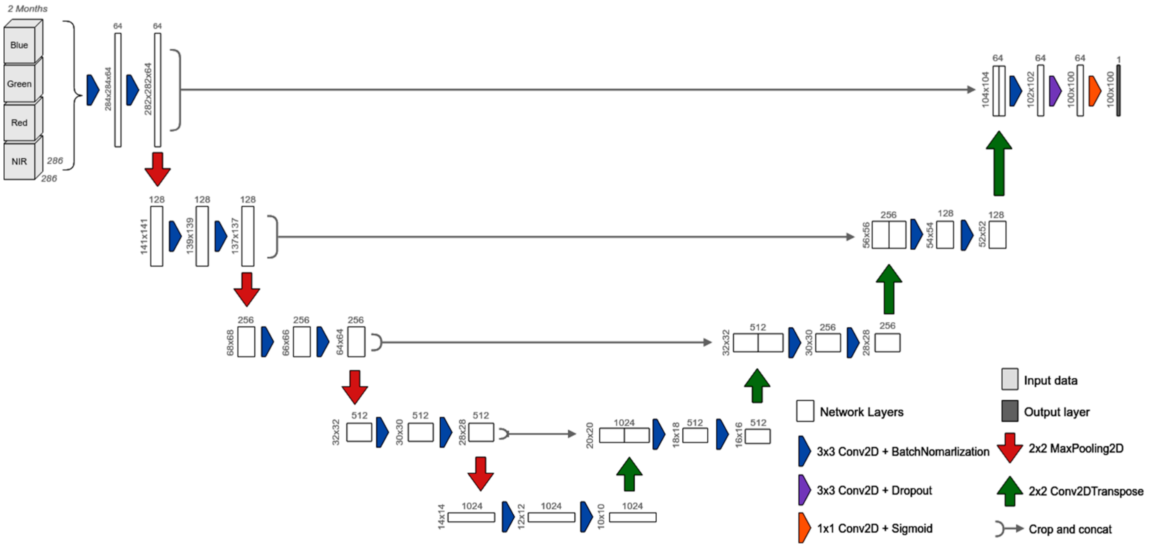

**Figure 5.** Semantic segmentation approach used to classify pasture areas in the study area. This architecture, derived from U-Net, analyzed two images via an early-fusion technique by simultaneously considering the spatial, temporal, and spectral dimensions.

Originally, the U-net was designed to receive, as an input, an image larger than its output layer, and for the sake of this work, this feature was maintained in the training of the model (i.e., 286 × 286 pixels input chips and 100 × 100 pixels output). By considering pixels adjacent to and external to the classification area, this strategy can yield more spatially consistent mapping results. In this sense, the trained model was applied throughout the study area by using a sequential scanning approach and a floating window of 100 × 100 pixels. For the classification of regions at the borders of the study area (e.g., upper-left and lower-right corners), the missing pixels needed for the input chip were artificially filled with a reflect technique implemented by the SciPy library [56]. The other processing steps were developed with the Tensorflow [51] and GDAL [52] libraries, while the training and classification were performed in the same hardware environment used by the LSTM. The source code used by this classification approach can be accessed at https://github.com/NexGenMap/dl-semantic-segmentation.

## 2.6. Accuracy Analysis

The accuracy analysis was performed for the three pasture mappings produced by considering the validation samples previously separated from the training samples and the test samples obtained in the field. For both sample sets, we calculated the producer (i.e., recall) and user (i.e., precision) accuracies, the F1 score (i.e., a metric calculated as a function of both the producer and user accuracies) [56], and the population estimate of the error matrix [30], calculated according to the pasture area mapped by each classification approach.

## 3. Results

In the following sections, we present the results regarding the random forest feature engineering, the training and prediction stages of the classification models, and the pasture mapping products, with their respective accuracies.

### 3.1. Feature Engineering

The feature engineering step revealed that the best pasture area mapping, according to the F1 score, was produced using all the monthly mosaics (Figure 6). In general, this result was expected because the larger the temporal window used in the construction of the spectral-temporal and texture metrics, the greater the variability of the feature space analyzed by the random forest, which tends to increase its ability to discriminate between pasture and non-pasture pixels. On the other hand, it is worth mentioning that some mappings produced with only three months, particularly with the periods of November–January and February–April, presented an F1 score of ≈93%, which are very close to the values obtained with a semester of observations. This was an indication that the most relevant metrics for pasture mapping depended on images taken in the rainy season months (i.e., October to April), when pastures showed greater variation in the vegetative vigor and the agricultural areas changed from bare soil to the peak of the growing season. Finally, the worst mappings were produced with classifications that considered only one month, which had a feature space with fewer metrics and less spectral variability; however, March still showed a higher F1 score than November.

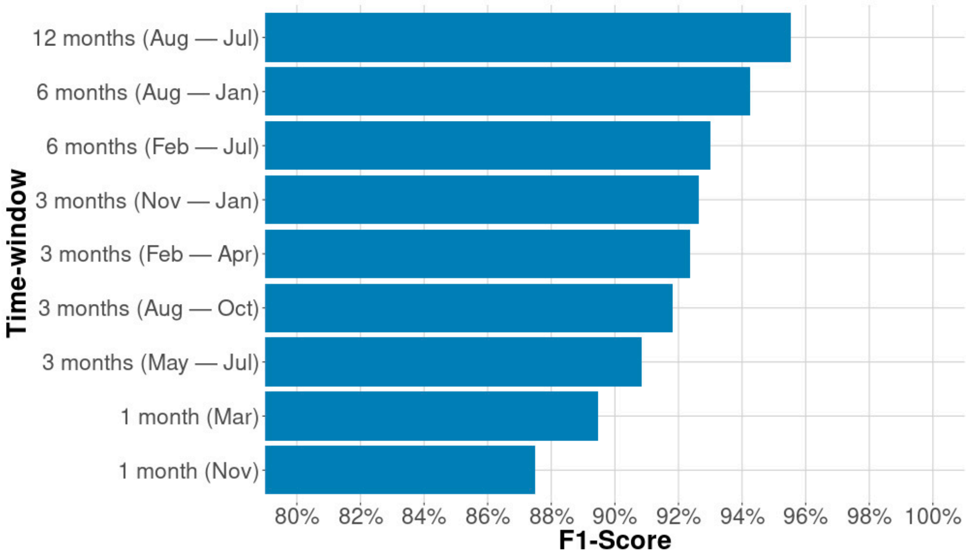

**Figure 6.** F1 score of random forest pasture area classifications produced with different time windows and the same set of texture and spectral-temporal metrics. Although the best classification result used the 12 PlanetScope monthly mosaics, classifications using just three months had a high F1 score.

### 3.2. Training and Prediction Performance

Considering 100 epochs, LSTM and U-Net training took ≈0.5 and ≈24 hours, respectively. This large difference between the training of the two models may be related to the larger amount of data analyzed by U-Net. Because U-Net is a semantic segmentation approach, its samples are organized into data blocks of 286 × 286 pixels, while LSTM samples are 3 × 3 pixels in size. In this regard, if we consider the data structure of the training samples (i.e., height, width, spectral band, and time) and their respective data augmentation cases, U-Net and LSTM analyzed ≈7538 and ≈12 million pixels, respectively (Table 2). On the other hand, it is interesting to note that in the prediction stage, this situation was reversed, and LSTM started to analyze more data than U-Net (Table 2). Because it is a pixel-level time-series analysis approach, LSTM needs to work with more monthly tiles than U-Net, which ultimately consumes more processing time when sorting all the study area pixels. In this process, the total prediction times of LSTM and U-Net were, respectively, ≈23 hours and ≈1.2 hours.

**Table 2.** Total amount of PlanetScope pixels, with a 4-m spatial resolution, analyzed in the training and prediction stages by the deep learning classification approaches. U-Net analyzed a larger data volume at the training stage, while LSTM processed more pixels at the prediction stage.

| # | Training Data | | Predicted Data | |
|---|---|---|---|---|
| | **U-Net** | **LSTM** | **U-Net** | **LSTM** |
| Data chunk | 1440 | 27,000 | 116,183 | 1,161,832,926 |
| Augmentation cases | 8 | - | - | - |
| Height | 286 | 3 | 286 | 3 |
| Width | 286 | 3 | 286 | 3 |
| Spectral Bands | 4 | 4 | 4 | 4 |
| Times | 2 | 12 | 2 | 12 |
| Total (million pixels) | 7538 | 12 | 76,027 | 501,912 |

Regarding the loss function, the LSTM achieved a loss of 0.0405 for the training data and 0.0395 for the validation data at the end of the training (Figure 7a), while U-Net obtained 0.0293 and 0.0842 for the data training and validation, respectively (Figure 7b). The loss curves for LSTM and U-Net presented decreasing trends, with a relative constancy in the distances between the training and validation over the epochs, suggesting a generalization capacity of the classification models.

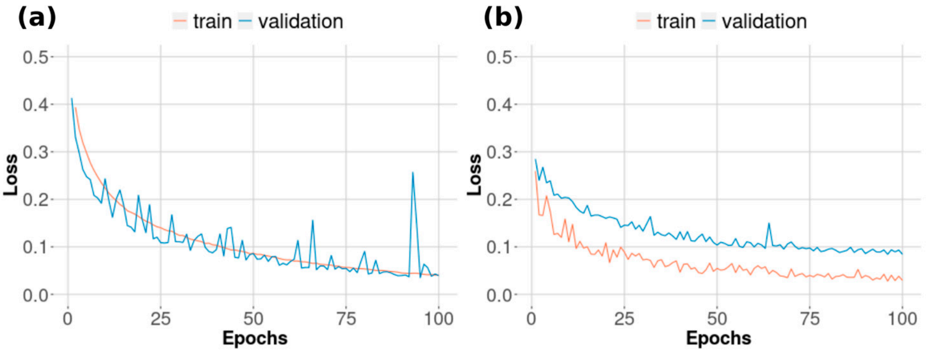

**Figure 7.** Training loss curves of the LSTM (**a**) and U-Net (**b**) classification approaches. All curves, referring to training and validation data, showed a downward trend along the 100 epochs.

Specifically for the random forest, the amount of pixels analyzed in the classification of the entire study area was equivalent to 42,987 million (i.e., 37 metrics × 1161 million pixels), such that the Google Earth Engine performed the training and prediction steps together in ≈3 hours. However, it makes little sense to compare the random forest runtime with LSTM/U-Net since the models ran in very different computing environments, with the implementation of random forest on a cloud-based platform and the other models on a single local server.

### 3.3. *Pasture Area Mapping*

The three mappings over an area of 18,000 km$^2$, using the random forest (i.e., 12-month classification), LSTM, and U-Net algorithms identified ≈7700, ≈8700, and ≈11,141 km$^2$ of pasture area, respectively (Figure 8). Thus, it is clear that the U-Net semantic segmentation method mapped a larger pasture area than the other approaches. As the mappings were produced with very high-resolution images (i.e., 4 m), this result was relatively expected as random forest and LSTM do not map pixels with the presence of, for example, isolated trees and patches of bare soil. On the other hand, the segment samples contained pasture areas with these characteristics, which allowed the U-Net to analyze the spatial context of these pixels and map them correctly according to the land use class and surrounding land cover. Specifically for the random forest, it was observed that several agricultural regions were erroneously classified as pasture (Figure 8b), while LSTM identified pasture pixels located in savannah vegetation areas (Figure 8c). These two commission errors were generally not observed in the U-Net result, which, on the other hand, erroneously mapped gallery forest regions bordering pasture paddocks and pixels located in the urban area (Figure 8d).

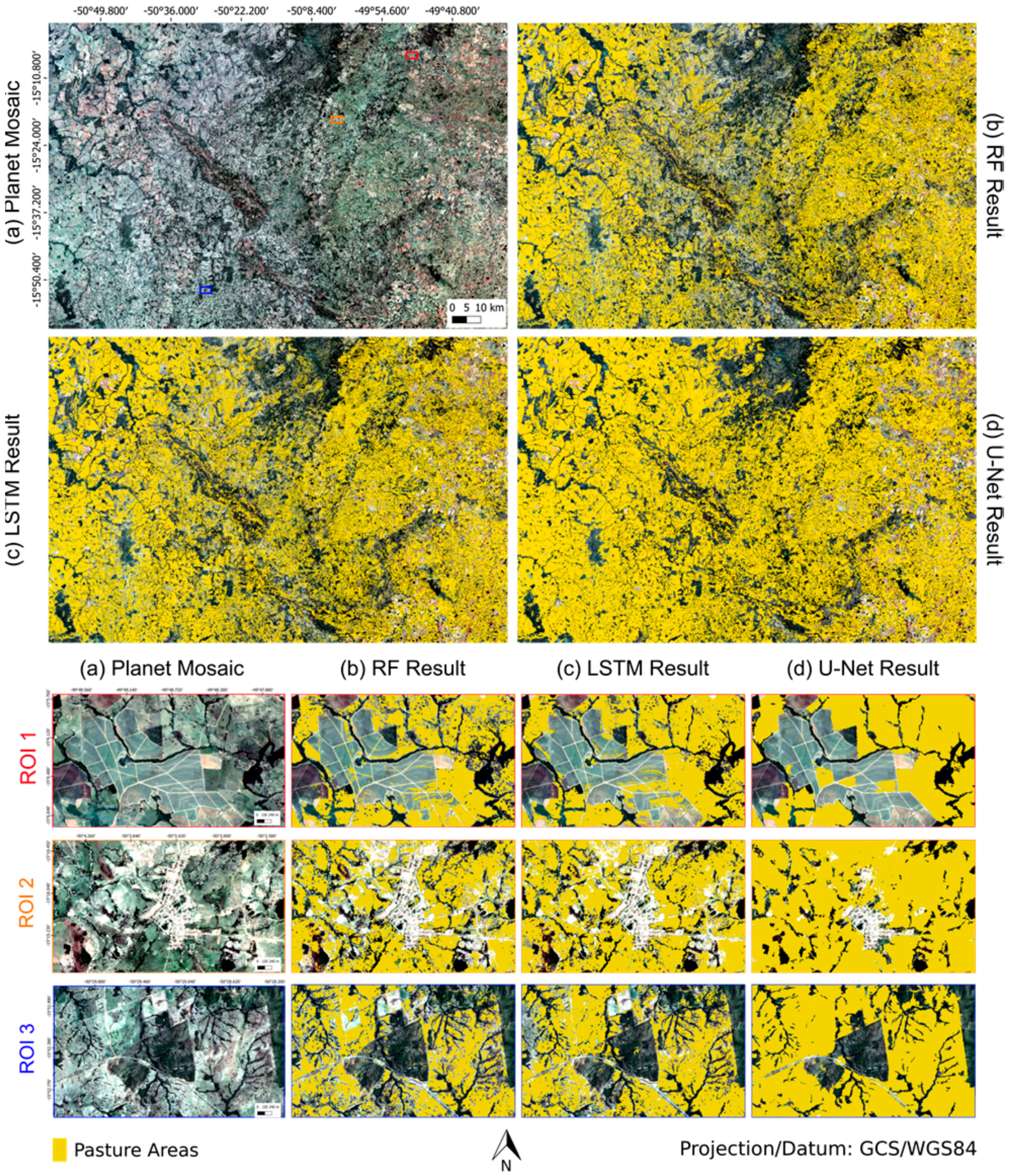

**Figure 8.** The monthly PlanetScope mosaic for November 2017 (**a**) and the pasture maps produced with random forest (**b**), LSTM (**c**) and U-Net (**d**). In the region of interest 1 (ROI 1), U-Net was better at separating agricultural areas, while in region 2 (ROI 2), LSTM and random forest distinguished urban area pixels the best. In region of interest 3 (ROI 3), U-Net better filled the pasture areas, but mis-mapped some gallery forest pixels.

Based on the validation set, producer and user accuracies presented close values in all pasture mappings, with the highest F1 score value of 98.83% for the LSTM result (Figure 9a). Assuming that the random forest and LSTM accuracy analyses used exactly the same samples, obtained under the same conditions, we can infer that for pasture mapping, LSTM is an approach with better classification efficiency. Although the U-Net and LSTM validation samples consisted of distinct and therefore not equivalent data structures, the fact that they were obtained under the same conditions (i.e., random selection prior to model training) allowed for some level of comparison between their respective F1 scores, which also revealed a better classification efficiency by LSTM.

In contrast, the accuracy analysis with the test set (Figure 9b) indicated a higher F1 score for the U-Net classification, equivalent to 94.06%, and did not present balanced values between producer and user accuracies, with a prevalence of commission errors for the results generated by the LSTM and random forest. These differences may be justified by the fact that the test set was obtained in the field, and consequently had a smaller representation than the validation samples, which was obtained by visual interpretation and was better distributed over the study area. On the other hand, relatively close F1 score values in both assessment sets may indicate a better generalization capacity of U-Net in relation to the other classification approaches.

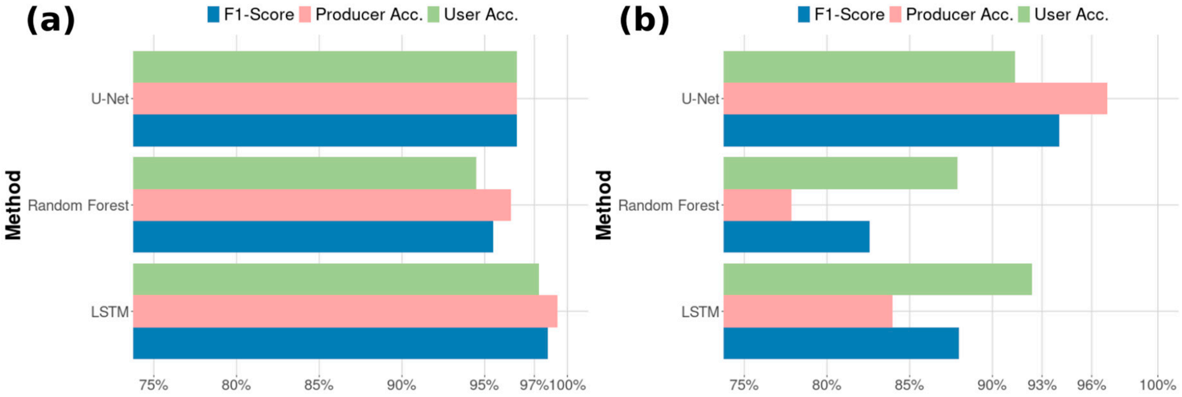

**Figure 9.** Accuracy analysis results with validation data (**a**) collected via visual interpretation and tests (**b**) obtained in the field. The highest F1 score values were obtained using LSTM in the validation data and by U-Net in the test data. Relatively close F1 score values in both assessment sets may indicate a better generalization of U-Net.

## 4. Discussion

Given the possibility of applying conventional and state-of-the-art machine learning algorithms to large volumes of satellite data, this study used different classification approaches to produce and evaluate three pasture maps for an area of 18,000 km$^2$. The use of images obtained by the PlanetScope constellation required a sampling strategy compatible with very high spatial resolution data that was representative for the study area. Given the classification results obtained, we can assume that the chosen strategy was adequate since the mapping products achieved F1 scores greater than 95% for the validation data and 82% for the test data. We believe that part of the success of these results can be attributed to the representativeness of the LULC classes and subclasses, defined by experts with knowledge of the study area, which eventually allowed for the algorithms to generate more generic and comprehensive classification rules to separate between pasture and non-pasture pixels. In this sense, we assume that the classes and subclasses identified by visual interpretation could also be differentiated by some machine learning approach.

On the other hand, the need for knowledge regarding the mapping region for the definition of classes and subclasses can be considered a limitation of this sample collection strategy, especially with regard to large-scale mappings. The use of an unsupervised classification approach [57] could be a possible solution to this lack of knowledge, allowing specialists to define regular and homogeneous polygons within a set of automatically defined classes. Another possible modification would be to include samples with some level of LULC class mixing, as conventional classification algorithms can handle different levels of heterogeneity in training samples, although this practice does not bring significant accuracy gains [58,59].

Specifically in terms of the feature engineering, the relationship between the best classification results with random forest and the monthly mosaics obtained in the rainy season was an interesting finding. This was because, in general, the images have a high cloud coverage during this period, and we can say that the high temporal resolution of the PlanetScope data was crucial for obtaining full coverage and therefore cloud-free mosaics for every month. Regarding the spectral resolution, we recognize that despite the low radiometric quality and lack of calibration of the PlanetScope constellation, a common characteristic in CubeSats, the four spectral bands were sufficient at generating satisfactory mapping products, as revealed by the accuracy analysis, which is an indication that the spectral and radiometric limitations were offset by the very high temporal and spatial resolutions.

Among the classification approaches evaluated, U-Net was the one that best used the spatial resolution of the PlanetScope data. Because it is a semantic segmentation approach, it was able to better populate the regions that were effectively pastures, in general producing a more spatially consistent mapping than the other methods. On the other hand, its use depended on segment samples for the totality of the input data, which usually required a greater effort of visual interpretation and generation compared to the point samples.

Specifically regarding the modifications we implemented in the U-Net, we believe that incorporating the temporal dimension with early-fusion and simultaneously analyzing two images contributed to a better identification of pasture pixels since the images used (i.e., November and March) contained complementary spectral data related to the seasonal and occupation dynamics of the study area. As the selection of these images made use of knowledge of the mapping region, for a scenario of application in large geographic regions, the input images could be automatically defined using statistical criteria (e.g., variation of NDVI values over a temporal window [33], or all available images could be effectively used as input [28]). However, the use of all images obtained in, for example, 12 months could increase the cost of segment sample generation as it would be necessary to consider their respective spatial variations during this period due to landscape dynamics and LULC changes [60,61].

In this scenario, the classification approach with the LSTM might be more appropriate because using a 3 × 3 pixel window and all available images, it considers both the spatial and temporal dimensions, such that its training samples were less affected by landscape dynamics and LULC changes. In this sense, our implementation allowed for the spatial window to be expanded to better consider the geographical context of the analyzed pixels. In fact, for this modification to improve the spatial consistency of the pasture classification results, our point sample set would have to be expanded beyond pure pixels (i.e., unmixed LULC class pixels).

Of all the evaluated approaches, the LSTM was the least dependent on some prior knowledge of the region and/or mapping classes, characteristics that make it very suitable for large-scale operational mappings. It is noteworthy that its classification efficiency has the potential to improve with a larger number of training samples, something that could be investigated by future studies. Because it is a per-pixel time-series approach, this classification model analyzed a data volume substantially larger than U-Net in the prediction step. Since, within the scope of satellite data classification, this step is embarrassingly parallel [62], its execution time could be dramatically reduced through a high performance processing solution (e.g., in-house grid computing, Google Earth Engine).

## 5. Conclusions

The rapid growth of satellites orbiting the planet was responsible for generating a massive volume of data on the earth's surface with the most varied scales and characteristics. At the same time, new deep-learning-based classification algorithms have been developed and are considered the state-of-the-art in many application domains. Given the possibility of using these algorithms on this growing volume of satellite data, this work evaluated three classification approaches based on PlanetScope images and the operational generation of mapping products for the pasture class. Our results indicated a better classification efficiency using deep learning approaches (i.e., LSTM and U-Net) over random forest, a conventional machine learning approach. However, additional comparison among these approaches are necessary, involving other studies areas and LULC classes. Nevertheless, considering the difficulty involved in the identification of pasture areas and the results obtained, we can infer that these classification approaches could be used to map other LULC classes (e.g., crops, planted forest, forest formations, water), taking into account new efforts to collect reference samples and training specialized models according to the classes of interest.

Likewise, and despite the fact that the generalization of these approaches to other geographic regions and/or mapping periods requires further investigations, the neural network architectures developed in this study could be used to map large geographic regions (e.g., the entire Brazilian territory), considering private (e.g., PlanetScope) and public (e.g., Sentinel 2, Landsat 8) satellite data. In this scenario, the strategy that we defined could be used to generate samples for the entire region and/or mapping period, allowing the training of classification models to use some level of temporal and/or geographic stratification, which is a procedure previously used to map pastures with random forest [15,33]. Finally, the effective rationalization of such classification approaches depends on high-performance processing solutions capable of combining GPU processing and access to a large volume of satellite data, which is likely to be available to governmental, private and research organizations in the next few years.

**Author Contributions:** L.P., E.T., and C.S.J. conceived the idea and the methodological approach of the study. L.P., E.T., and A.P.S. processed the data, and together with L.F. and C.S.J., analyzed the results. All the authors contributed to the discussions and writing or revision of the manuscript. C.S.J. was responsible for the funding acquisition.

**Funding:** This work, part of the MapBiomas and NexGenMap initiatives, was supported by the Gordon and Betty Moore Foundation (GBMF), The Nature Conservancy (TNC), the Brazilian Research Council (CNPq), and the Coordination for the Improvement of Higher Education Personnel (CAPES).

**Acknowledgments:** We thank the technical assistance of Planet and Google to implement the workflow to ingest the PlanetScope imagery dataset into Earth Engine. We are also thankful to Tara O'Shea from Planet, and Brian Sullivan from Google, Leonardo Fleck from GBMF, and Tasso Azevedo from MapBiomas, for their constructive feedback during the development of this project.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Liu, P. A survey of remote-sensing big data. *Front. Environ. Sci.* **2015**, *3*, 45. [CrossRef]
2. Houborg, R.; McCabe, M.F. A Cubesat enabled Spatio-Temporal Enhancement Method (CESTEM) utilizing Planet, Landsat and MODIS data. *Remote Sens. Environ.* **2018**, *209*, 211–226. [CrossRef]
3. Chi, M.; Plaza, A.; Benediktsson, J.A.; Sun, Z.; Shen, J.; Zhu, Y. Big Data for Remote Sensing: Challenges and Opportunities. *Proc. IEEE* **2016**, *104*, 2207–2219. [CrossRef]
4. Nemani, R.; Votava, P.; Michaelis, A.; Melton, F.; Milesi, C. Collaborative supercomputing for global change science. *Eos Trans. Am. Geophys. Union* **2011**, *92*, 109–110. [CrossRef]
5. Gorelick, N.; Hancher, M.; Dixon, M.; Ilyushchenko, S.; Thau, D.; Moore, R. Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sens. Environ.* **2017**, *202*, 18–27. [CrossRef]
6. Yan, J.; Ma, Y.; Wang, L.; Choo, K.K.R.; Jie, W. A cloud-based remote sensing data production system. *Future Gener. Comput. Syst.* **2018**, *86*, 1154–1166. [CrossRef]
7. Wang, L.; Ma, Y.; Yan, J.; Chang, V.; Zomaya, A.Y. pipsCloud: High performance cloud computing for remote sensing big data management and processing. *Future Gener. Comput. Syst.* **2018**, *78*, 353–368. [CrossRef]
8. Pekel, J.F.; Cottam, A.; Gorelick, N.; Belward, A.S. High-resolution mapping of global surface water and its long-term changes. *Nature* **2016**, *540*, 418–422. [CrossRef]

9. Gong, P.; Wang, J.; Yu, L.; Zhao, Y.; Zhao, Y.; Liang, L.; Niu, Z.; Huang, X.; Fu, H.; Liu, S.; et al. Finer resolution observation and monitoring of global land cover: First mapping results with Landsat TM and ETM+ data. *Int. J. Remote Sens.* **2013**, *34*, 2607–2654. [CrossRef]
10. Hansen, M.C. High-Resolution Global Maps of. *Science* **2013**, *850*, 850–854. [CrossRef]
11. Oliphant, A.J.; Thenkabail, P.S.; Teluguntla, P.; Xiong, J.; Gumma, M.K.; Congalton, R.G.; Yadav, K. Mapping cropland extent of Southeast and Northeast Asia using multi-year time-series Landsat 30-m data using a random forest classifier on the Google Earth Engine Cloud. *Int. J. Appl. Earth Obs. Geoinf.* **2019**, *81*, 110–124. [CrossRef]
12. Xu, Y.; Yu, L.; Feng, D.; Peng, D.; Li, C.; Huang, X.; Lu, H.; Gong, P. Comparisons of three recent moderate resolution African land cover datasets: CGLS-LC100, ESA-S2-LC20, and FROM-GLC-Africa30. *Int. J. Remote Sens.* **2019**, *40*, 6185–6202. [CrossRef]
13. Graesser, J.; Ramankutty, N. Detection of cropland field parcels from Landsat imagery. *Remote Sens. Environ.* **2017**, *201*, 165–180. [CrossRef]
14. Griffiths, P.; Nendel, C.; Hostert, P. Intra-annual reflectance composites from Sentinel-2 and Landsat for national-scale crop and land cover mapping. *Remote Sens. Environ.* **2019**, *220*, 135–151. [CrossRef]
15. Parente, L.; Mesquita, V.; Miziara, F.; Baumann, L.; Ferreira, L. Assessing the pasturelands and livestock dynamics in Brazil, from 1985 to 2017: A novel approach based on high spatial resolution imagery and Google Earth Engine cloud computing. *Remote Sens. Environ.* **2019**, *232*, 111301. [CrossRef]
16. Yan, L.; Roy, D.P. Conterminous United States crop field size quantification from multi-temporal Landsat data. *Remote Sens. Environ.* **2016**, *172*, 67–86. [CrossRef]
17. Hansen, M.C.; Loveland, T.R. A review of large area monitoring of land cover change using Landsat data. *Remote Sens. Environ.* **2012**, *122*, 66–74. [CrossRef]
18. Jordan, M.I.; Mitchell, T.M. Machine learning: Trends, perspectives, and prospects. *Science* **2015**, *349*, 255–260. [CrossRef]
19. Belgiu, M.; Drăgu, L. Random forest in remote sensing: A review of applications and future directions. *ISPRS J. Photogramm. Remote Sens.* **2016**, *114*, 24–31. [CrossRef]
20. Kotsiantis, S.B.; Zaharakis, I.; Pintelas, P. Supervised Machine Learning: A Review of Classification Techniques S. *J. Manuf. Sci. Eng. Trans. ASME* **2007**, *125*, 394–395.
21. Zhong, L.; Hu, L.; Zhou, H. Deep learning based multi-temporal crop classification. *Remote Sens. Environ.* **2019**, *221*, 430–443. [CrossRef]
22. 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]
23. 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]
24. Lecun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
25. Chen, Y.; Lin, Z.; Zhao, X.; Wang, G.; Gu, Y. Deep learning-based classification of hyperspectral data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 2094–2107. [CrossRef]
26. Gerke, M. *Use of the Stair Vision Library within the ISPRS Use of the Stair Vision Library within the ISPRS 2D*; ResearcheGate: Berlin, Germany, 2015.
27. Yang, Y.; Newsam, S. Bag-Of-Visual-Words and Spatial Extensions for Land-Use Classificatio. In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, San Jose, CA, USA, 2–5 November 2010.
28. Stoian, A.; Poulain, V.; Inglada, J.; Poughon, V.; Derksen, D. Land Cover Maps Production with High Resolution Satellite Image Time Series and Convolutional Neural Networks: Adaptations and Limits for Operational Systems. *Remote Sens.* **2019**, *11*, 1986. [CrossRef]
29. Maulik, U.; Chakraborty, D. Remote Sensing Image Classification: A survey of support-vector-machine-based advanced techniques. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 33–52. [CrossRef]
30. Stehman, S.V.; Foody, G.M. Key issues in rigorous accuracy assessment of land cover products. *Remote Sens. Environ.* **2019**, *231*, 111199. [CrossRef]
31. Liu, P.; Yuan, T.; Ma, Y.; Wang, L.; Liu, D.; Yue, S.; Kołodziej, J. Parallel processing of massive remote sensing images in a GPU architecture. *Comput. Inform.* **2014**, *33*, 197–217.
32. Ma, Y.; Chen, L.; Liu, P.; Lu, K. Parallel programing templates for remote sensing image processing on GPU architectures: Design and implementation. *Computing* **2016**, *98*, 7–33. [CrossRef]

33. Parente, L.; Ferreira, L.; Faria, A.; Nogueira, S.; Araújo, F.; Teixeira, L.; Hagen, S. Monitoring the brazilian pasturelands: A new mapping approach based on the landsat 8 spectral and temporal domains. *Int. J. Appl. Earth Obs. Geoinf.* **2017**, *62*, 135–143. [CrossRef]
34. Saraiva, M.; Silva, D.; Ferreira, L.; Galano, S.; Siqueira, J.; Souza, C., Jr. Construção De Mosaicos Temporais Normalizados De Imagens Planet. In *Proceedings of the XIX Brazilian Symposium on Remote Sensing*; INPE: Santos, Brazil, 2019; pp. 3053–3056.
35. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
36. Schmidhuber, J.; Hochreiter, S. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780.
37. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. *Lect. Notes Comput. Sci.* **2015**, *9351*, 234–241.
38. IBGE. *Mapas e Outros Materiais Cartográficos na Biblioteca Central do IBGE*; Diretoria de Formação e Aperfeiçoamento de Pessoal, B.C., Ed.; IBGE: Rio de Janeiro, Brazil, 1983; Volume 2, ISBN 8524000112.
39. Teodoro, A.C.; Araujo, R. Comparison of performance of object-based image analysis techniques available in open source software (Spring and Orfeo Toolbox/Monteverdi) considering very high spatial resolution data. *J. Appl. Remote Sens.* **2016**, *10*, 016011. [CrossRef]
40. Jaya, M.T.S.; Fajar, A.N. Analysis of The Implementation Quantum GIS: Comparative Rffect and User Performance. *J. Theor. Appl. Inf. Technol.* **2019**, *97*, 2596–2605.
41. Huete, A.R.; Liu, H.Q.; Van Leeuwen, W. A Comparison of Vegetation Indices over a Global Set of TM Images for EOS-MODIS. *Remote Sens. Environ.* **1997**, *59*, 440–451. [CrossRef]
42. Gamon, J.A.; Serrano, L.; Surfus, J.S. The photochemical reflectance index: An optical indicator of photosynthetic radiation use efficiency across species, functional types, and nutrient levels. *Oecologia* **1997**, *112*, 492–501. [CrossRef]
43. Mcfeeters, S.K. The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features. *Int. J. Remote Sens.* **1996**, *17*, 1425–1432. [CrossRef]
44. Parente, L.; Ferreira, L. Assessing the spatial and occupation dynamics of the Brazilian pasturelands based on the automated classification of MODIS images from 2000 to 2016. *Remote Sens.* **2018**, *10*, 606. [CrossRef]
45. Wan, S.; Chang, S.H. Crop classification with WorldView-2 imagery using Support Vector Machine comparing texture analysis approaches and grey relational analysis in Jianan Plain, Taiwan. *Int. J. Remote Sens.* **2019**, *40*, 8076–8092. [CrossRef]
46. Li, W.; He, C.; Fang, J.; Zheng, J.; Fu, H.; Yu, L. Semantic segmentation-based building footprint extraction using very high-resolution satellite images and multi-source GIS data. *Remote Sens.* **2019**, *11*, 403. [CrossRef]
47. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
48. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 345–350.
49. Dozat, T. Incorporating Nesterov Momentum into Adam. In Proceedings of the ICLR Workshop, San Juan, Puerto Rico, 2–4 May 2016.
50. Dong, S.; Kaeli, D. DNNMark: A deep neural network benchmark suite for GPUs. In Proceedings of the General Purpose GPUs, Austin, TX, USA, 4–8 February 2017; pp. 63–72.
51. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A System for Large-Scale Machine Learning. In Proceedings of the 12th Symposium on Operating Systems Design and Implementation, Savannah, GA, USA, 2–4 November 2016; p. 44.
52. Qin, C.Z.; Zhan, L.J.; Zhu, A.X. How to Apply the Geospatial Data Abstraction Library (GDAL) Properly to Parallel Geospatial Raster I/O? *Trans. GIS* **2014**, *18*, 950–957. [CrossRef]
53. Jaiswal, S.; Mehta, A.; Nandi, G.C. Investigation on the Effect of L1 an L2 Regularization on Image Features Extracted using Restricted Boltzmann Machine. In Proceedings of the Second International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 14–15 June 2019; pp. 1548–1553.
54. Rahman, M.A.; Wang, Y. Optimizing intersection-over-union in deep neural networks for image segmentation. In *International Symposium on Visual Computing*; Springer: Cham, Switzerland, 2016; pp. 234–244.
55. Hadad, O.; Bakalo, R.; Ben-Ari, R.; Hashoul, S.; Amit, G. Classification of breast lesions using cross-modal deep learning. In Proceedings of the IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), Melbourne, Australia, 18–21 April 2017; pp. 109–112.

56. Lemaître, G.; Nogueira, F.; Aridas, C.K. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *J. Mach. Learn. Res.* **2016**, *18*, 559–563.
57. Sidorova, V.S. Detecting clusters of specified separability for multispectral data on various hierarchical levels. *Pattern Recognit. Image Anal.* **2014**, *24*, 151–155. [CrossRef]
58. Foody, G.M.; Mathur, A. The use of small training sets containing mixed pixels for accurate hard image classification: Training on mixed spectral responses for classification by a SVM. *Remote Sens. Environ.* **2006**, *103*, 179–189. [CrossRef]
59. Berhane, T.M.; Costa, H.; Lane, C.R.; Anenkhonov, O.A.; Chepinoga, V.V.; Autrey, B.C. The influence of region of interest heterogeneity on classification accuracy in wetland systems. *Remote Sens.* **2019**, *11*, 551. [CrossRef]
60. Lhermitte, S.; Verbesselt, J.; Jonckheere, I.; Nackaerts, K.; van Aardt, J.A.N.; Verstraeten, W.W.; Coppin, P. Hierarchical image segmentation based on similarity of NDVI time series. *Remote Sens. Environ.* **2008**, *112*, 506–521. [CrossRef]
61. Zhang, X.; Xiao, P.; Feng, X.; Yuan, M. Separate segmentation of multi-temporal high-resolution remote sensing images for object-based change detection in urban area. *Remote Sens. Environ.* **2017**, *201*, 243–255. [CrossRef]
62. Simpson, J.J.; MeIntire, T.J.; Berg, J.S.; Tsou, Y.L. The Parallel Image Processing Environment (PIPE): Automated parallelization of satellite data analyses. *Concurr. Comput. Pract. Exp.* **2007**, *19*, 685–701. [CrossRef]