

Combining Sentinel-1 and Sentinel-2 Satellite Image Time Series for land cover mapping via a multi-source deep learning architecture

Dino Ienco, R. Interdonato, R. Gaetano, Dinh Ho Tong Minh

▶ **To cite this version:**

Dino Ienco, R. Interdonato, R. Gaetano, Dinh Ho Tong Minh. Combining Sentinel-1 and Sentinel-2 Satellite Image Time Series for land cover mapping via a multi-source deep learning architecture. ISPRS Journal of Photogrammetry and Remote Sensing, 2019, 158, pp.11-22. 10.1016/j.isprsjprs.2019.09.016 . hal-02609707

HAL Id: hal-02609707
https://hal.inrae.fr/hal-02609707

Submitted on 20 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Version of Record: https://www.sciencedirect.com/science/article/pii/S0924271619302278
Manuscript_9da0dbc3925c4768842475109cf29752

Combining Sentinel-1 and Sentinel-2 Satellite Image Time Series for land cover mapping via a multi-source deep learning architecture

Dino Ienco*

IRSTEA, UMR TETIS
LIRMM, University of Montpellier, Montpellier

Roberto Interdonato

CIRAD, UMR TETIS, Montpellier

Raffaele Gaetano

CIRAD, UMR TETIS, Montpellier

Dinh Ho Tong Minh

IRSTEA, UMR TETIS, University of Montpellier, Montpellier

Abstract

The huge amount of data currently produced by modern Earth Observation (EO) missions has allowed for the design of advanced machine learning techniques able to support complex Land Use/Land Cover (LULC) mapping tasks. The Copernicus programme developed by the European Space Agency provides, with missions such as Sentinel-1 (S1) and Sentinel-2 (S2), radar and optical (multi-spectral) imagery, respectively, at 10 m spatial resolution with revisit time around 5 days. Such high temporal resolution allows to collect Satellite Image Time Series (SITS) that support a plethora of Earth surface monitoring tasks. How to effectively combine the complementary information provided by such sensors remains an open problem in the remote sensing field. In this work,

*Corresponding author
Email addresses: dino.ienco@irstea.fr (Dino Ienco), roberto.interdonato@cirad.fr (Roberto Interdonato), raffaele.gaetano@cirad.fr (Raffaele Gaetano), dinh.ho-tong-minh@irstea.fr (Dinh Ho Tong Minh)

© 2019 published by Elsevier. This manuscript is made available under the CC BY NC user license
https://creativecommons.org/licenses/by-nc/4.0/

we propose a deep learning architecture to combine information coming from S1 and S2 time series, namely TWINNS (TWIn Neural Networks for Sentinel data), able to discover spatial and temporal dependencies in both types of SITS. The proposed architecture is devised to boost the land cover classification task by leveraging two levels of complementarity, i.e., the interplay between radar and optical SITS as well as the synergy between spatial and temporal dependencies. Experiments carried out on two study sites characterized by different land cover characteristics (i.e., the *Koumbia* site in Burkina Faso and *Reunion Island*, a overseas department of France in the Indian Ocean), demonstrate the significance of our proposal.

Keywords: Satellite Image Time Series, Deep Learning, Land Cover Classification, Sentinel-2, Sentinel-1, Data fusion

---

1. Introduction

In recent years, monitoring the Earth surface has become possible thanks to the development of several modern Earth Observation (EO) systems continuously providing massive amounts of satellite data.

A notable example is represented by the Copernicus programme developed by the European Space Agency (ESA), consisting of several constellations of satellites (i.e., the Sentinel missions), which monitor different aspects of the Earth Surface. In the context of area monitoring tasks, the Sentinel-1 (S1) and Sentinel-2 (S2) missions are of particular interest, since they provide publicly available radar (S1) and optical/multi-spectral (S2) satellite imagery with an high revisit time. The Sentinel-1 mission (composed of two satellites, Sentinel-1A and Sentinel-1B) is operating day and night, performing dual polarization C-band synthetic aperture radar (SAR) imaging acquired at a global scale in Terrain Observation with Progressive Scan (TOPS) mode with a revisit period of 6 days.

The Sentinel-2 mission involves a constellation of two twin satellites (Sentinel-2A and Sentinel-2B), supplying optical information ranging from visible to near

and medium infrared with a spatial resolution between 10 m and 20 m with a revisit period between 5 and 10 days.

Thanks to the high revisiting period, such high spatial resolution satellite images can be effectively organized in Satellite Image Time Series (SITS), which represent a practical tool to monitor Earth Surface evolution through time, supporting a wide number of application domains. For this reason, SITS have been successfully exploited to address tasks related to ecology [1, 2], agriculture [3, 4, 5], mobility, health, risk assessment [6], land management planning [7], forest [8, 9] and natural habitat monitoring [10, 11].

In recent years, Sentinel-1 and Sentinel-2 SITS have been successfully exploited in the context of Land Use/Land Cover (LULC) mapping, demonstrating how the availability of such radar and optical SITS is beneficial in this domain. Some notable examples include the use of optical S2 SITS to produce land cover maps at country scale [7] and to characterize grassland areas as a proxy indicator for biodiversity, food production, and global carbon cycle [1]. Radar S1 SITS have also been effectively applied to different LULC-related tasks, such as analyzing the impact of seasonality in urban land cover mapping [12] and mapping the quality of the land cover in winter season [13].

An attractive challenge in the remote sensing community is how to effectively combine the properties of surface materials provided by the optical sensor (S2) and the structural characteristics of landscape elements provided by the radar sensor (S1), i.e., two aspects that can be considered complementary with respect to the land cover mapping task. Combinations of optical and radar data have shown to perform better than the single sensors in different scenarios, such as grassland [14] and crop [15] monitoring, change detection [16], urban mapping [17], wildfire assessment [18] and detection of invasive plants [19].

As regards LULC mapping, several approaches focus on data fusion techniques, i.e., on the opportunity to integrate the information contained in radar and optical data before processing it [20]. In [21] S1 images are combined with Landsat-8 optical images in order to produce an enhanced forest cover composite (i.e., by suppressing the green component over the non-forested vegetative

areas). In [22] a multimodal image fusion approach based on Latent Semantic Analysis is proposed, in order to deal with unsupervised land-cover mapping. In general LULC tasks, proposed approaches have often exploited standard machine learning techniques (i.e., Random Forest, SVM) on a simple concatenation of radar and optical input data [23, 24, 25, 26, 27, 28, 29] without really leveraging the interplay between these two sources of information. Even though these approaches already prove how combining radar and optical data can improve performance over the use of a single sensor, they treat the two data sources as completely independent from each other. Furthermore, standard approaches ignore both spatial and temporal dependencies that may be present in the data, and they do not explicitly leverage the complementarity carried out by radar and optical SITS.

Models based on artificial neural networks, i.e., deep learning based models, are gaining attention in the remote sensing domain [30, 5, 31, 32, 33]. The main attractive of these models is that they are able to *learn* features from the input data optimized for a specific task (e.g., image classification), by simultaneously training the associated classifier. Moreover, they can be exploited to discover spatial as well as temporal dependencies in SITS. Most commonly used deep learning models are convolutional [30] (CNNs) and recurrent [34] (RNNs) neural networks, which focus respectively on the analysis of spatial and temporal information in the data.

Recently, approaches incorporating both recurrent and convolutional operations were also proposed to deal with spatio-temporal data [35], namely convRNN models (e.g., ConvGru or ConvLSTM). Those architectures extend recurrent neural networks including convolutional operations in the recurrent unit and working on sequences of multi-dimensional data instead of sequences of vectors. These solutions are starting to get attention in the field of remote sensing, for instance, in [36] the authors propose to use convRNN to tackle land cover classification from a Sentinel-2 SITS modeling the task as semantic segmentation [37].

Combinations of optical and radar satellite images based on Deep learning

techniques have been proposed to address tasks such as optical image simulation [38], change detection [39] and river discharge estimation [40]. Nevertheless, the same opportunity has not been fully exploited yet in the context of LULC classification tasks. A first attempt has been made in [5], where stacked Sentinel-1 and Landsat-8 satellite images are used to feed a CNN-based classifier. However, the proposed CNN-based approach is rather simple, as it does not take into account temporal dependencies, and does not fully exploit the opportunity to learn features from radar and optical data separately.

In a previous work [41], we have shown how an architecture based on the combination of CNN and RNN models can help to leverage both spatial and temporal dependencies in optical SITS, thus being beneficial for LULC mapping tasks. In this work, our ambition is to introduce a further level of interaction based on the use of SITS coming from multiple sensors. The proposed TWINNS (TWIn Neural Networks for Sentinel data) architecture, is in fact devised to boost the land cover classification task by exploiting two levels of complementarity: the one between radar (S1) and optical (S2) satellite images, and the one between spatial and temporal dependencies in each image type. While the former point is possible due to the fact that specific and complementary per-source features are extracted, the latter point is addressed by exploiting Convolutional as well as Recurrent Neural Networks to manage spatial autocorrelation and temporal dependencies, respectively.

The rest of the article is structured as follows: the study sites and the associated data are introduced in Section 2; Section 3 describes the deep learning architecture for land cover classification from radar/optical SITS data, while the experimental setting and the evaluations are carried out and discussed in Section 4. Finally, Section 5 concludes the work.

2. Data

The analysis was carried out on *Reunion Island*, a French overseas department located in the Indian Ocean and *Koumbia*, a rural municipality in the

province of Tuy, Burkina Faso.

The *Reunion Island* dataset consists of a time series of 24 S1 images and a time series of 34 S2 images acquired between April 2016 and May 2017. The *Koumbia* dataset consists of a time series of 29 S1 images and a time series of 23 S2 images acquired between January 2016 and December 2016.

Considering the radar imagery (S1), images are acquired in TOPS mode with dual-polarization (VV+VH). The backscatter images were generated and radiometrically calibrated using parameters included in the S1 SAR header, then coregistered with the S2 time series. The pixel size of the orthorectified image data is 10 m. After geocoding, all backscatter images are converted to the logarithm dB scale, normalized to values between 0-255 (8 bits).

All the S2 images we used are those provided at level 2A by the THEIA pole[1] and preprocessed in surface reflectance via the *MACCS-ATCOR Joint Algorithm* [42] developed by the National Centre for Space Studies (CNES). Bands at 20m resolution were resampled at 10m via bicubic interpolation. A preprocessing was performed to fill cloudy observations through a linear multi-temporal interpolation over each band (cfr. *Temporal Gapfilling*, [7]), and six radiometric indices were calculated for each date: NDVI, NDWI, brightness index (BI), NDVI and NDWI of infrared means (MNDVI and MNDWI), and vegetation index Red-Edge (RNDVI) [7, 43]. A total of 16 variables (10 surface reflectances plus 6 indices) are considered for each pixel of each image in the time series.

The spatial extent of the *Reunion Island* site is 6 656 × 5 913 pixels corresponding to 3 935 Km^2 while the extent for the *Koumbia* site is 5 253 × 4 797 pixels corresponding to 2 519 Km^2. Considering the first dataset, reference data for classification was built from various sources: (i) the *Registre parcellaire graphique* (RPG)[2] reference data of 2014, (ii) GPS records from June 2017 and

[1]Data are available via `http://theia.cnes.fr`

[2]RPG is part of the European Land Parcel Identification System (LPIS), provided by the French Agency for services and payment

(iii) photo interpretation of the VHR (Very High Resolution) image conducted by an expert, with knowledge of the territory, for distinguishing between natural and urban spaces.

As for the second dataset, the reference database is a collection of (i) digitized plots from a GPS field mission performed in October 2016 and mostly covering classes within cropland and (ii) additional reference plots on non-crop classes obtained by photo-interpretation by an expert.

For S1, images are acquired in TOPS mode with dual-polarization (VV+VH). The backscatter images were generated and radiometrically calibrated using parameters included in the S1 SAR header, then coregistered with the S2 time series. The pixel size of the orthorectified image data is 10 m. After geocoding, all backscatter images are converted to the logarithm dB scale, normalized to values between 0-255 (8 bits).

Class	Label	# **Polygons**	# **Pixels**
0	*Crop cultivations*	380	12 090
1	*Sugar cane*	496	84 136
2	*Orchards*	299	15 477
3	*Forest plantations*	67	9 783
4	*Meadow*	257	50 596
5	*Forest*	292	55 108
6	*Shrubby savannah*	371	20 287
7	*Herbaceous savannah*	78	5 978
8	*Bare rocks*	107	18 659
9	*Urbanized areas*	125	36 178
10	*Greenhouse crops*	50	1 877
11	*Water surfaces*	96	7 349
12	*Shadows*	38	5 230

Table 1: Per Class ground truth statistics for the Reunion Island Dataset

Class	Label	# **Polygons**	# **Pixels**
0	*Annual Cropland*	671	31 075
1	*Fallows*	57	1 808
2	*Natural Forest*	64	15 843
3	*Savannah*	87	25 156
4	*Grassland*	142	12 883
5	*Rocks*	29	852
6	*Built up*	71	1 096
7	*Water*	16	1 410

Table 2: Per Class ground truth statistics for the Koumbia Dataset

2.1. Ground Truth Statistics

Considering both datasets, ground truth comes in GIS vector file format containing a collection of polygons each attributed with a unique land cover class label. To ensure a precise spatial matching with image data, all geometries have been suitably corrected by hand using the corresponding Sentinel-2 images as reference. Successively, the GIS vector file containing the polygon information has been converted in raster format at the Sentinel-2 spatial resolution (10m).

The final ground truth data includes 322 748 pixels (resp. 2 656 objects) corresponding to an area of 32.27 Km^2 (0.8% of the total surface) distributed over 13 classes for the *Reunion Island* dataset (Table 1) and 90 123 pixels (resp. 1 137 objects) corresponding to an area of 9.12 Km^2 (0.3% of the total surface) distributed over 8 classes for the *Koumbia* benchmark (Table 2).

3. TWINNS

Figure 1 shows a visual representation of the proposed TWINNS architecture. The model takes as input S1 and S2 SITS (which may have different length) that are fed to two structurally identical streams. Each stream is composed of two branches consisting of two different neural networks: an Attentive Convolutional Gated Recurrent Neural Network (ConvGRU, top part of

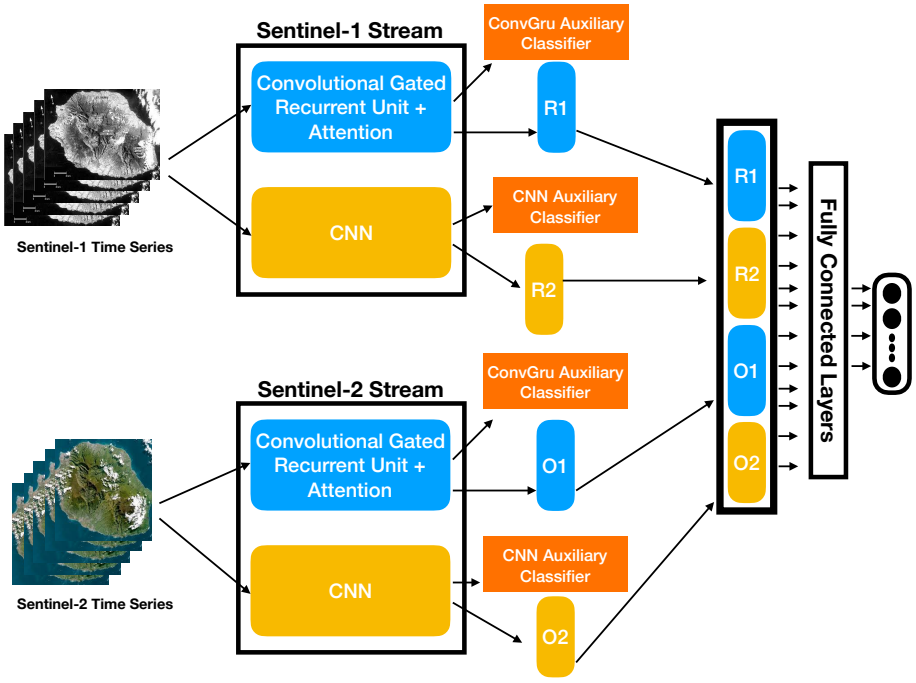


Figure 1: General overview of the TWINNS Deep Learning architecture. A radar SITS of n timestamps and an optical SITS of m timestamps are fed to two separate streams. The per source features $R1$ and $R2$ (resp. $O1$ and $O2$) for the radar (resp. optical) SITS are successively extracted and combined to perform the final classification. The Sentinel-1 and Sentinel-2 Stream (highlighted in the black boxes) are structurally identical and they are composed by two branches: a Convolutional Gated Recurrent Unit [35] (ConvGRU) with Attention and a CNN. Each branch is also attached to an auxiliary classifier with the aim to boost the predictive performance of each branch. The set of extracted features $(R1, R2, O1, O2)$ are exploited by several Fully Connected Layers to produce the Land Cover classification.

each stream) and a Convolutional Neural network (CNN, bottom part of each stream). Each branch supplies complementary information for the discriminative process since they look at the information from different points of view. The result of each branch is a feature vector summarizing the extracted knowledge. The output of the two couples of ConvGRU and CNN branches is represented by four different sets of features (vectors): two sets of features for the radar SITS ($R1$ and $R2$) and two sets of features for the optical SITS ($O1$ and $O2$). Each

vector is used independently to train an auxiliary classifier (cf. Section 3.3), while the concatenation of all the feature vectors (i.e., in a single feature vector of 3 072 descriptors) is used to fed several fully connected layers that produce the land cover decision. In the following, we will provide a detailed description of the ConvGRU and CNN branches (which, we recall, are separated but structurally identical for the S1 and the S2 stream).

3.1. CNN Branch

For this branch, we took inspiration from the VGG model [44], a well-known network architecture usually adopted to tackle with standard Computer Vision tasks. The basic idea behind this model is to constantly increase the number of filters along the network, as long as a reasonable size of the feature maps has been reached. The CNN Branch involves three convolutional layers with kernel size 3×3, 3×3 and 1×1, respectively. We indicate with l the number of filters of the first convolution, while the second and third convolutional layers have $l \times 2$ filters. Each convolutional layer is applied on the valid portion of the image (without any kind of padding) and it is associated with a convolution combined with a Rectifier Linear Unit (ReLU) activation function [45] to induce non-linearity. Successively, a batch normalization step [46] is employed to accelerate the network convergence and reduce the internal covariate shift. Finally, we adopt a Dropout [47] with a drop rate equal to 0.4, i.e., 40% of the neurons are randomly deactivated at each propagation step.

The ReLU activation function is defined as follows:

$$ReLU(x) = Max(0, z) \tag{1}$$

where, in our case, $z = W \circledast x + b$ and $\circledast$ is the convolution operator. This activation function is defined on the positive part of the linear transformation of its argument ($W \circledast x + b$) where x is the input information and W and b are parameters learned by the neural network model. The choice of ReLU non linearities is motivated by two factors: i) the good convergence properties it guarantees and ii) the low computational complexity it provides [45]. The

output of this branch is a feature vector cnn_{feat} that summarizes the spatial information, so allowing to easily combine together different timestamps of the SITS.

3.2. Attentive Convolutional Gated Recurrent Neural Network Branch

Motivated by recent research results [35, 36, 48, 49, 50], here we introduce a new convRNN unit to manage spatio-temporal information from (radar or optical) SITS data. In our model, we choose the GRU unit (Gated Recurrent Unit) introduced in [51] as recurrent unit and we integrate two convolutional layers inside. redThe GRU unit is preferred to the LSTM one since it has already shown its effectiveness in the field of remote sensing considering optical [32] and radar [52] land cover classification via multitemporal spatial data and hyperspectral data analysis [53]. Furthermore, the Gated Recurrent Unit network involves a lower number of parameters to learn compared to the LSTM unit. Furthermore, due to the fact that we consider input patches of size $5 \times 5 \times dim$ (where dim is the number of spectral bands describing the patch) we consider, for our *convGRU* unit, convolutions with a kernel 3×3 on the valid domain of the patch. After two convolutions, the obtained feature maps have a spatial extent of size 1×1 and they can be considered as simple vectors.

Equations 2 to 6 formally describe the *convGRU* neuron we propose.

$$conv^1_{t_i} = BN(ReLU(x_{t_i} \circledast W_{c^1 h} + b^1_c)) \tag{2}$$

$$conv^2_{t_i} = BN(ReLU(conv^1_{t_i} \circledast W_{c^2 h} + b^2_c)) \tag{3}$$

$$z_{t_i} = \sigma(W_{zx} conv^2_{t_i} + W_{zh} h_{t_{i-1}} + b_z) \tag{4}$$

$$r_{t_i} = \sigma(W_{rx} conv^2_{t_i} + W_{rh} h_{t_{i-1}} + b_r) \tag{5}$$

$$h_{t_i} = z_t \odot h_{t-1} + \tag{6}$$
$$(1 - z_{t_i}) \odot \tanh(W_{hx} conv^2_{t_i} + W_{hr}(r_t \odot h_{t_{i-1}}) + b_h)$$

where BN is the batch normalization operation, $ReLU$ is the rectifier linear function, $\circledast$ is the convolution operation, $\odot$ is the element-wise multiplication

(also known as Hadamard product) while σ and tanh represent Sigmoid and Hyperbolic Tangent functions, respectively. As we explained just before, in our case, $conv^2_{t_i}$ can be considered as a vector and then employed in equations 4, 5 and 6 that are the standard operations performed in a GRU unit. Furthermore, we have also considered Dropout for the $ConvGru$ unit since we have empirically observed that such component of our architecture is also prone to overfit the data distribution. Also in this case we use a drop rate equals to 0.4.

Finally, we couple our Convolutional Gated Recurrent Unit with an *attention* mechanism. redAttention mechanisms [54] are widely used in automatic signal processing (1D signal, language or 2D signal) and they allow to join together the information extracted by a recurrent neural network model at different time stamps. Intuitively, the attention mechanism supports the model to focus (give more attention) on important part of the signal and, discarding useless portion of the information [54].

The output returned by the *convGRU* model is a sequence of learned feature vectors for each time stamp (h_{t_1},..., h_{t_N}) where each h_{t_i} has the same dimension d. Their matrix representation $H \in \mathbb{R}^{T,d}$ is obtained by vertically stacking the set of vectors. The attention mechanism allows us to combine together these different vectors h_{t_i}, in a single one rnn_{feat}, to attentively combine the information returned by the GRU unit at each time stamp. The attention formulation we use, starting from a sequence of vectors encoding the learned descriptors (h_{t_1},..., h_{t_T}), is the following one:

$$v_a = tanh(H \cdot W_a + b_a) \tag{7}$$

$$\omega = SoftMax(v_a \cdot u_a) \tag{8}$$

$$rnn_{feat} = \sum_{i=1}^{T} \omega_i \cdot h_{t_i} \tag{9}$$

where matrix $W_a \in \mathbb{R}^{d,d}$ and vectors $b_a, u_a \in \mathbb{R}^d$ are parameters learned during the process. These parameters allow to combine the vectors contained in matrix H. The purpose of this procedure is to learn a set of weights (ω_{t_1},..., ω_{t_T}) to weight the contribution of each time stamp h_{t_i}. The $SoftMax(\cdot)$ [32]

function is used to normalize weights ω so that their sum is equal to 1. The output of the attentive Convolutional Gated Recurrent branch is the feature vector rnn_{feat} that encodes temporal information for time series associated to pixel i.

The features extracted by each branch of the two streams of the architecture are combined by concatenation (cnn^{S1}_{feat}, rnn^{S1}_{feat} ,cnn^{S2}_{feat} and rnn^{S2}_{feat}). Many other combination techniques are possible (e.g., sum, gating, and so on) but we rely on standard concatenation following recent practices introduced by works on multi-source remote sensing analysis [55, 56, 57].

3.3. Training of TWINNS model

One of the advantages of deep learning approaches, compared to standard machine learning methods, is the ability to link, in a single pipeline, the feature extraction step and the associated classifier [30]. This ability is particularly important when different flows of information need to be combined, as in our scenario where we need to couple different representations coming from different sources (i.e., radar and optical SITS). In addition, the different features learned via multiple non-linear combination of the radiometric information are optimized for the specific task at hand, i.e., land cover mapping.

To further strengthen the complementarity and the discriminative power of the learned features for each branch, we adapt the technique proposed in [58] to our problem. In [58], the authors propose to learn two complementary representations (using two convolutional networks) from the same image. The discriminative power is enhanced by two auxiliary classifiers, linked to each group of features, in addition to the classifier that uses the merged information. The complementarity is enforced by alternating the optimization of the parameters of the two branches. Such kind of approach has already shown its effectiveness in remote sensing analysis focusing on land cover mapping considering multi-view analysis on Sentinel-2 SITS [41] as well as multi-source analysis coupling SITS and Very High Spatial Resolution imagery [59]. In our case, we manage each information source via two processing branches that differ from each other in

terms of employed deep learning strategy (i.e., previously described ConvGRU and CNN branches). This strategy results in four sets of information to merge and exploit together.

More in detail, the classifier that exploits the full set of features is fed by concatenating the output features of all the branches :cnn_{feat}^{S1}, rnn_{feat}^{S1} ,cnn_{feat}^{S2} and rnn_{feat}^{S2}. Due to the number of different branches we have to deal with, the learning process involves the optimization of five different classifiers at the same time, defined on the same set of classes. Among the five classifiers, one deals with the combination of all the feature sets, while the others are dedicated to each one of the four feature sets.

The cost function associated to our model is :

$$L_{total} = L_{fus}([cnn_{feat}^{S1}, rnn_{feat}^{S1}, cnn_{feat}^{S2}, rnn_{feat}^{S2}]) \tag{10}$$

$$+ \alpha * \sum_{s \in \{S1,S2\}} \sum_{p \in \{rnn,cnn\}} L_{s,p}(p_{feat}^{s}) \tag{11}$$

where $L_{s,p}(p_{feat}^{s}$ is the loss function associated to the classifier fed with the features obtained by the branch p_{feat}^{s} (cnn_{feat}^{S1}, rnn_{feat}^{S1} ,cnn_{feat}^{S2} and rnn_{feat}^{S2}) and the α parameter is used to weight the contribution of auxiliary classifiers in the backpropagation process.

For the classifier considering the concatenation of the four feature sets we adopt two fully connected layers of 1 024 neurons with ReLU activation function, plus a final output layer with as many neurons as the number of land cover classes to predict. For the four auxiliary classifiers we apply a linear fully connected layer with as many neurons as the number of land cover classes. In order to produce a probability distribution over the class labels, the SoftMax activation function is finally applied [30] on the output layer of all the five classifiers.

Each cost function is modeled through categorical cross entropy, a typical choice for multi-class supervised classification tasks [32]. As highlighted in [58], the auxiliary loss functions operate a regularization that forces, within the network, the features produced by each feature extractor to be discriminative alone

(i.e., independently from each other).

L_{total} is optimized end-to-end. Once the network has been trained, the inference is carried out considering the classifier trained on the whole set of feature as well as the four auxiliary classifiers. Similarly to what done concerning the loss function, we adopt the same strategy to compute the classification score at inference time:

$$score = score_{fus} + \alpha * \sum_{s \in \{S1,S2\}} \sum_{p \in \{rnn,cnn\}} score_{s,p} \tag{12}$$

where $score_{fus}$ and $score_{s,p}$ are the predictions of the main classifier and a generic auxiliary classifier, respectively. Empirically, the α hyperparameter is set to 0.5 for both training and inference.

4. Experiments

In this section, we describe and discuss the experimental results obtained on the study sites introduced in Section 2. We carried out several experiments with the aim to supply a complete analysis of the performance of TWINNS. We investigate different aspects: (i) we perform an ablation study in which we assess the importance and the interplay of the different components of our framework, (ii) we perform an in-depth comparative analysis of the performance of TWINNS with respect to competing methods and (iii) we provide a qualitative evaluation considering land cover maps produced by TWINNS and the most valuable competing methods.

4.1. Competing methods

With the aim to evaluate TWINNS and compare its performance with standard and advanced methods, we consider the following competitors:

- A Random Forest classifier that considers as input the stacked Sentinel-1 and Sentinel-2 SITS, following the lead of recent land cover mapping approaches [24, 25, 28]. An example is described by the whole set of

radiometric information (i.e., both radar and optical one). We name such baseline $RF(S1, S2)$.

- A late fusion classifier that trains two Random Forest classifiers (i.e., one for the radar data and one for the optical data) and then makes a fusion at decision level. For each per-source Random Forest, we extract a per-class score vector. Successively, we sum such vectors and we get the class assignment considering the class obtaining the highest score (argmax operation). The per-class score vector is automatically obtained via the Scikit-learn python library[3]. For each class, the vector (which sums up to 1) contains the proportion of trees that predicts such class. We name such competitor $RF_{LF}(S1, S2)$.

- A deep learning classifier representing an adaptation of the approach recently proposed in [36], which demonstrated the high quality performance of ConvLSTM recurrent units [35] in the remote sensing field. To adapt such model to our multi-source scenario, we employ one ConvLSTM per source, concatenating the features obtained from the two branches and, finally, adding two fully connected layers (1 024, 1 024) plus an additional layer to obtain the classification with a linear plus softmax operation. Also in this case, this competitor is learned end-to-end. We name such competitor $2ConvLSTM$.

As a further solution, similarly to what proposed in other recent works [60, 32, 61, 62, 63], we investigate the possibility to use the deep learning architecture to obtain a new data representation for the classification task. To this end, we feed a Random Forest classifier with the features extracted by TWINNS, naming this approach $RF(TWINNS)$.

[3] `https://scikit-learn.org`

4.2. Experimental Settings

As regards the hyperparameters setting of TWINNS (cf. Section 3), for each stream of analysis (radar/Sentinel-1 and optical/Sentinel-2), we need to fix the number of filters for the CNN Branch (l) and the number of filters for the two convolutional layers of the Attentive Convolutional Recurrent Unit. In our scenario, for both benchmarks, we fix the number of filters for the CNN Branch to 256 (resp. 512) for the radar (resp. optical) stream. Considering the Attentive Convolutional Gated Recurrent Neural Network Branch, we fix the number of filters for the two convolutional layers in the recurrent unit to 256 and 512 (resp. 512 and 1 024) for the radar (resp. optical) data source. Due to the fact that optical time series quantitatively contains more information (i.e., in our case each optical image is associated to 16 bands while a radar image has only two bands) we use twice as many filters to analyze optical information as the number used to manage the radar source. We remind that our framework takes as input time series of image patches (radar and optical) of size $5 \times 5 \times dim$ where dim is equal to 2 (resp. 16) for the radar (resp. optical) SITS.

We split the dataset into three parts: training, validation and test set. Training data are used to learn the model, while validation data are exploited for model selection by varying the parameters of each method. Finally, the model that achieves the best accuracy on the validation set is successively employed to perform the classification on the test set. For the *Reunion Island* (resp. *Koumbia*) dataset we use 30% (resp. 50%) of the objects for the training phase, 20% (resp. 30%) of the objects for the validation set while the remaining 50% (resp. 20%) are employed for the test phase. We consider a different splitting percentage for each dataset due to the different benchmark size in terms of labeled pixels. We impose that all the pixels of the same object belong exclusively to one of the splits (training, validation or test) to avoid spatial bias in the evaluation procedure [7].

Considering the models leveraging the Random Forest classifier, we optimize the model via the tuning of two parameters: the maximum depth of each tree and the number of trees in the forest. For the former parameter, we vary it in

the range {20,40,60,80,100} while for the latter one we take values in the set {100, 200, 300,400,500}.

Experiments are carried out on a workstation with an Intel (R) Xeon (R) CPU E5-2667 v4@3.20Ghz with 256 GB of RAM and four TITAN X GPU. All the Deep Learning methods (including TWINNS) are implemented using the Python Tensorflow library, while Random Forest approaches are implemented using Python scikit-learn library.

Considering the TWINNS approach, the average training, test and map production time, on the considered workstation, are 1 205, 10 and 247 minutes on the *Reunion Island* study site and 594, 7 and 200 minutes on the *Koumbia* study site, respectively. The assessment of the classification performances is done considering global precision (*Accuracy*), *F-Measure* [32] and *Kappa* measures. While Accuracy and Kappa measures are standard metrics in remote sensing, here, we also adopt the *F-Measure* assessment criteria due to the fact that this measure is particularly interesting in unbalanced classification tasks like the one we are dealing with. The *F-Measure* is usually employed in Machine Learning to evaluate the classification performances and it is defined as the harmonic average of precision and recall. It reaches its best value at 1 (perfect precision and recall) and worst at 0 [64].

It is known that, depending on the split of the data, the performances of the different methods may vary as simpler or more difficult examples are involved in the training or test set. To alleviate this issue, for each dataset and for each evaluation metric, we report results averaged over ten different random splits performed with the previously presented strategy.

4.3. Ablation Analysis

In this set of experiments we investigate the interplay among the different components of TWINNS, setting up an ablation analysis in which we disentangle the benefits of the different parts of our framework. To this end, we compare TWINNS with several variants of the original model:

(1) Considering only one SITS source at time (radar or optical). We name TWINNS(S1) (resp. TWINNS(S2)) the ablation of our model considering only the radar (resp. optical) stream.

(2) Considering both SITS sources (radar and optical) but only one type of deep learning structure (Convolutional Neural Network or Attentive ConvGru Neural Network). We name $FullCNN$ (resp. $FullRNN$) the variant of our approach that only involves the Convolutional Neural network branch (resp. Attentive ConvGru Neural network branch).

(3) Excluding the use of the four auxiliary classifiers. We name such ablation TWINNS$_{NoAux}$.

Results are reported in Table 3 (*Reunion Island*) and Table 4 (*Koumbia*). Considering the ablations on the SITS sources (i.e., TWINNS(S1) and TWINNS(S2)), on both study sites TWINNS always shows better performances, confirming our hypothesis that radar and optical SITS indeed provide complementary information for the land cover mapping task. As regards relative performances, TWINNS(S2) always outperforms TWINNS(S1), even though results are closer on *Koumbia*. In fact, the area around *Koumbia* is mostly flat, while *Reunion Island* is characterized by a significantly rugged topography. Due to the sensitivity of backscattering with respect to the orientation of reliefs, in the latter case the radar signal is much more heterogeneous, in a way that is unrelated to the spatial distribution of land cover classes. This finally leads to a reduced discriminative potential compared to optical data, which are generally much less sensitive to topographic effects.

Concerning the use of the auxiliary classifiers (TWINNS vs TWINNS$_{NoAux}$), we can note that such architectural detail positively influences the final classification performance on both study sites, with an improvement around 6 points (resp. 5 points) on *Reunion Island* (resp. *Koumbia*). This finding is in line with the results obtained in [41].

Considering the structural ablations on the type of neural models (i.e., $FullCNN$ vs $FullRNN$), we can observe that, on *Reunion Island*, the $FullRNN$

variant outperforms the $FullCNN$ one, while an opposite behavior is evident on *Koumbia*. TWINNS always outperforms its ablations on *Reunion Island* (Table 3), while it performs very closely to the best performing method (i.e., $FullCNN$) on *Koumbia*. The lower performance on *Koumbia* may be related to the smaller size of this benchmark, i.e., the small quantity of training data may cause the ConvGRU neural network to overfit, and results may be highly data dependent (i.e., it can be observed from the relatively high standard deviation that results show significant variations on the different random splits).

	F-Measure	*Kappa*	*Accuracy*
TWINNS(S1)	73.22 ± 1.23	0.6926 ± 0.0144	73.89 ± 1.24
TWINNS(S2)	84.29 ± 1.19	0.8159 ± 0.0143	84.26 ± 1.26
FullCNN	87.69 ± 0.85	0.8560 ± 0.0107	87.71 ± 0.92
FullRNN	88.23 ± 1.43	0.8620 ± 0.0169	88.22 ± 1.45
TWINNS_{NoAux}	83.92 ± 1.05	0.8109 ± 0.0117	83.84 ± 0.97
TWINNS	**89.87** ± 0.65	**0.8814** ± 0.0080	**89.88** ± 0.69

Table 3: Accuracy, F-Measure and Kappa considering different ablations of TWINNS on the *Reunion Island* dataset (average over ten different random splits)

	F-Measure	*Kappa*	*Accuracy*
TWINNS(S1)	80.93 ± 2.18	0.7530 ± 0.0283	81.84 ± 2.13
TWINNS(S2)	81.47 ± 4.12	0.7563 ± 0.0556	81.99 ± 4.30
FullCNN	**86.81** ± 2.38	**0.8303** ± 0.0303	**87.51** ± 2.29
FullRNN	85.90 ± 2.72	0.8186 ± 0.0363	86.65 ± 2.75
TWINNS_{NoAux}	81.87 ± 4.43	0.7631 ± 0.0599	82.49 ± 4.61
TWINNS	86.65 ± 2.50	0.8298 ± 0.0322	87.50 ± 2.44

Table 4: Accuracy, F-Measure and Kappa considering different ablations of TWINNS on the *Koumbia* dataset (average over ten different random splits)

4.4. Comparative evaluation

Table 5 and Table 6 report the results obtained by TWINNS and all competing methods on the *Reunion Island* and *Koumbia* study sites, respectively. We can observe how, on both benchmarks, TWINNS outperforms all the competing methods. However, we can also note that using TWINNS as feature

extractor ($RF(TWINNS)$) provides better land cover mapping on *Reunion Island*, while on *Koumbia* this strategy does not help to ameliorate classification performances. The behavior on the *Koumbia* dataset can be related to the relatively small amount of available labeled data used to train the model. More precisely, our explanation is that the high-dimensional space induced by the representation learned by TWINNS (3 072 features) cannot be effectively leveraged by the Random Forest classifier due to the small amount of labeled pixels.

Concerning the use of standard machine learning approaches (i.e. Random Forest) on multi-source land cover mapping, it should be noted how $RF_{LF}(S1, S2)$ always outperforms the $RF(S1, S2)$ baseline. This is a further example of the fact that training per source classifiers seems to be more adequate for this task than feeding a single model with the full set of heterogeneous information (e.g., concatenated data coming from radar and optical SITS). As regards $2ConvLSTM$, we can note that it outperforms all the Random Forest based methods on *Koumbia* while it shows poor performance on *Reunion Island*. A reason for this opposite behavior can be the different length of the SITS, i.e., *Koumbia* is described by shorter SITS with respect to the ones employed for the characterization of the *Reunion Island* area. Such difference can impact the learning phase of the $2ConvLSTM$ model, since (as all Recurrent Neural Network models) it can be affected by vanishing gradient issues related to gradient propagation during the learning of parameters. This phenomenon is amplified when longer time series are considered, e.g., like in the *Reunion Island* case. Conversely, TWINNS, due to its attention mechanism, is not affected by vanishing issues since the gradient propagation flows directly to all the timestamps.

4.4.1. Per-Class Analysis

Table 7 and Table 8 report on the per class *F-Measure* performance of the different methods for the *Reunion Island* and *Koumbia* study sites, respectively.

Considering *Reunion Island* (Table 7), we can observe that both TWINNS and RF(TWINNS) outperform all competing methods on almost all the land

	F-Measure	*Kappa*	*Accuracy*
$RF(S1, S2)$	86.10 ± 0.58	0.8402 ± 0.0065	86.42 ± 0.54
$RF_{LF}(S1, S2)$	87.73 ± 0.58	0.8611 ± 0.0069	88.27 ± 0.59
$2ConvLSTM$	83.21 ± 0.90	0.8031 ± 0.0103	83.17 ± 0.90
TWINNS	89.87 ± 0.65	0.8814 ± 0.0080	89.88 ± 0.69
$RF(TWINNS)$	**90.07** ± 1.04	**0.8840** ± 0.0124	**90.10** ± 1.07

Table 5: Accuracy, F-Measure and Kappa considering TWINNS and different competing methods on the *Reunion* dataset (average over ten different random splits)

	F-Measure	*Kappa*	*Accuracy*
$RF(S1, S2)$	79.79 ± 5.30	0.7424 ± 0.0694	81.25 ± 5.16
$RF_{LF}(S1, S2)$	84.78 ± 2.36	0.8079 ± 0.0315	86.00 ± 2.35
$2ConvLSTM$	85.73 ± 2.24	0.8165 ± 0.0276	86.48 ± 2.08
TWINNS	**86.65** ± 2.50	**0.8298** ± 0.0322	**87.50** ± 2.44
$RF(TWINNS)$	85.79 ± 2.62	0.8172 ± 0.0351	86.54 ± 2.68

Table 6: Accuracy, F-Measure and Kappa considering TWINNS and different competing methods on the *Koumbia* dataset (average over ten different random splits)

cover classes with the exception of the 8–*Bare rocks* class, where $RF_{LF}(S1, S2)$ obtains slightly better results. It is interesting to observe how the land cover class where TWINNS obtains the larger gain with respect to competitors is 10–*Greenhouse crops* (i.e., 18 points of F-Measure w.r.t. the best competitor). While other methods easily confuse this land cover with 9–*Urbanized areas*, TWINNS is able to better contextualize these objects due to the specific temporal and spatial characteristics they show in optical and radar SITS. Regarding agricultural and forest classes (0–*Crop Cultivations*, 1–*Sugar cane*, 2–*Orchards*, 3–*Forest plantations* and 5–*Forest*), we can note that both TWINNS and RF(TWINNS) clearly outperform all the other approaches. The highest gain is achieved considering the 2–*Orchards* land cover class where TWINNS gains eight points of F-Measure with respect to the best competitors. Similar gains can be observed on 0–*Crop Cultivations* (about 6 points) and 3–*Forest plantations* (about 7 points). On the other hand, TWINNS has shown some difficulties in the identification of Savannah classes (i.e., 6–*Shrubby savannah* and 7–*Herbaceous savannah*), where the best performance is achieved by

RF(TWINNS), followed by $RF_{LF}(S1, S2)$.

As concerns *Koumbia* (Table 8), we can observe that both TWINNS and RF(TWINNS) outperform other competing methods on all the land cover classes. Larger gains with respect to the best competitors correspond to the 5-*Rocks* (about 7 points) and 6–*Built up* (about 3 points) land cover classes. A significant gain (about 4 points) is also obtained on the 1–*Fallows* class, that is known to be highly problematic to detect due to the heterogeneous nature of such type of land cover in this African region. Despite the low absolute performance, it can be observed how all the deep learning approaches (or derived methods) clearly take advantage from the multi-source data and are able to leverage as much as possible the interplay between the radar and optical SITS to classify such land cover. As a side remark, it is interesting to note how the proposed methods outperform the competitors on land cover classes like 3–*Savannah* and 5–*Rocks*, which had shown to pose some problems in the *Reunion Island* case. This underlines how it is difficult to generalize the results of such per-class analysis on different study sites, due to completely different spatial and temporal characteristics (e.g., even on similar land cover classes) and differences in the amount of available training data that may impact on methods' performances.

Method	0–*Crop cultivations*	1–*Sugar cane*	2–*Orchards*	3–*Forest plantations*	4–*Meadow*	5–*Forest*	6–*Shrubby savannah*	7–*Herbaceous savannah*	8–*Bare rocks*	9–*Urbanized areas*	10–*Greenhouse crops*	11–*Water surfaces*	12–*Shadows*
$RF(S1, S2)$	70.16	94.47	72.44	79.6	88.79	86.69	75.94	61.01	80.58	87.28	38.76	97.33	87.97
$RF_{LF}(S1, S2)$	73.06	94.71	77.18	80.0	89.97	87.33	78.78	64.92	**88.69**	90.15	33.33	95.84	85.57
$2ConvLSTM$	62.09	93.5	71.92	78.72	87.8	83.24	66.16	54.03	71.66	86.52	26.83	95.42	82.48
TWINNS	**79.06**	**96.48**	**85.0**	87.4	91.72	**90.51**	76.28	61.96	85.34	91.88	56.69	95.61	**90.8**
$RF(TWINNS)$	77.58	96.26	83.1	**88.53**	**91.81**	90.21	**78.93**	**67.75**	87.01	**92.26**	**57.11**	**97.52**	90.01

Table 7: *F-Measure* per class for the *Reunion Island* Dataset (average over ten random splits)

We also investigate the confusion between each pair of classes and we report, for each competing method, its confusion matrix (Figure 2 and Figure 3). The visual results support the observations drawn for the per-class F-Measure analysis, since the heat maps representing the confusion matrices confirm that

TWINNS and $RF(TWINNS)$ are more precise than competing methods. This consideration emerges from the fact that the corresponding heat maps (Figures 2d and 2e for *Reunion Island* and Figures 3d and 3e for *Koumbia*) have a visible diagonal structure (the dark blue blocks concentrated on the diagonal). This is not the case for the other competitors where the distinction between different classes is less sharp. The confusion matrices are also coherent with quantitative results observed for specific land cover classes, e.g., it is easy to see the lower confusion between 9–*Urbanized areas* and 10–*Greenhouse crops* for the proposed methods w.r.t. other ones.

Method	0–*Annual Cropland*	1–*Fallows*	2–*Natural Forest*	3–*Savannah*	4–*Grassland*	5–*Rocks*	6–*Built up*	7–*Water*
$RF(S1, S2)$	90.01	2.3	76.92	77.35	76.6	58.46	65.32	65.87
$RF_{LF}(S1, S2)$	93.64	2.62	81.05	83.97	81.97	67.41	66.48	73.56
$2ConvLSTM$	93.82	9.35	84.66	83.77	81.49	76.25	71.13	77.78
TWINNS	94.71	**13.8**	**85.74**	**84.59**	82.58	77.02	**74.37**	**79.83**
$RF(TWINNS)$	**94.94**	10.62	82.79	83.36	**82.92**	**83.81**	69.91	78.04

Table 8: *F-Measure* per class for the *Koumbia* Dataset (average over ten random splits)

4.5. Qualitative Inspection of Land Cover Maps

In order to investigate the differences among the land cover classification maps produced by TWINNS and its main competitors (*RF_LF(S1,S2)* and *2ConvLSTM*) from a qualitative point of view, we report in Figures 4 and 5 some representative map classification details on the *Reunion Island* and the *Koumbia* study sites, respectively. For each study site, the land cover map is generated classifying the whole set of pixels covering the area. The full land cover maps obtained on *Reunion Island* and *Koumbia* by using TWINNS, *RF_LF(S1,S2)* and *2ConvLSTM* can be found on our website [4]. With the purpose to supply

[4] `http://mdl4eo.irstea.fr/twinns_maps/`

a reference image with natural colors for the map classification details, we have used the multispectral information obtained from a Very High spatial Resolution (VHR) image acquired on the same area in the interval spanned by the time series. More in detail, for each study site, we used the multispectral bands of a SPOT7 image at a spatial resolution of 6m.

Considering the *Reunion Island* study site, the first example (Figures 4a, 4b, 4c and 4d) depicts the coastal urban area of *Le Port*. It can be noted how the classifier based on *Random Forest* (Figure 4b) is not able to preserve the geometry of the area, showing an evident salt and pepper error. Conversely, both deep learning-based classifiers (Figures 4c and 4d) provide a sharper representation of the city area. As a further detail, TWINNS is able to correctly identify the border of the harbor on the northern part of the city, differently from *2ConvLSTM*. The second example (Figures 4e, 4f, 4g and 4h) shows a coastal area in the south of the Island, near the city of *Saint Pierre*. It is easy to see how TWINNS (Figure 4h) is the only method to correctly identify the greenhouse land cover at the center of the scene, also providing a cleaner representation of the orchards on its left, with respect to its competitors. These results clearly confirm the discussion carried out for the per-class quantitative analysis (cf. Section 4.4.1).

As regards the *Koumbia* study site, the first example (Figures 5a, 5b, 5c and 5d) depicts an area in the heart of the *Mou Forest*, in the southern part of the site. While TWINNS is able to provide a rather clean representation of the mix of *Natural Forest*, *Savannah* and *Grassland* which characterizes this area, both *RF_LF(S1,S2)* (Figure 5b) and *2ConvLSTM* (Figure 5c) tend to incorrectly classify wide forest areas as *Annual Cropland*. The second example (Figures 5e, 5f, 5g and 5h) shows two small villages placed in the western zone of the site. While all methods are somewhat able to identify some parts of the built-up surfaces, TWINNS provides a better classification of the spatial extent of the villages, while both competitors tend to overestimate the *Annual Cropland*, and *2ConvLSTM* also tends to erroneously classify some cropland as built-up surfaces.

To wrap up, the visual inspection is coherent with the quantitative results discussed in Section 4.4. Furthermore, the qualitative analysis confirms again the ability of our proposal, TWINNS, to smartly exploit the complementarity of the Sentinel-1 and Sentinel-2 SITS data sources providing an end-to-end framework to effectively deal with the complexity of multitemporal radar/optical data fusion in the context of land cover mapping.

5. Conclusion

In this paper, a novel Deep Learning architecture has been proposed, namely TWINNS (TWIn Neural Networks for Sentinel data), devised to exploit radar and optical Satellite Image Time Series in the context of a land cover mapping task. The aim is to leverage the complementary representations produced by different sensors, in order to obtain a set of descriptors that helps to better discriminate among the land cover classes, i.e., with respect to similar models exploiting a single type of sensor.

The proposed architecture includes a dedicated stream for each information source (i.e., S1 and S2 SITS). Each stream is composed in turn by a two-branch architecture that processes the SITS via different deep learning basic blocks (i.e., CNN and attentive ConvGru) to enforce a diverse and complete representation of the data that takes into account both spatial and temporal contexts. The final land cover classification is achieved by concatenating the features extracted by each stream. The framework is learned end-to-end from scratch.

Quantitative and qualitative evaluation on two real-world study sites, characterized by different land cover characteristics, has demonstrated the significance of our approach, showing how our proposal outperforms state of the art solutions considering operational settings. As future work, we plan to extend the proposed multi-source scenario by integrating further types of remote sensing data. A first step will be to extend our Deep Learning strategy by combining VHR optical imagery with the S1 and S2 data sources, in order to improve the land cover mapping process by leveraging fine-grained spatial knowledge.

6. Acknowledgements

This work was supported by the French National Research Agency under the Investments for the Future Program, referred as ANR-16-CONV-0004 (DigitAg), the GEOSUD project with reference ANR-10-EQPX-20, the Programme National de Télédétection Spatiale (PNTS, `http://www.insu.cnrs.fr/pnts`), grant n^oPNTS-2018-5, as well as from the financial contribution from the French Ministry of agriculture "Agricultural and Rural Development" trust account.

References

[1] N. Kolecka, C. Ginzler, R. Pazur, B. Price, P. H. Verburg, Regional scale mapping of grassland mowing frequency with sentinel-2 time series, Remote Sensing 10 (8) (2018) 1221.

[2] L. Chen, Z. Jin, R. Michishita, J. Cai, T. Yue, B. Chen, B. Xu, Dynamic monitoring of wetland cover changes using time-series remote sensing imagery, Ecological Informatics 24 (2014) 17–26.

[3] A. Bégué, D. Arvor, B. Bellón, J. Betbeder, D. de Abelleyra, R. P. D. Ferraz, V. Lebourgeois, C. Lelong, M. Simões, S. R. Verón, Remote sensing and cropping practices: A review, Remote Sensing 10 (1) (2018) 99.

[4] B. Bellón, A. Bégué, D. L. Seen, C. A. de Almeida, M. Simões, A remote sensing approach for regional-scale mapping of agricultural land-use systems based on NDVI time series, Remote Sensing 9 (6) (2017) 600.

[5] N. Kussul, M. Lavreniuk, S. Skakun, A. Shelestov, Deep learning classification of land cover and crop types using remote sensing data, IEEE Geosci. Remote Sensing Lett. 14 (5) (2017) 778–782.

[6] S. Olen, B. Bookhagen, Mapping damage-affected areas after natural hazard events using sentinel-1 coherence time series, Remote Sensing 10 (8) (2018) 1272.

[7] J. Inglada, A. Vincent, M. Arias, B. Tardy, D. Morin, I. Rodes, Operational high resolution land cover map production at the country scale using satellite image time series, Remote Sensing 9 (1) (2017) 95.

[8] M. A. Wulder, J. G. Masek, W. B. Cohen, T. R. Loveland, C. E. Woodcock, Opening the archive: How free data has enabled the science and monitoring promise of landsat author links open overlay panel, Remote Sensing of Environment 122 (2012) 2–10.

[9] M. A. Wulder, J. C. White, S. N. Goward, J. G. Masek, J. R. Irons, M. Herold, W. B.Cohen, T. R. Loveland, C. E. Woodcock, Landsat continuity: Issues and opportunities for land cover monitoring, Remote Sensing of Environment 122 (2008) 955–969.

[10] F. Guttler, D. Ienco, J. Nin, M. Teisseire, P. Poncelet, A graph-based approach to detect spatiotemporal dynamics in satellite image time series, ISPRS Journal of Photogrammetry and Remote Sensing 130 (2017) 92–107.

[11] L. Khiali, D. Ienco, M. Teisseire, Object-oriented satellite image time series analysis using a graph-based representation, Ecological Informatics 43 (2018) 52–64.

[12] T. Zhou, M. Zhao, C. Sun, J. Pan, Exploring the impact of seasonality on urban land-cover mapping using multi-season sentinel-1a and GF-1 WFV images in a subtropical monsoon-climate region, ISPRS Int. J. Geo-Information 7 (1) (2018) 3.

[13] D. H. T. Minh, D. Ienco, R. Gaetano, N. Lalande, E. Ndikumana, F. Osman, P. Maurel, Deep recurrent neural networks for winter vegetation quality mapping via multitemporal sar sentinel-1, IEEE GRSL Preprint (-) (2018) –.

[14] P. Dusseux, T. Corpetti, L. Hubert-Moy, S. Corgne, Combined use of multi-temporal optical and radar satellite images for grassland monitoring, Remote Sensing 6 (7) (2014) 6163–6182.

[15] J. Betbeder, M. Laslier, T. Corpetti, E. Pottier, S. Corgne, L. Hubert-Moy, Multi-temporal optical and radar data fusion for crop monitoring: Application to an intensive agricultural area in brittany(france), in: 2014 IEEE Geoscience and Remote Sensing Symposium, IGARSS 2014, Quebec City, QC, Canada, July 13-18, 2014, 2014, pp. 1493–1496.

[16] Q. Gao, M. Zribi, M. J. Escorihuela, N. Baghdadi, Synergetic use of sentinel-1 and sentinel-2 data for soil moisture mapping at 100 m resolution, Sensors 17 (9) (2017) 1966.

[17] G. C. Iannelli, P. Gamba, Jointly exploiting sentinel-1 and sentinel-2 for urban mapping, in: 2018 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2018, Valencia, Spain, July 22-27, 2018, 2018, pp. 8209–8212.

[18] D. Colson, G. P. Petropoulos, K. P. Ferentinos, Exploring the potential of sentinels-1 & 2 of the copernicus mission in support of rapid and cost-effective wildfire assessment, Int. J. Applied Earth Observation and Geoinformation 73 (2018) 262–276.

[19] P. Rajah, J. Odindi, O. Mutanga, Feature level image fusion of optical imagery and synthetic aperture radar (sar) for invasive alien plant species detection and mapping, Remote Sensing Applications: Society and Environment 10 (2018) 198–208.

[20] N. P. Joshi, M. Baumann, A. Ehammer, R. Fensholt, K. Grogan, P. Hostert, M. R. Jepsen, T. Kuemmerle, P. Meyfroidt, E. T. A. Mitchard, J. Reiche, C. M. Ryan, B. Waske, A review of the application of optical and radar remote sensing data fusion to land use mapping and monitoring, Remote Sensing 8 (1) (2016) 70.

[21] R. C. Sharma, K. Hara, R. Tateishi, Developing forest cover composites through a combination of landsat-8 optical and sentinel-1 SAR data for the visualization and extraction of forested areas, J. Imaging 4 (9) (2018) 105.

[22] R. Fernández-Beltran, J. M. Haut, M. E. Paoletti, J. Plaza, A. Plaza, F. Pla, Multimodal probabilistic latent semantic analysis for sentinel-1 and sentinel-2 image fusion, IEEE Geosci. Remote Sensing Lett. 15 (9) (2018) 1347–1351.

[23] M. J. Steinhausen, P. D. Wagner, B. Narasimhan, B. Waske, Combining sentinel-1 and sentinel-2 data for improved land use and land cover mapping of monsoon regions, International Journal of Applied Earth Observation and Geoinformation 73 (2018) 595 – 604.

[24] J. Denize, L. Hubert-Moy, J. Betbeder, S. Corgne, J. Baudry, E. Pottier, Evaluation of using sentinel-1 and -2 time-series to identify winter land use in agricultural landscapes, Remote Sensing 11 (1).

[25] K. V. Tricht, A. Gobin, S. Gilliams, I. Piccard, Synergistic use of radar sentinel-1 and optical sentinel-2 imagery for crop mapping: A case study for belgium, Remote Sensing 10 (10) (2018) 1642.

[26] L. Lu, Y. Tao, L. Di, Object-based plastic-mulched landcover extraction using integrated sentinel-1 and sentinel-2 data, Remote Sensing 10 (11) (2018) 1820.

[27] P. Hedayati, D. Bargiel, Fusion of sentinel-1 and sentinel-2 images for classification of agricultural areas using a novel classification approach, in: 2018 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2018, Valencia, Spain, July 22-27, 2018, 2018, pp. 6643–6646.

[28] J. Erinjery, M. Singh, R. Kent, Mapping and assessment of vegetation types in the tropical rainforests of the western ghats using multispectral sentinel-2 and sar sentinel-1 satellite imagery, Remote Sensing of Environment 216 (2018) 345–354.

[29] A. Whyte, K. P. Ferentinos, G. P. Petropoulos, A new synergistic approach for monitoring wetlands using sentinels -1 and 2 data with object-based

machine learning algorithms, Environmental Modelling and Software 104 (2018) 40–54.

[30] L. Zhang, B. Du, Deep learning for remote sensing data: A technical tutorial on the state of the art, IEEE Geoscience and Remote Sensing Magazine 4 (2016) 22–40.

[31] X. Zhu, D. Tuia, L. Mou, G. X. L. Zhang, F. Xu, F. Fraundorfer, Deep learning in remote sensing: A comprehensive review and list of resources, IEEE Geosci. Remote Sens. Mag. 5 (2017) 8–36.

[32] D. Ienco, R. Gaetano, C. Dupaquier, P. Maurel, Land cover classification via multitemporal spatial data by deep recurrent neural networks, IEEE GRSL 14 (10) (2017) 1685–1689.

[33] H. Lyu, H. Lu, L. Mou, Learning a transferable change rule from a recurrent neural network for land cover change detection, Remote Sensing 8 (6).

[34] Y. Bengio, A. C. Courville, P. Vincent, Representation learning: A review and new perspectives, IEEE TPAMI 35 (8) (2013) 1798–1828.

[35] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, W. Woo, Convolutional LSTM network: A machine learning approach for precipitation nowcasting, in: NIPS, 2015, pp. 802–810.

[36] M. Rußwurm, M. Körner, Multi-temporal land cover classification with sequential recurrent encoders, ISPRS Int. J. Geo-Information 7 (4) (2018) 129.

[37] M. Volpi, D. Tuia, Dense semantic labeling of subdecimeter resolution images with convolutional neural networks, IEEE Trans. Geoscience and Remote Sensing 55 (2) (2017) 881–893.

[38] W. He, N. Yokoya, Multi-temporal sentinel-1 and -2 data fusion for optical image simulation, ISPRS Int. J. Geo-Information 7 (10) (2018) 389.

[39] J. Liu, M. Gong, K. Qin, P. Zhang, A deep convolutional coupling network for change detection based on heterogeneous optical and radar images, IEEE Trans. Neural Netw. Learning Syst. 29 (3) (2018) 545–559.

[40] A. Tarpanelli, E. Santi, M. J. Tourian, P. Filippucci, G. Amarnath, L. Brocca, Daily river discharge estimates by merging satellite optical sensors and radar altimetry through artificial neural network, IEEE Trans. Geoscience and Remote Sensing 57 (1) (2019) 329–341.

[41] R. Interdonato, D. Ienco, R. Gaetano, K. Ose, Duplo: A dual view point deep learning architecture for time series classification, ISPRS Journal of Photogrammetry and Remote Sensing 149 (2019) 91 – 104.

[42] O. Hagolle, M. Huc, D. Villa Pascual, G. Dedieu, A Multi-Temporal and Multi-Spectral Method to Estimate Aerosol Optical Thickness over Land, for the Atmospheric Correction of FormoSat-2, LandSat, VENμS and Sentinel-2 Images, Remote Sensing 7 (3) (2015) 2668–2691.

[43] V. Lebourgeois, S. Dupuy, E. Vintrou, M. Ameline, S. Butler, A. Bégué, A combined random forest and OBIA classification scheme for mapping smallholder agriculture at different nomenclature levels using multisource data (simulated sentinel-2 time series, VHRS and DEM), Remote Sensing 9 (3) (2017) 259.

[44] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, CoRR abs/1409.1556.
URL `http://arxiv.org/abs/1409.1556`

[45] V. Nair, G. E. Hinton, Rectified linear units improve restricted boltzmann machines, in: ICML10, 2010, pp. 807–814.

[46] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: ICML, 2015, pp. 448–456.

[47] G. E. Dahl, T. N. Sainath, G. E. Hinton, Improving deep neural networks for LVCSR using rectified linear units and dropout, in: ICASSP, 2013, pp. 8609–8613.

[48] K. Soma, R. Mori, R. Sato, N. Furumai, S. Nara, Simultaneous multi-channel signal transfers via chaos in a recurrent neural network, Neural Computation 27 (5) (2015) 1083–1101.

[49] T. Linzen, E. Dupoux, Y. Goldberg, Assessing the ability of lstms to learn syntax-sensitive dependencies, TACL 4 (2016) 521–535.

[50] L. Mou, L. Bruzzone, X. X. Zhu, Learning spectral-spatial-temporal features via a recurrent convolutional neural network for change detection in multispectral imagery, IEEE Transactions on Geoscience and Remote Sensing (2018) 1–12.

[51] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, in: EMNLP, 2014, pp. 1724–1734.

[52] E. Ndikumana, D. H. T. Minh, N. Baghdadi, D. Courault, L. Hossard, Deep recurrent neural network for agricultural classification using multitemporal SAR sentinel-1 for camargue, france, Remote Sensing 10 (8) (2018) 1217.

[53] L. Mou, P. Ghamisi, X. X. Zhu, Deep recurrent neural networks for hyperspectral image classification, IEEE TGRS 55 (7) (2017) 3639–3655.

[54] D. Britz, M. Y. Guan, M. Luong, Efficient attention using a fixed-size memory representation, in: EMNLP, 2017, pp. 392–400.

[55] P. Benedetti, D. Ienco, R. Gaetano, K. Ose, R. G. Pensa, S. Dupuy, M3fusion: A deep learning architecture for multi-{Scale/Modal/Temporal} satellite data fusion, CoRR abs/1803.01945.

[56] R. Gaetano, D. Ienco, K. Ose, R. Cresson, Mrfusion: A deep learning architecture to fuse pan and ms imagery for land cover mapping, CoRR abs/1806.11452.

[57] X. Liu, L. Jiao, J. Zhao, J. Zhao, D. Zhang, F. Liu, S. Yang, X. Tang, Deep multiple instance learning-based spatial-spectral classification for PAN and MS imagery, IEEE Trans. Geoscience and Remote Sensing 56 (1) (2018) 461–473.

[58] S. Hou, X. Liu, Z. Wang, Dualnet: Learn complementary features for image recognition, in: IEEE ICCV, 2017, pp. 502–510.

[59] P. Benedetti, D. Ienco, R. Gaetano, K. Ose, R. G. Pensa, S. Dupuy, M^3fusion: A deep learning architecture for multiscale multimodal multitemporal satellite data fusion, IEEE J Sel. Topics in Appl. Earth Observ. and Remote Sensing 11 (12) (2018) 4939–4949.

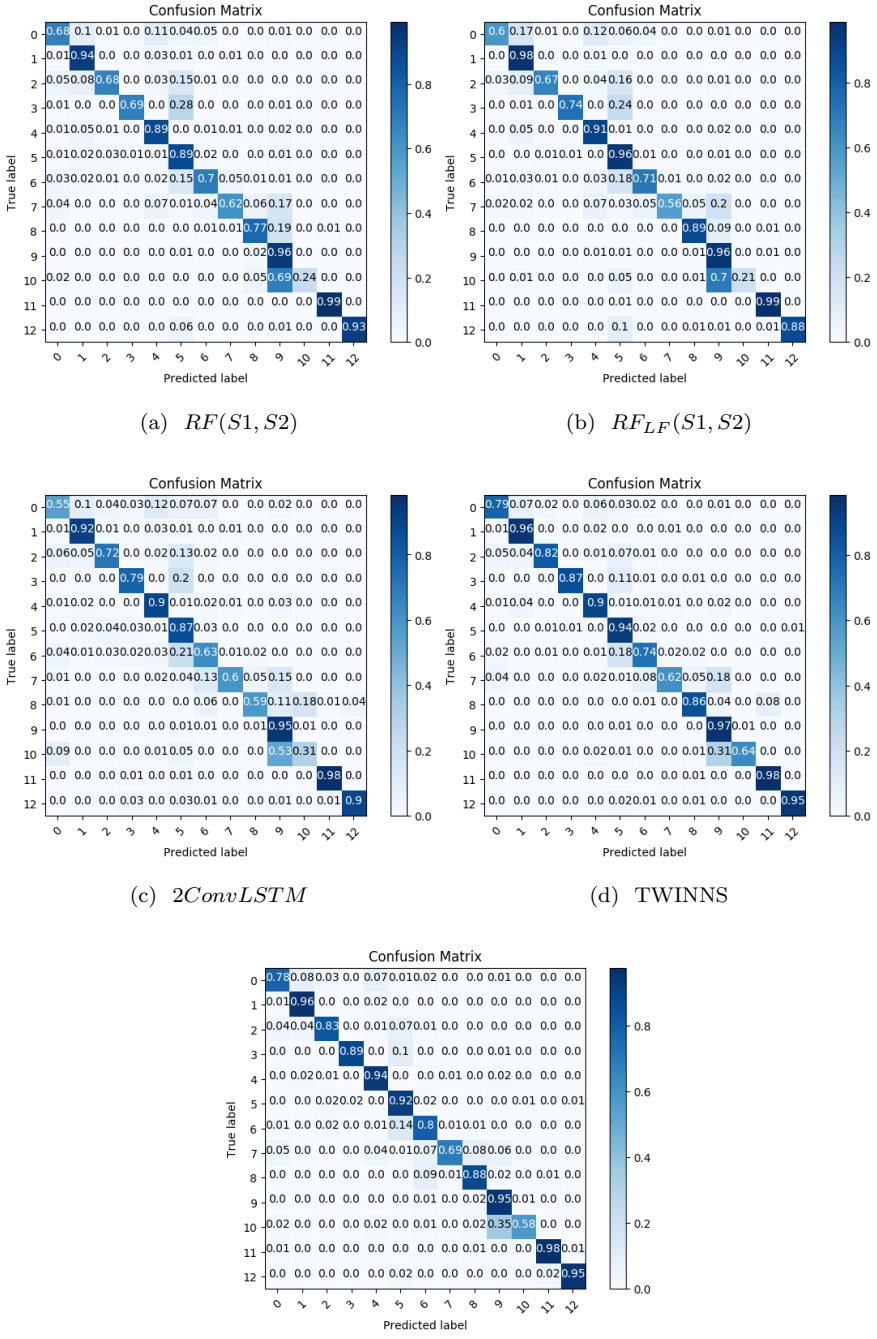
[60] A. Romero, C. Gatta, G. Camps-Valls, Unsupervised deep feature extraction for remote sensing image classification, IEEE Trans. Geoscience and Remote Sensing 54 (3) (2016) 1349–1362.

[61] Q. Weng, Z. Mao, J. Lin, W. Guo, Land-use classification via extreme learning classifier based on deep convolutional features, IEEE Geosci. Remote Sensing Lett. 14 (5) (2017) 704–708.

[62] Q. Liu, R. Hang, H. Song, Z. Li, Learning multiscale deep features for high-resolution satellite image scene classification, IEEE Trans. Geoscience and Remote Sensing 56 (1) (2018) 117–126.

[63] B. Liu, X. Yu, P. Zhang, A. Yu, Q. Fu, X. Wei, Supervised deep feature extraction for hyperspectral image classification, IEEE Trans. Geoscience and Remote Sensing 56 (4) (2018) 1909–1921.

[64] P.-N. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, Addison Wesley, 2005.



(a) $RF(S1, S2)$

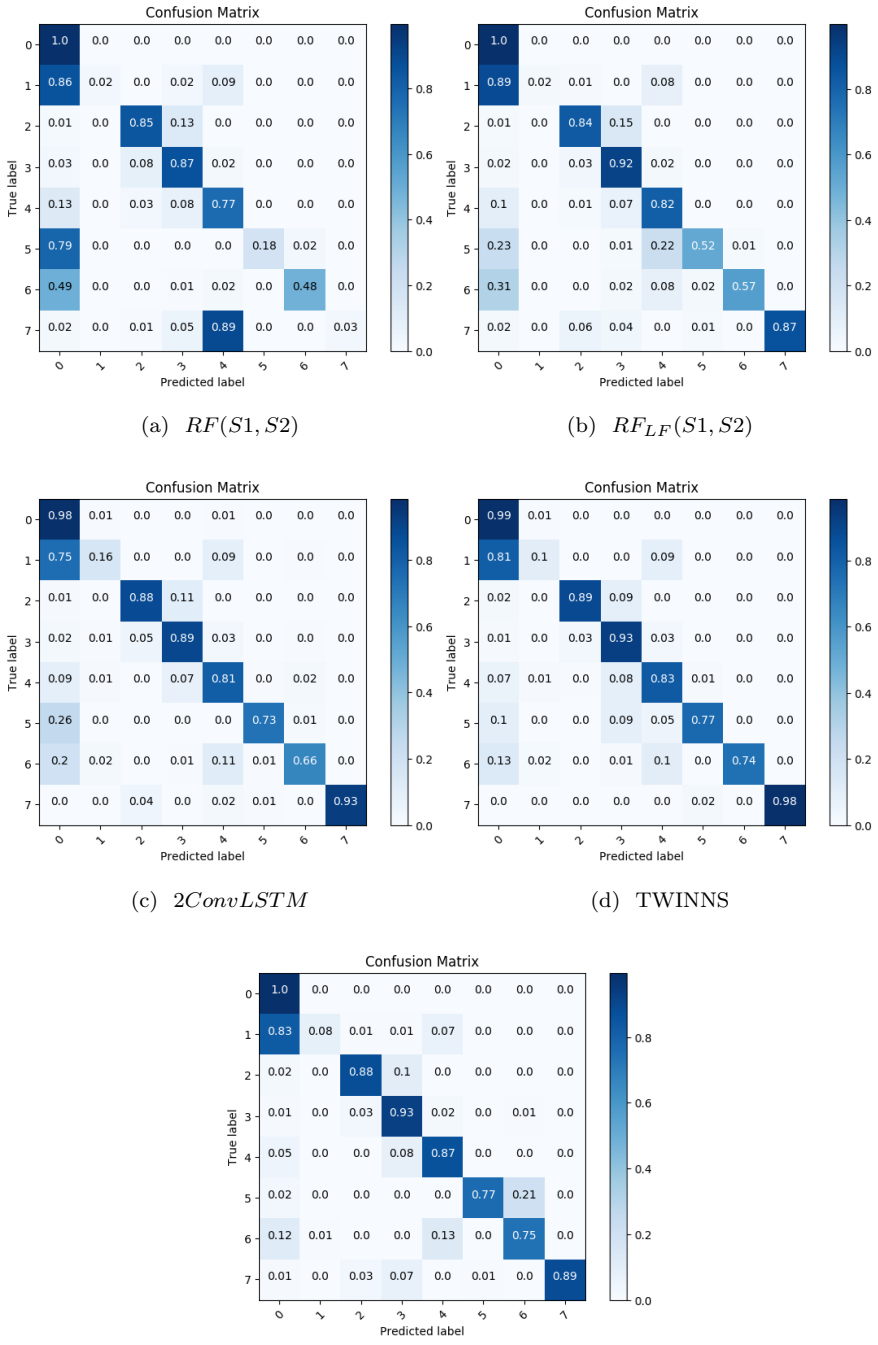
(b) $RF_{LF}(S1, S2)$

(c) $2ConvLSTM$

(d) TWINNS

(e) $RF(TWINNS)$

Figure 2: Heat Maps representing the confusion matrices of: (a) $RF(S1, S2)$, (b) $RF_{LF}(S1, S2)$, (c) $2ConvLSTM$, (d) TWINNS and (e) $RF(TWINNS)$ on the *Reunion Island* study site.



(a) $RF(S1, S2)$

(b) $RF_{LF}(S1, S2)$

(c) $2ConvLSTM$

(d) TWINNS

(e) $RF(TWINNS)$

Figure 3: Heat Maps representing the confusion matrices of: (a) $RF(S1, S2)$, (b) $RF_{LF}(S1, S2)$, (c) $2ConvLSTM$, (d) TWINNS and (e) $RF(TWINNS)$ on the *Koumbia* study site.

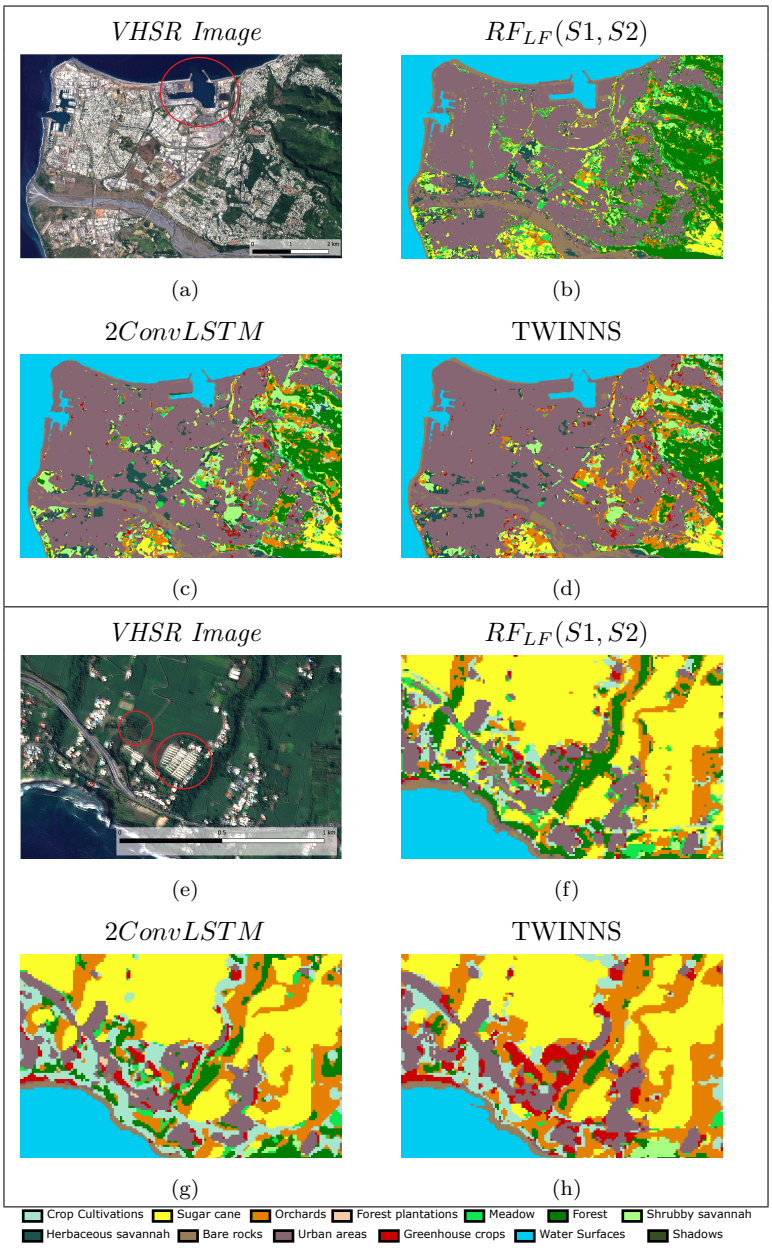


Figure 4: Qualitative investigation of Land Cover Map details produced on the *Reunion Island* study site by *RF_LF(S1,S2)*, *2ConvLSTM* and TWINNS on a coastal urban area (top) and a area containing greenhouses and orchards (bottom) .

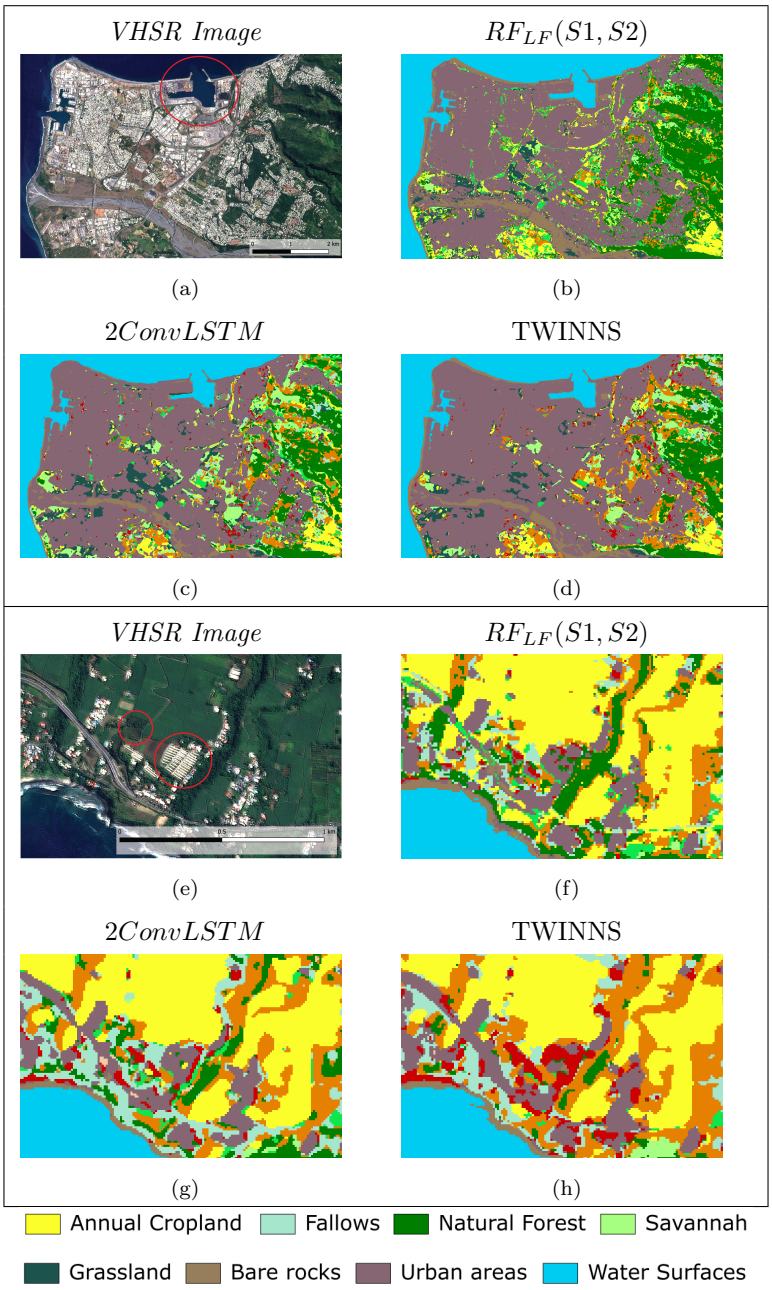


Figure 5: Qualitative investigation of Land Cover Map details produced on the *Koumbia* study site by *RF_LF(S1,S2)*, *2ConvLSTM* and TWINNS on a forest area (top) and a urban area (bottom) .