



Article

Deep Recurrent Neural Network for Agricultural Classification using multitemporal SAR Sentinel-1 for Camargue, France

Emile Ndikumana ¹, Dinh Ho Tong Minh ^{1,*}, Nicolas Baghdadi ¹, Dominique Courault ² and Laure Hossard ³

¹ UMR TETIS, IRSTEA, University of Montpellier, 34093 Montpellier, France; emile.ndikumana@irstea.fr (E.N.); nicolas.baghdadi@irstea.fr (N.B.)

² UMR 1114 EMMAH, INRA, University of Avignon, 84914 Avignon, France; dominique.courault@inra.fr

³ UMR 0951 INNOVATION, INRA, University of Montpellier, 34060 Montpellier, France; laure.hossard@inra.fr

* Correspondence: dinh.ho-tong-minh@irstea.fr

Received: 5 July 2018; Accepted: 1 August 2018; Published: 3 August 2018



Abstract: The development and improvement of methods to map agricultural land cover are currently major challenges, especially for radar images. This is due to the speckle noise nature of radar, leading to a less intensive use of radar rather than optical images. The European Space Agency Sentinel-1 constellation, which recently became operational, is a satellite system providing global coverage of Synthetic Aperture Radar (SAR) with a 6-days revisit period at a high spatial resolution of about 20 m. These data are valuable, as they provide spatial information on agricultural crops. The aim of this paper is to provide a better understanding of the capabilities of Sentinel-1 radar images for agricultural land cover mapping through the use of deep learning techniques. The analysis is carried out on multitemporal Sentinel-1 data over an area in Camargue, France. The data set was processed in order to produce an intensity radar data stack from May 2017 to September 2017. We improved this radar time series dataset by exploiting temporal filtering to reduce noise, while retaining as much as possible the fine structures present in the images. We revealed that even with classical machine learning approaches (K nearest neighbors, random forest, and support vector machines), good performance classification could be achieved with F-measure/Accuracy greater than 86% and Kappa coefficient better than 0.82. We found that the results of the two deep recurrent neural network (RNN)-based classifiers clearly outperformed the classical approaches. Finally, our analyses of the Camargue area results show that the same performance was obtained with two different RNN-based classifiers on the *Rice* class, which is the most dominant crop of this region, with a F-measure metric of 96%. These results thus highlight that in the near future these RNN-based techniques will play an important role in the analysis of remote sensing time series.

Keywords: SAR; Sentinel-1; agricultural land cover map; recurrent neural network; long-short term memory; gated recurrent unit; K nearest neighbors; random forest; vector support machines; Camargue; France

1. Introduction

Spatial information about agricultural practices plays an important role for the sustainable development of agronomics, environment, and economics [1,2]. In fact, the importance of agricultural practices has long been recognized by the international community (e.g., Food and Agriculture Organization) [3]. Remote sensing satellite imagery is a valuable aid in providing and understanding this spatial distribution of agricultural practices. Particularly, recent years have seen the arrival of many

satellites to acquire high spatial resolution data on various spectral domains. The Sentinel-1 radar and Sentinel-2 optical sensors from the European Space Agency (ESA) are suited for monitoring agricultural areas. However, like all optical sensors, the use of Sentinel-2 data is limited if the cloud layer is large [4]. In contrast, Sentinel-1 is a Synthetic Aperture Radar (SAR) system that can acquire images in any type of weather with the advantage of providing images regardless of weather conditions. SAR data are well suited to distinguish rice from other types of vegetation cover [5]. The ESA Sentinel-1 SAR sensor (launched in 2014) (short revisit time: 12 days, and then 6 days after the launch of the second satellite in 2016, 20 m spatial resolution and two polarizations) allows a precise temporal follow-up of agricultural crop growth [6]. The ESA provides free data which makes it possible to envisage fine agricultural monitoring for various applications, in particular for providing detailed spatial agricultural land cover distribution.

In the Camargue region, farming is a major activity contributing to the productivity of the region. Among agricultural practices, rice cultivation is the most important one. It plays a crucial role in the development of cropping systems because the irrigation of rice allows the leaching of salt and, consequently, the introduction of other species into the rotation of crops [7]. In this region, to preserve the essential services for regulating agricultural systems for the environment, it is important to understand the operation of the farms [8]. In response to this demand, the spatial extent of agricultural land cover is essential.

In the literature of remote sensing for classification, the natural choice is based on supervised machine learning methods [9,10], which use training sets to classify pixels of unknown identity. Various supervised learning algorithms are available, each with its strengths and weaknesses [9,11,12]. The most recent methodological developments are focused on active learning and semisupervised learning approaches, which make use of unlabeled data for training [13–16]. Although the improvement in learning accuracy is considerable when unlabeled data are used in conjunction with a small amount of labeled data, the use of this approaches is still not common in agricultural land cover classifications. In practice, for agricultural applications, most of works in remote sensing are based on the standard algorithms, such as K nearest neighbor (KNN), random forest (RF), and support vector machine (SVM) [17,18]. These approaches, however, are not designed to work with time series data and, therefore, they ignore their temporal dependency.

Unlike the literature cited above, in this paper we assess the use of deep neural networks to consider the temporal correlation of the data. In fact, through recent advances in machine learning, there has been an increased interest in time series classification using deep convolutional neural networks (CNNs) and recurrent neuron networks (RNNs) that can take advantage of neural networks for end-to-end classification of a time series [19–21]. Moreover, RNN approaches can be used to work on pixel-based time series [19]. Accordingly, we focus our attention on RNN approaches for the classification. Thanks to their property, RNNs offer models to explicitly manage temporal dependencies among data (e.g., long short term memory ($LSTM$) [22] and Gated Recurrent Unit (GRU) [23]), which makes them suitable for the mining of multitemporal SAR Sentinel-1 data.

The objective of this work is to evaluate the potential of high spatial and temporal resolution Sentinel-1 remote sensing data to : (i) Map different agricultural land covers; and (ii) assess the new deep learning technique by comparing it with the standard machine learning approaches. For this, we propose to use two deep RNN approaches to explicitly consider the temporal correlation of Sentinel-1 data, which will be applied on the region of Camargue.

This paper is organized as follows: In Section 2, the Camargue study area is introduced; in Section 3, the processing SAR Sentinel-1 is reported; in Section 4, the classical machine learning approaches are briefly introduced; in section 5, the two deep RNNs models are presented; in Section 6, results are shown and a discussion is provided; and finally conclusions are drawn in Section 7.

2. Study Area

2.1. Camargue Site

The Camargue region, located in south-eastern France, is a coastal area of the Mediterranean Sea. It has a Mediterranean climate with mild winters, a long summer period (hot and dry), irregular rainfall and sunshine. Its climate has peculiarities related to its geographical location south of the Rhone corridor, between the Cevennes and the Southern Alps. The autumns are watered by brief but important precipitations and winters sometimes are rigorous due to mistral (<https://fr.wikipedia.org/wiki/Camargue#Climat>, last accessed March 2018). This zone covers the current perimeter of the Camargue Regional Natural Park, with an area of around 110,000 ha [24]. The study site is composed of five landscapes: Agriculture area, urban zone, water area, forests and natural environment. The agriculture zone can be defined by using the official Graphical Parcel Register (RPG) data, provided freely by the French government (<https://www.data.gouv.fr>). We used the recent version of RPG 2015 for our study in order to delimit the agricultural areas. In our study area, this corresponds to 54,082 ha (see the cyan polygon in Figure 1), in which the permanent moors, orchards, and olive trees areas are 17,859 ha, and the remaining 36,223 ha is used for common agricultural activities. In this 36,223 ha agricultural zone, rice is the most dominant crop (with 44% and 16,000 ha in 2011 [8]) and has an important function in the economic, ecological and social equilibrium of the region.

2.2. Ground Data

A field survey was carried out on 921 reference plots in July 2017 to collect the land cover information. The boundary of reference plots was drawn manually, based on the high resolution 60 cm World Imagery layer of ArcGIS data online (accessed in July 2017). We chose 11 observed surface classes: (1) Rice; (2) sunflower; (3) lawn; (4) irrigated grassland; (5) durum wheat (winter); (6) alfalfa; (7) tomato; (8) melon; (9) clover; (10) swamps; and (11) vineyard. Figure 1 shows the position of ground samples and the distribution of the pixel number per class and number of plots is shown on Table 1.

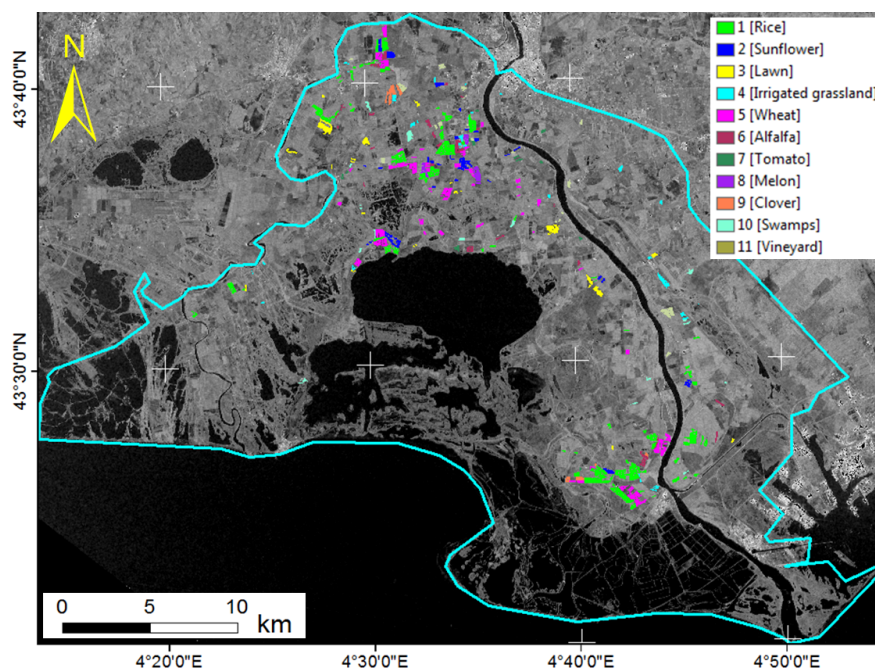


Figure 1. Camargue study area. Colored polygons represent 921 reference plots location. The study area is limited by the cyan polygon.

Table 1. The distribution of the number of pixel and plots per class.

ID	Vegetation Class	Number of Plots	Number of Pixels	Surface Area (ha)	Area (%)
(1)	Rice	284	23,275	931	35.2
(2)	Sunflower	77	5541	222	8.4
(3)	Lawn	27	5018	201	7.6
(4)	Irrigated grassland	53	3528	141	5.3
(5)	Wheat	304	13,057	522	19.7
(6)	Alfalfa	59	4580	183	6.9
(7)	Tomato	15	1362	55	2.1
(8)	Melon	24	1978	79	2.9
(9)	Clover	26	1918	77	2.9
(10)	Swamps	19	2535	101	3.8
(11)	Vineyard	33	3409	136	5.2
Total		921	66,201	2648	100

3. SAR Data and Processing

3.1. SAR Data

Since wheat cultivation is the only winter crop presence after May and the major agricultural practices in Camargue are in summer (e.g., from May to September), we focus our data analysis on this period. The Sentinel-1A/1B SAR dataset includes 25 acquisitions in terrain observation with progressive scan (TOPS) mode from May to September 2017 (5 months), with a revisit period of 6 days. This is dual-polarization (VV + VH) data, resulting in 50 images. Figure 2 summarizes the temporal profiles of the 11 agricultural classes per polarization. Each time series is made up of 25 points (one for each acquisition). Figure 3 provides information on the temporal dynamic of these classes by giving their average and standard deviation.

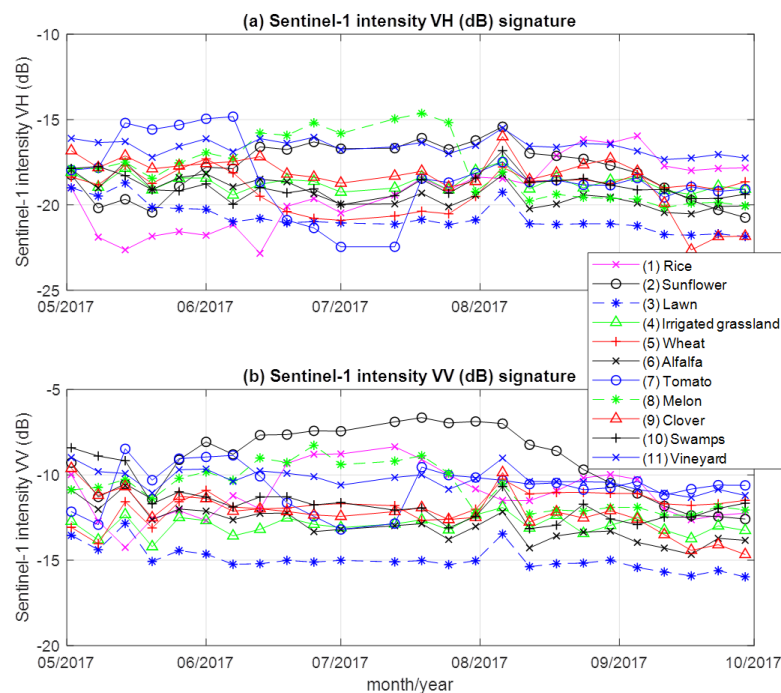


Figure 2. The temporal profiles of the eleven different classes which respect to the VH (a) and VV (b) polarizations.

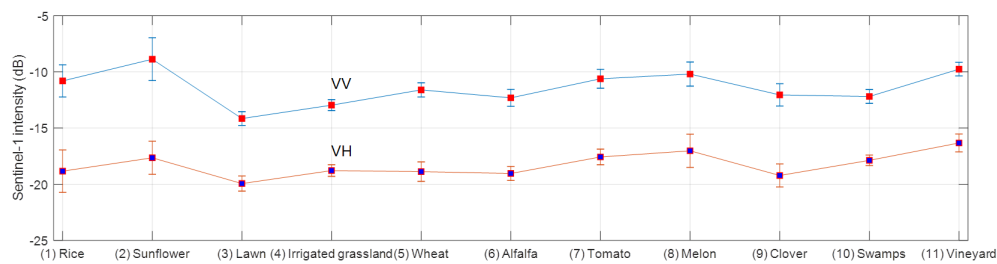


Figure 3. The average and standard deviation of the eleven different classes for VV and VH polarizations.

3.2. Pre-Processing Data

First, a master image was chosen and all images are coregistered, taking into account TOPS mode, to the master image [25]. Five-look (5 range looks) intensity images are generated and radiometrically calibrated for range spreading loss, antenna gain, normalized reference area and the calibration constant that depends on the parameters in the Sentinel-1 SAR header.

3.3. Temporal Filtering

Reliable estimates of the intensity from a distributed target require that the estimated number of looks (ENL) is sufficiently large. Speckle filtering is often used to increase the ENL with loss of spatial resolution [26]. In properly coregistered multitemporal datasets, it is possible to employ the technique of temporal filtering, which, in principle, increases radiometric resolution without degrading spatial resolution. The temporally filtered images usually show markedly diminished speckle, with little or no reduction in spatial resolution. In this paper, we improve the time series SAR Sentinel-1 dataset by exploiting a temporal filtering method developed by [26] to reduce noise, while retaining, as much as possible, the fine structures present in the images.

3.4. Geocoding

After pre-processing and filtering, all the processed images are in the imaging geometries of the master image. In order to create a unified dataset, all image data have to be orthorectified into map coordinates. This is done by creating a simulated SAR image from a SRTM DEM 30 m, and using the simulated SAR image to coregister the two image sets (polarizations). The pixel size of the orthorectified image data is 20 m. After geocoding, all intensity images are transformed to the logarithmic dB scale, normalized to values between 0–255 (8 bits) and input into classifiers. The SAR Sentinel-1 data are processed by the TomoSAR platform, which offers SAR, interferometry and tomography processing [27,28]. Finally, for each pixel, a 25×2 matrix (25 acquisitions and two polarizations: VV + VH) is generated as input for classifiers.

4. Classical Machine Learning Approaches

Among supervised machine learning approaches, we use K nearest neighbor (KNN), random forest (RF), and support vector machine (SVM) approaches as a baseline for comparing the performance of Sentinel-1 image data for land cover classification with deep recurrent neural network approaches in Section 6. The rationale for this choice is mainly because they are the most popular in remote sensing [17,18,29] and remain competitive w.r.t. other approaches in many scenarios. In this section, we provide a brief introduction of these methods for the sake of complete.

4.1. K Nearest Neighbors

KNN is classified as a non-parametric machine learning method, because it simply remembers all of its training data. Despite its simplicity, KNN has been successful in a large number of classification

problems, including remote sensing satellite images. The principle of *KNN* is that in the set of training data, it finds a group of K samples that are closest to unknown samples (based on a certain distance function, e.g., Euclidean) [30]. In *KNN* approach, the class of an unknown sample is determined by applying a majority vote on the classes of its K nearest neighbors [30,31].

4.2. Random Forest

The Random Forest algorithm has demonstrated its ability to yield high quality mappings for a different varieties of crop type systems with a much faster computation when compared to other state of the art classifiers [32–34]. The classifier relies on aggregating the results of an ensemble of simpler decision tree classifiers. In other words, it is a meta-estimator that fits a number of decision tree classifiers on various subsamples of the dataset, and uses averaging to improve the predictive accuracy and control over-fitting [35]. To reduce the computational complexity of the algorithm and the correlation between subsamples, tree construction can be stopped when a maximum depth is reached or when the number of samples on the node is less than a minimum sample threshold.

4.3. Support Vector Machine

The Support Vector Machine algorithm has been proved to be superior to most other image classification algorithms in terms of classification accuracy [36]. *SVM* is basically a binary classifier that delineates two classes by fitting an optimal separating hyperplane to the training data in the multidimensional feature space to maximize the margin between them [37]. *SVM* uses a kernel function to project the data from input space into feature space. Several kernels can be used: Linear, radial basis function (RBF), and polynomial or sinusoidal kernels. Linear and RBF kernels are the most commonly used. While linear kernels are computationally more efficient, nonlinear kernels, such as RBF kernels, tend to outperform linear kernels. For RBF kernels, two parameters should be specified: (i) The complexity parameter C which controls the trade-off between the maximization of the margin between the training data vectors and the training error decision limit; and (ii) the gamma parameter that is the width of the kernel function. Increasing C values usually require increasing computational time and high C values also increase risks of over-fitting.

5. Recurrent Neural Network

Recurrent neural networks are well-designed machine learning techniques that stand out for their quality in different fields of activity, such as signal processing, natural language processing and speech recognition [38,39]. Contrary to convolutional neural networks, RNNs clearly manage the temporal data dependencies, since the output of the neuron in time $t - 1$ is used with the next input, to feed the neuron itself at time t . A diagram of a typical neural RNN is detailed in Figure 4. Among different RNN models, we have Long-Short Term Memory (*LSTM*) [22] and Gated Recurrent Unit (*GRU*) [23], which are the two most well known RNN units. The main difference between them is related to the number of parameters to learn. Considering the same size of the hidden state, the *LSTM* model has more parameters than the *GRU* unit.

In the following, we will briefly analyze the two RNN units (*LSTM* and *GRU*). For each of them, we aim to provide and discuss the equations that describe its internal behavior. The \odot symbol represents an element-wise multiplication, while σ and \tanh are Sigmoid and Hyperbolic Tangent function, respectively. The input of a RNN unit is a sequence of variables (x_1, \dots, x_t) , where each element x_t is a vector and t refers to the corresponding timestamp.

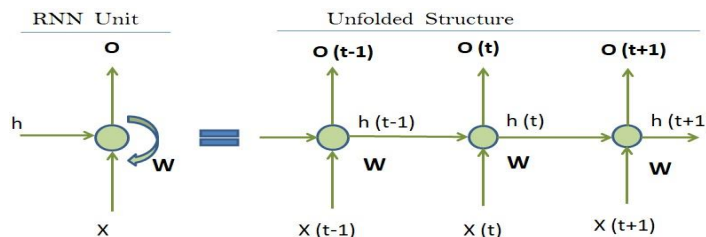


Figure 4. RNN Unit (on the left) and unfolded structure (on the right).

5.1. Long-Short Term Memory (LSTM)

The existing RNN models fail to learn long-term dependencies because of the problem of vanishing and exploding gradients. To overcome this challenge, the LSTM model is used [22]. The Equations (1)–(6) formally describes the LSTM neuron. The LSTM set consists of two cell states: The C_t memory and the h_t hidden state. Three different gates intervene in the control of the flow of information: The input (i_t), the forget (f_t) and the output (o_t). All three gates mix the current entry, x_t , with the hidden state, h_{t-1} , from the previous timestamp. Also, the gates have two major functions: (i) They regulate the quantity of information to forget/remember during the process; (ii) they deal with the problem of gradient disappearance/bursting. We can see that the gates are implemented by a sigmoid. This function gives values between 0 and 1. The LSTM unit also uses a temporary cell state, y_t , that resizes the current input. This current cell is applied by a hyperbolic tangent function that gives values between -1 and 1 . The sigmoid and the hyperbolic tangent work per element. i_t sets the amount of information to keep ($i_t \odot y_t$), while f_t indicates how much memory should be kept in the current step ($f_t \odot c_{t-1}$). The input of an RNN is a sequence of variables (x_1, \dots, x_n) , where x_t is a generic element that represents a feature vector and t refers to the corresponding timestamp.

Finally, o_t has an impact on the new hidden state, h_t , which determines how much information from the current memory will be on the output step. The different matrices, W_{**} , and bias coefficients, b_* , are the parameters used during model formation. The memory, C_t , and the hidden state, h_t , are both transmitted at the next step.

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \tag{1}$$

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \tag{2}$$

$$y_t = \tanh(W_{yx}x_t + W_{yh}h_{t-1} + b_y) \tag{3}$$

$$c_t = i_t \odot y_t + f_t \odot c_{t-1} \tag{4}$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \tag{5}$$

$$h_t = o_t \odot \tanh(c_t) \tag{6}$$

5.2. Gated Recurrent Unit (GRU)

To facilitate the computation and implementation of the LSTM model, the authors [23] develop a new RNN unit. The neuron GRU is illustrated in Equations (7)–(9). This new unit operates as the LSTM model by performing the gates and cell states but, conversely to the LSTM pattern, the GRU unit has two gates: Update (z_t) and reset (r_t), and also we have a cell state, the hidden state, (h_t). In addition, the two gates merge the current input (x_t) with information from previous timestamps (h_{t-1}). The update gate efficiently regulates the compromise between the amount of information from the previously hidden state (which will be included in the current hidden state) and the amount of information of the current timestamp that is to be kept. This acts in the same way as the unit memory cell LSTM that supports the RNN to remember information in the future.

Furthermore, the reset gate analyzes the amount of information from previous timestamps that could be embedded in the current information. Since each hidden unit has reset and updates gates, they register dependencies at different levels. Units that are more likely to capture short-term dependencies may have a frequently-enabled reset gate [23].

$$z_t = \sigma(W_{zx}x_t + W_{zh}h_{t-1} + b_z) \quad (7)$$

$$r_t = \sigma(W_{rx}x_t + W_{rh}h_{t-1} + b_r) \quad (8)$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tanh(W_{hx}x_t + W_{hr}(r_t \odot h_{t-1}) + b_h) \quad (9)$$

5.3. RNN-Based Time Series Classification

To complete the classification task, at each RNN unit we make a deep architecture, putting together five units. The use of several units, similar to what is frequently done for CNN models in several convolutional layers [40], will extract high-level non-linear time dependencies that are in the remote sensing time series. This is done for both *LSTM* and *GRU*.

The RNN model follows a new sequence at the input, but it makes no prediction by itself. For this purpose, A SoftMax [41] layer is stacked on the last recurrent unit to predict the final multi-class. The SoftMax layer has the same number of neurons as classes to predict. For layer normalization reasons, the SoftMax priority is given instead of the Sigmoid function, because the value of the SoftMax layer can be considered as a probability distribution on classes that total up to 1, whereas in the case of the sigmoid, the neurons give values between 0 and 1. Each sample belongs absolutely to a single class, which leads us to the choice of SoftMax. This schema is instantiated for both *LSTM* and *GRU* units, thus coming up with two different classifiers: An *LSTM*-based and a *GRU*-based classification scheme. Figure 5 shows a schematic view of the *LSTM*-based architecture for each pixel in our paper (e.g., 25 points input VV/VH, 5 *LSTM* units, 512 hidden dimensions and 11 classes output).

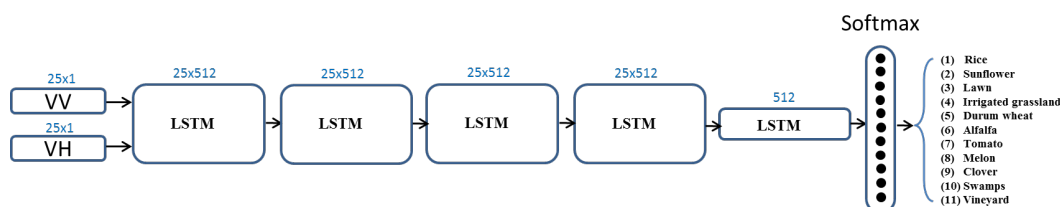


Figure 5. The schematic view of the RNN *LSTM*-based architecture. By replacing *LSTM* to *GRU* unit, we get the RNN *GRU*-based architecture.

6. Experimental Results

6.1. Experimental Settings

We compared RNN-based classification approaches (*LSTM* and *GRU*) with standard machine learning approaches.

For the *RF* model, we set the number of trees at 400 and a maximum tree depth of 25. For the *KNN* model, we set the number of nearest neighbors at 10. For the *RF* model, the number of randomly selected features at each node is kept at its default value for a classification problem (i.e., the square root of the total number of variables [42]). For the *SVM* model, we use the RBF kernel with a default gamma (i.e., $1/\text{number of samples}$ [43]) and a parameter of complexity equal to 10^5 . For *RF* and *KNN*, we use the python implementation provided by the Scikit-learn library [42], while for *SVM* we use the LibSVM implementation [43].

For RNN-based classifiers, we define the number of hidden dimensions equal to 512 (General speaking, the choice of the number of hidden dimensions depends on the volume of the training dataset available. If the dataset is small, we have to use a simple model in order to estimate fewer weight

parameters and vice versa. For example, in the work [20], the number of units is set to 64 and 512 for *Thau* and *Reunion* test sites, respectively. We experimentally found that our best performance was achieved by setting 512, after trying with 64, 128, 254, 512 and 1024). An initial learning rate of 5×10^{-4} , ρ of 0.9 and a decline of 5×10^{-5} was employed. We implement the model via the *Keras* python library with *Theano* as the back end [44]. To train the model, we used the *Rmsprop* strategy, which is a variant of stochastic gradient descent [45]. The loss function being optimized is categorical cross entropy, which is the standard loss function used in all multiclass classification jobs [46]. The model is trained for 250 epochs, with a batch size set at 64. For the different methods, we apply cross-validation 5 times on the dataset, with a split procedure in two steps: (1) At the polygon-level to separate train and test instances and (2) at the pixel-level to extract each instance [47,48]. In other words, we impose that pixels of the same object belong exclusively to the training or to the test set. In this way, the initial set of data (66,201 pixels) is reduced to 57,585 pixels, since our polygons are not the same size. In detail, the set of data (57,585 pixels) is randomly divided into 5 partitions (folds) of equal size. Then, four folds are used to train the models (about 46,068 pixels), and the fifth fold (about 11,517 pixels) is used for the test phase. The operation is repeated five times, so that each fold is like a possible test set. Finally, the score measurements are calculated by concatenating the predicted labels per fold with respect to the real classes. Experiments were performed on a workstation Intel(R) Xeon(R) CPU E5-2667 v4@3.20Ghz with 256 GB of RAM and GPU TITAN X. The training step takes, on average, 166 min to learn each RNN model on the training set and less than 3 min to classify the pixel time series in the test set.

In order to assess classification performances, we use not only the global accuracy and kappa measures, but also average and per-class F-measures.

6.2. Results

The multitemporal Sentinel-1 data, processed as in Section 3, are used as an input for classification using classical approaches (*KNN*, *RF*, and *SVM*) in Section 4 and two RNN-based models (*LSTM* and *GRU*) in Section 5. The summary of the results of different classification approaches is reported in Table 2. This is the performance from cross-validation 5 times on Sentinel SAR-1 time series data, showing the average and standard deviation values of the F-measure, Accuracy, and Kappa assessment metrics from 5 repetitions. For this multi-temporal SAR Sentinel-1, all classifier performance metrics are very high, showing the quality of the dataset for agricultural classification tasks.

Table 2. The average and standard deviation from cross-validation 5 times on the time series SAR Sentinel-1 data.

Classifier	F-Measure	Accuracy	Kappa
<i>KNN</i>	86.1 ± 0.6%	85.6 ± 0.6%	0.823 ± 0.009
<i>RF</i>	87.1 ± 0.9%	86.9 ± 1.2%	0.833 ± 0.015
<i>SVM</i>	87.3 ± 1.5%	87.1 ± 1.6%	0.837 ± 0.019
<i>LSTM</i>	89.2 ± 1.7%	89.1 ± 1.6%	0.862 ± 0.020
<i>GRU</i>	89.8 ± 1.6%	89.6 ± 1.6%	0.869 ± 0.019

Second, to illustrate more precise comprehension of the behavior of the different classifiers, we report a per-class F-measure comparison in Figure 6. In this figure, the eleven classes are evaluated according to the different methods (*KNN*, *RF*, *SVM*, *LSTM* and *GRU*) used in the classification. In addition, confusion matrices are also reported in Figure 7. In both figures, results show a better performance using RNN-based classification approaches over classical machine learning methods (*KNN*, *RF* and *SVM*). Between the two RNN models, the *GRU*-based method obtains slightly better results than the *LSTM*. This is as expected, due to the fact that the *GRU* unit is considered as an improvement of the *LSTM* unit [23].

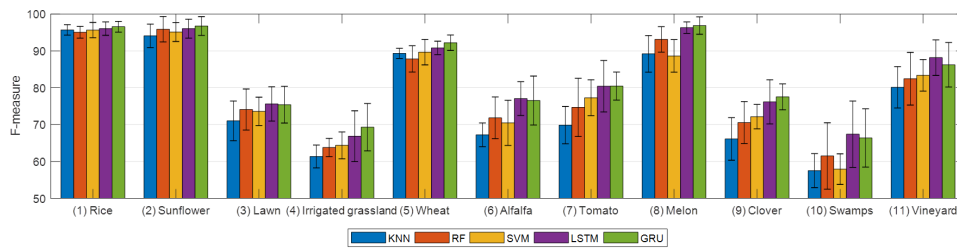


Figure 6. Per Class F-Measure of the different approaches.

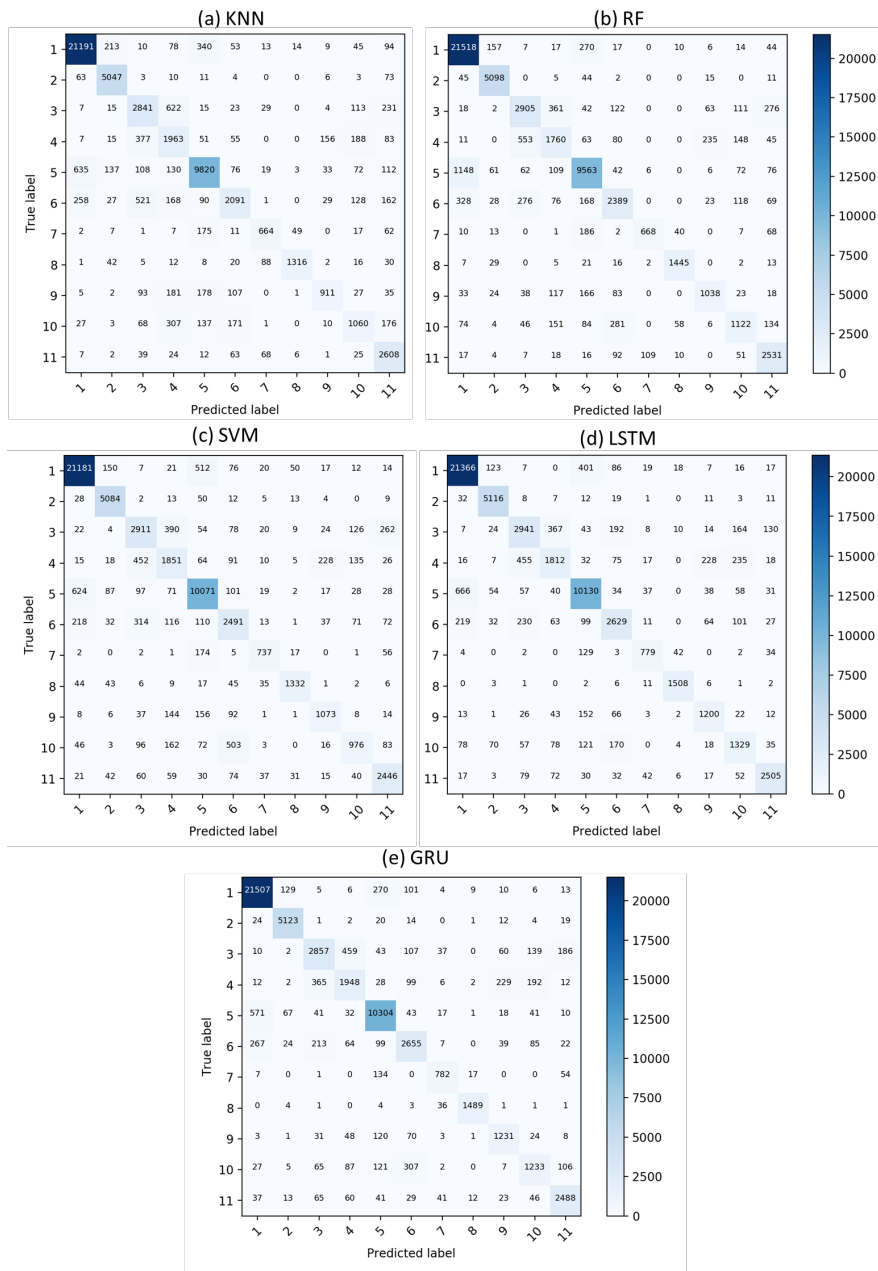


Figure 7. Confusion matrices on the SAR Sentinel-1 time series data of the different approaches: (a) KNN, (b) RF; (c) SVM; (d) LSTM and (e) GRU. The name of the labels: (1) rice; (2) sunflower; (3) lawn; (4) irrigated grassland; (5) durum wheat (winter); (6) alfalfa; (7) tomato; (8) melon; (9) clover; (10) swamps; and (11) vineyard.

Finally, by applying the best RNN-based GRU classifier for the whole area study, we established the agricultural land cover map for Camargue in 2017 (see Figure 8). Figure 9 is a zoomed version of the white-border box in the Figure 8, to facilitate visualization of classification results for the RNN-based GRU and the SVM approach w.r.t the reference plots. The classification results for both approaches were matched with the reference plots, and were slightly smoother in the RNN-based GRU result. Among different agricultural classes, rice is the dominant practice (with 29.3% and 10,627 ha, see Table 3) by its extent and presence in almost all areas of the region.

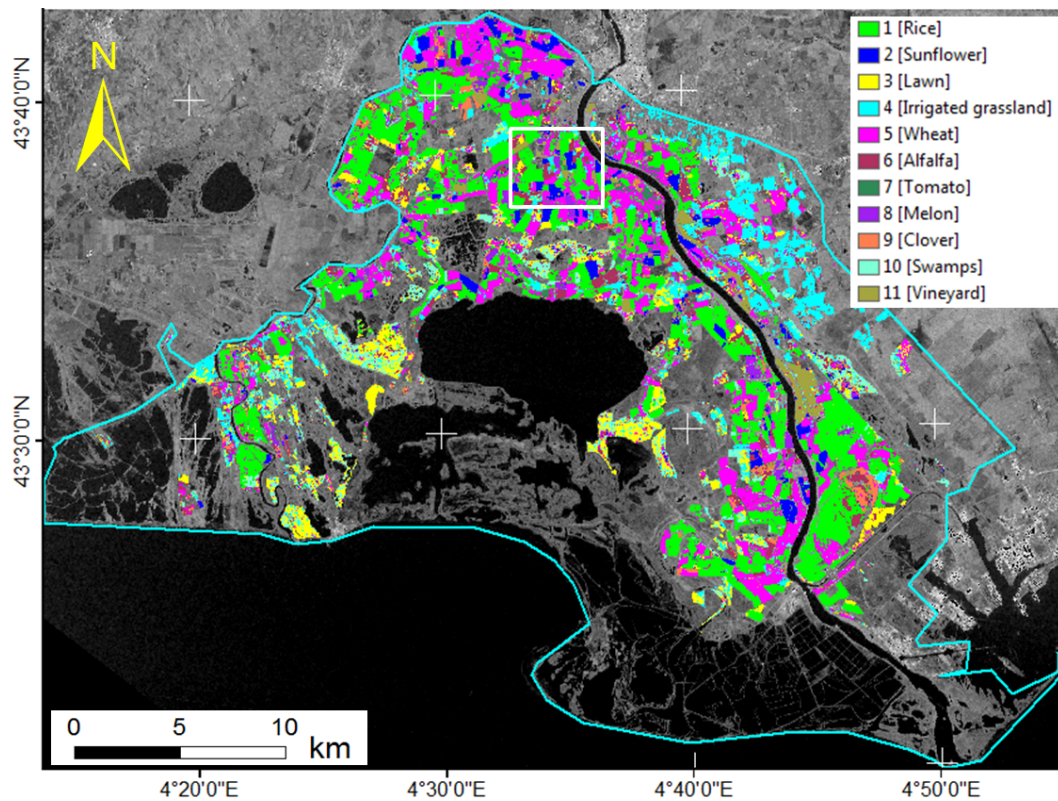


Figure 8. The agricultural land cover map in Camargue using the RNN-based GRU multitemporal SAR Sentinel-1.

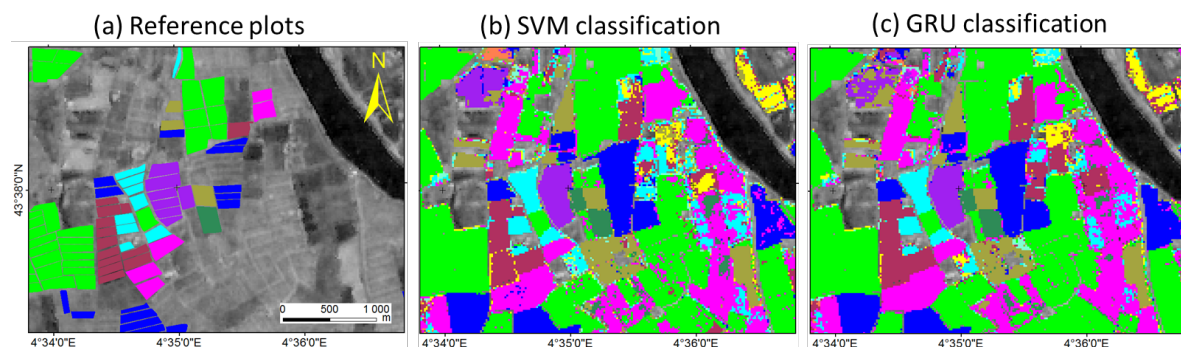


Figure 9. A zoom version of the white-border box in the Figure 8 is provided to facilitate visualization of classification results. (a) reference plots; (b) the classical SVM result and (c) the RNN-based GRU result.

Table 3. The distribution of the agricultural land cover class in 2017.

ID	Class	Area (ha)	Percent
(1)	Rice	10,627	29.3
(2)	Sunflower	1676	4.6
(3)	Lawn	3357	9.4
(4)	Irrigated grassland	4148	11.5
(5)	Wheat	7439	20.5
(6)	Alfalfa	2593	7.2
(7)	Tomato	592	1.6
(8)	Melon	622	1.7
(9)	Clover	884	2.4
(10)	Swamps	2591	7.2
(11)	Vineyard	1694	4.6
Total		36,223	100

6.3. Discussion

In this work we show that the multitemporal Sentinel-1 data can be used to classify different agriculture classes in Camargue, France. We obtained good results, using both classical approaches and the advance deep RNN techniques. The metric of validation indicates good performance, in which F-measure/Accuracy was greater than 86% and Kappa coefficient was better than 0.82. Together, these results confirm the suitability of Sentinel-1 time series data for agriculture land cover applications.

First, we show that even with the classical approaches, good classification performance could be achieved with radar time series data. This was expected to be challenging, because radar images are characterized by considerable speckle noise, which does not exist in optical images. We note that the same performance can not be straightforwardly transferred to the case of having few radar data. Good performance can be achieved by the fact that Sentinel-1 SAR with 6 days revisit time allows not only a precise temporal follow-up of the agriculture crop growth, but also mostly noise-free data, thanks to the multitemporal speckle filtering. It is worth pointing out that, nowadays, the Sentinel-1 constellation is the only satellite system providing dense time series with global coverage. It is therefore a good candidate for operational agriculture land cover mapping.

We can observe that RNN-based classification approaches have better performance over classical machine learning methods (*KNN*, *RF* and *SVM*). Between the two deep RNN models, the *GRU*-based method has slightly better results than the *LSTM* one. To give a more precise comprehension of the behavior of the different methods, from Figure 6, we can see that the performance gain offered by the RNN-based methods involve all the eleven classes, resulting in equally good result on all the them. Conversely, *KNN*, *RF* and *SVM* show different behaviors for different classes. Both classifiers obtain the best performances on the *Rice* class (1) and the lowest performances on the *Irrigated grassland* class (4). This behavior can be explained by considering the temporal profiles of both VV and VH presented in Figure 2. The *Rice* class (depicted with pink lines in Figure 2) has a clear and distinct profile, with a strong dynamic (with large standard deviation value, see Figure 3), facilitating its detection. The *Irrigated grassland* class (depicted with green diamond lines in Figure 2) has a weak temporal behavior, with a small standard deviation value (see Figure 3), and intersects the temporal profiles of all the other classes multiple times. This is probably the reason why the standard approaches are not able to correctly detect this class, since they ignore the temporal correlation of the data. On the other hand, the RNN-based approaches discriminate well among all the classes, since they can extract and summarize the important signal portions that help the discriminative task among the different agriculture classes.

Based on Figure 7, which shows the confusion matrices for each method, we can see that a high misclassification rate is recorded between the *Irrigated grassland* (4) and *Swamps* (10) classes. This is true for all the different classifiers. However, for the RNN-based approaches, this misclassification error is not as high. Conversely, in the case of *KNN*, *RF* and *SVM*, this misclassification behavior

is significant. In fact, the standard machine learning approaches are under strain, and they are responsible for a considerable misclassification rate. In addition, the standard machine learning approaches are not always good at dealing with imbalanced classification problems [49,50]. Thanks to the joint optimization of nonlinear input transformations along with classification, deep learning approaches provide here a valuable strategy to discriminate among the different agriculture classes. In addition, as expected, the ability of RNNs to deal with the temporal correlations characterizing the SAR Sentinel-1 data result in a gain in performance on all classes, with particular emphasis on those classes that exhibit similar temporal behaviors for longer periods. Together, these results confirm that the RNN models (both *LSTM*-based and *GRU*-based) are well suited to detect and exploit temporal dependencies, as opposed to common classification approaches that do not explicitly leverage temporal correlations. Our finding is consistent with previous reports, in which Deep Learning outperformed the classical machine learning approaches [19–21].

In this paper, we selected *KNN*, *RF* and *SVM* due to their most popular supervised classification algorithms in the remote sensing community. Although they were introduced way back in the early 2000s, they still remain competitive w.r.t. other approaches in many scenarios, and they represent the algorithms to which any new method needs to be compared to. Considering the setting of remote sensing time series analysis (both optical and SAR), to date, no other classifier reaches the same generality and classification performance of these approaches, and this is why we decided to compare our RNNs proposals to them. Unfortunately, to the best of our knowledge, we are not aware of any other (more recent) classification methods that are commonly employed to perform supervised classification on time series SAR data. To summarize our position, this comparison allows us to appreciate more keenly how important it is to have techniques that can be employed to intelligently exploit temporal dependency among data w.r.t. standard machine learning approaches employed in the remote sensing field. Due to the promising results we have obtained with RNNs, we think that, in the near future, these techniques will play an important role in the analysis of remote sensing time series.

For agricultural land cover in Camargue 2017, we can observe that the region areas are occupied mostly by rice (29.3% and 10,627 ha) and wheat (20.5% and 7439 ha) distributions. We note that the performance of both RNN-based *LSTM* and *GRU* classifiers is best on the *Rice* class, with the same F-measure metric of 96%. As a consequence, an RNN-based classifier is a valuable tool to discriminate the rice from other agriculture classes. In Camargue, there is a great variability in the types of rice farms [8], and the rice areas vary according to the year. The rice areas cultivated have significantly decreased from 16,000 ha in 2011 (e.g., in [8]) to 10,627 ha (our estimation) in 2017. The decreased phenomena of the rice extent could lead to a negative effect to the sustainable development of Camargue. Future works on this region could be focused on considering the assimilation of new source remote sensing data, such as Sentinel-1 radar, in crop models to estimate rice production, to follow farming practices, and to be able to propose strategies for sustainable agricultural development.

7. Conclusions

In this paper, we studied the potential of high spatial and temporal resolution Sentinel-1 remote sensing data for different agriculture land cover mapping applications and assessed the new deep learning techniques. We proposed to use two deep RNN approaches to explicitly consider the temporal correlation of Sentinel-1 data, which were applied on the Camargue region.

We demonstrated that even with the classical approaches (*KNN*, *RF* and *SVM*), good classification performance could be achieved with Sentinel-1 SAR image time series. We experimentally demonstrated that the use of recurrent neural networks to deal with SAR Sentinel-1 time series data yields a consistent improvement in agricultural classes as compared with classical machine learning approaches. The experiments highlight the appropriateness of a specific class of deep learning models (RNNs) which explicitly consider the temporal correlation of the data in order to discriminate among agricultural classes of land cover, typically characterized by similar but complex temporal behaviors.

We observed for the *Rice* class the same high performance of both RNN-based *LSTM* and *GRU* classifiers, with a F-measure of 96%. We found that in the Camargue region, rice land cover is still dominant, but decreasing significantly. Future work could be focused on estimating rice production yields in order to propose strategies for sustainable agriculture development.

Finally, although the number of exploitable optical images are limited due to cloud cover, the use of multi-source data including optical and SAR radar data will be an important subject of future research.

Author Contributions: E.N. and D.H.T.M. developed the main idea that led to this paper. E.N. and D.H.T.M. provided SAR processing, classifications and their descriptions. E.N. collected the reference data and was the main author of the paper. INRA has provide the main contacts with farmers on the Camargue area and the main shapefiles including ground observations on crop for various farmers. All authors read and approved the final manuscript.

Funding: This research received no external funding.

Acknowledgments: This work was supported in part by European Space Agency, Centre National d'Etudes Spatiales/Terre, Ocean, Surfaces Continentales, Atmosphere and in part by the National Research Agency within the framework of the program "Investissements d'Avenir" for the GEOSUD under Project ANR-10-EQPX-20. We thank the government of Burundi for funding the PhD research of Emile Ndikumana. We thank Jarlath SLEVIN for English correction. The authors wish to thank Dino Ienco (UMR TETIS—IRSTEA) and Raffaele Gaetano (UMR TETIS—CIRAD) for insight discussion on the deep learning.

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

References

- Buckley, C.; Carney, P. The potential to reduce the risk of diffuse pollution from agriculture while improving economic performance at farm level. *Environ. Sci. Policy* **2013**, *25*, 118–126. [[CrossRef](#)]
- Foley, J.A.; DeFries, R.; Asner, G.P.; Barford, C.; Bonan, G.; Carpenter, S.R.; Chapin, F.S.; Coe, M.T.; Daily, G.C.; Gibbs, H.K.; et al. Global consequences of land use. *Science* **2005**, *309*, 570–574. [[CrossRef](#)] [[PubMed](#)]
- Polso, A.; Speedy, A.; Kueneman, E. Good Agricultural Practices—A Working Concept. In Proceedings of the FAO Internal Workshop on Good Agricultural Practices, Rome, Italy, 27–29 October 2004; Volume 1, p. 41.
- Drusch, M.; Bello, U.D.; Carlier, S.; Colin, O.; Fernandez, V.; Gascon, F.; Hoersch, B.; Isola, C.; Laberinti, P.; Martimort, P.; et al. Sentinel-2 ESA Optical High-Resolution Mission for GMES Operational Services. *Remote Sens. Environ.* **2012**, *120*, 25–36. [[CrossRef](#)]
- Le Toan, T.; Ribbes, F.; Wang, L.F.; Floury, N.; Ding, K.H.; Kong, J.A.; Fujita, M.; Kurosu, T. Rice crop mapping and monitoring using ERS-1 data based on experiment and modeling results. *IEEE Trans. Geosci. Remote Sens.* **1997**, *35*, 41–56. [[CrossRef](#)]
- Torres, R.; Snoeij, P.; Geudtner, D.; Bibby, D.; Davidson, M.; Attema, E.; Potin, P.; Rommen, B.; Floury, N.; Brown, M.; et al. GMES Sentinel-1 mission. *Remote Sens. Environ.* **2012**, *120*, 9–24. [[CrossRef](#)]
- Mouret, J.C. Etude de l'Agrosystème Rizicole en Camargue dans ses Relations avec le Milieu et le Systeme Cultural: Aspects Particuliers de la Fertilité. Ph.D. Thesis, Université des Sciences et Techniques du Languedoc, Montpellier, France, 1988.
- Delmotte, S.; Tittonell, P.; Mouret, J.C.; Hammond, R.; Lopez-Ridaura, S. On farm assessment of rice yield variability and productivity gaps between organic and conventional cropping systems under Mediterranean climate. *Eur. J. Agron.* **2011**, *35*, 223–236. [[CrossRef](#)]
- Friedl, M.A.; Brodley, C.E. Decision tree classification of land cover from remotely sensed data. *Remote Sens. Environ.* **1997**, *61*, 399–409. [[CrossRef](#)]
- Li, C.; Wang, J.; Wang, L.; Hu, L.; Gong, P. Comparison of classification algorithms and training sample sizes in urban land classification with Landsat thematic mapper imagery. *Remote Sens.* **2014**, *6*, 964–983. [[CrossRef](#)]
- Waske, B.; Braun, M. Classifier ensembles for land cover mapping using multitemporal SAR imagery. *ISPRS J. Photogramm. Remote Sens.* **2009**, *64*, 450–457. [[CrossRef](#)]
- Lu, D.; Weng, Q. A survey of image classification methods and techniques for improving classification performance. *Int. J. Remote Sens.* **2007**, *28*, 823–870. [[CrossRef](#)]
- Li, J.; Bioucas-Dias, J.M.; Plaza, A. Semisupervised Hyperspectral Image Segmentation Using Multinomial Logistic Regression With Active Learning. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 4085–4098. [[CrossRef](#)]

14. Gomez-Chova, L.; Camps-Valls, G.; Munoz-Mari, J.; Calpe, J. Semisupervised Image Classification with Laplacian Support Vector Machines. *IEEE Geosci. Remote Sens. Lett.* **2008**, *5*, 336–340. [[CrossRef](#)]
15. Munoz-Mari, J.; Bovolo, F.; Gomez-Chova, L.; Bruzzone, L.; Camp-Valls, G. Semisupervised One-Class Support Vector Machines for Classification of Remote Sensing Data. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 3188–3197. [[CrossRef](#)]
16. Tuia, D.; Volpi, M.; Copa, L.; Kanevski, M.; Munoz-Mari, J. A survey of active learning algorithms for supervised remote sensing image classification. *IEEE J. Sel. Top. Signal Process.* **2011**, *5*, 606–617. [[CrossRef](#)]
17. Inglada, J.; Vincent, A.; Arias, M.; Marais-Sicre, C. Improved Early Crop Type Identification by Joint Use of High Temporal Resolution SAR and Optical Image Time Series. *Remote Sens.* **2016**, *8*, 362. [[CrossRef](#)]
18. Flamary, R.; Fauvel, M.; Mura, M.D.; Valero, S. Analysis of Multitemporal Classification Techniques for Forecasting Image Time Series. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 953–957. [[CrossRef](#)]
19. Ho Tong Minh, D.; Ienco, D.; Gaetano, R.; Lalande, N.; Ndikumana, E.; Osman, F.; Maurel, P. Deep Recurrent Neural Networks for Winter Vegetation Quality Mapping via Multitemporal SAR Sentinel-1. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 464–468. [[CrossRef](#)]
20. Ienco, D.; Gaetano, R.; Dupaquier, C.; Maurel, P. Land Cover Classification via Multitemporal Spatial Data by Deep Recurrent Neural Networks. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1685–1689. [[CrossRef](#)]
21. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [[CrossRef](#)]
22. Hochreiter, S.; Schmidhuber, J. LSTM can Solve Hard Long Time Lag Problems. In Proceedings of the NIPS, Denver, CO, USA, 2–5 December 1996; pp. 473–479.
23. Cho, K.; van Merriënboer, B.; Gülçehre, Ç.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In Proceedings of the EMNLP, Doha, Qatar, 25–29 October 2014; pp. 1724–1734.
24. Manfron, G.; Delmotte, S.; Busetto, L.; Hossard, L.; Ranghetti, L.; Brivio, P.A.; Boschetti, M. Estimating inter-annual variability in winter wheat sowing dates from satellite time series in Camargue, France. *Int. J. Appl. Earth Observ. Geoinform.* **2017**, *57*, 190–201. [[CrossRef](#)]
25. Prats-Iraola, P.; Scheiber, R.; Marotti, L.; Wollstadt, S.; Reigber, A. TOPS Interferometry With TerraSAR-X. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 3179–3188. [[CrossRef](#)]
26. Quegan, S.; Jiong Jiong, Y. Filtering of multichannel SAR images. *IEEE Trans. Geosci. Remote Sens.* **2001**, *39*, 2373–2379. [[CrossRef](#)]
27. Ho Tong Minh, D.; Ngo, Y.-N.; Baghdadi, N.; Maurel, P. TomoSAR platform: A new Irstea service as demand for SAR, Interferometry, Polarimetry and Tomography. In Proceedings of the 2016 ESA Living Planet Symposium, Prague, Czech Republic, 9–13 May 2016; pp. 1–8.
28. Ho Tong Minh, D.; Ngo, Y.-N. Tomosar platform supports for Sentinel-1 tops persistent scatterers interferometry. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 1680–1683.
29. Khatami, R.; Mountrakis, G.; Stehman, S.V. A meta-analysis of remote sensing research on supervised pixel-based land-cover image classification processes: General guidelines for practitioners and future research. *Remote Sens. Environ.* **2016**, *177*, 89–100. [[CrossRef](#)]
30. Weinberger, K.Q.; Saul, L.K. Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.* **2009**, *10*, 207–244.
31. Qian, Y.; Zhou, W.; Yan, J.; Li, W.; Han, L. Comparing machine learning classifiers for object-based land cover classification using very high resolution imagery. *Remote Sens.* **2014**, *7*, 153–168. [[CrossRef](#)]
32. Rodriguez-Galiano, V.F.; Ghimire, B.; Rogan, J.; Chica-Olmo, M.; Rigol-Sanchez, J.P. An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS J. Photogramm. Remote Sens.* **2012**, *67*, 93–104. [[CrossRef](#)]
33. Inglada, J.; Arias, M.; Tardy, B.; Hagolle, O.; Valero, S.; Morin, D.; Dedieu, G.; Sepulcre, G.; Bontemps, S.; Defourny, P.; et al. Assessment of an operational system for crop type map production using high temporal and spatial resolution satellite optical imagery. *Remote Sens.* **2015**, *7*, 12356–12379. [[CrossRef](#)]
34. Belgiu, M.; Drăguț, L. Random forest in remote sensing: A review of applications and future directions. *ISPRS J. Photogramm. Remote Sens.* **2016**, *114*, 24–31. [[CrossRef](#)]
35. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]

36. Abade, N.A.; Júnior, O.A.d.C.; Guimarães, R.F.; de Oliveira, S.N. Comparative analysis of MODIS time-series classification using support vector machines and methods based upon distance and similarity measures in the Brazilian Cerrado–Caatinga boundary. *Remote Sens.* **2015**, *7*, 12160–12191. [CrossRef]
37. Kotsiantis, S.B.; Zaharakis, I.D.; Pintelas, P.E. Machine learning: A review of classification and combining techniques. *Artif. Intell. Rev.* **2006**, *26*, 159–190. [CrossRef]
38. Soma, K.; Mori, R.; Sato, R.; Furumai, N.; Nara, S. Simultaneous Multichannel Signal Transfers via Chaos in a Recurrent Neural Network. *Neural Comput.* **2015**, *27*, 1083–1101. [CrossRef] [PubMed]
39. Linzen, T.; Dupoux, E.; Goldberg, Y. Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies. *TACL* **2016**, *4*, 521–535.
40. Bengio, Y.; Courville, A.C.; Vincent, P. Representation Learning: A Review and New Perspectives. *IEEE TPAMI* **2013**, *35*, 1798–1828. [CrossRef] [PubMed]
41. Graves, A.; Mohamed, A.; Hinton, G.E. Speech recognition with deep recurrent neural networks. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vancouver, BC, Canada, 26–31 May 2013; pp. 6645–6649.
42. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
43. Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM TIST* **2011**, *2*, 27. [CrossRef]
44. Chollet, F. Keras. Available online: <https://github.com/fchollet/keras> (accessed on 10 May 2015).
45. Dauphin, Y.N.; de Vries, H.; Chung, J.; Bengio, Y. RMSProp and equilibrated adaptive learning rates for non-convex optimization. *arXiv* **2015**, arXiv:1502.04390.
46. Zhang, L.; Du, B. Deep Learning for Remote Sensing Data: A Technical Tutorial on the State of the Art. *IEEE Geosci. Remote Sens. Mag.* **2016**, *4*, 22–40. [CrossRef]
47. Inglada, J.; Vincent, A.; Arias, M.; Tardy, B.; Morin, D.; Rodes, I. Operational High Resolution Land Cover Map Production at the Country Scale Using Satellite Image Time Series. *Remote Sens.* **2017**, *9*, 95. [CrossRef]
48. Kussul, N.; Lavreniuk, M.; Skakun, S.; Shelestov, A. Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 778–782. [CrossRef]
49. Tang, Y.; Zhang, Y.Q.; Chawla, N.V.; Krasser, S. SVMs modeling for highly imbalanced classification. *IEEE Trans. Syst. Man Cybern. Part B* **2009**, *39*, 281–288. [CrossRef] [PubMed]
50. Zhao, Z.Q. A novel modular neural network for imbalanced classification problems. *Pattern Recognit. Lett.* **2009**, *30*, 783–788. [CrossRef]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).