# A COMPARISON OF FULLY CONVOLUTIONAL AND RECURRENT NETWORKS FOR MULTI-TEMPORAL CROP RECOGNITION USING SAR IMAGES

*Jorge Andrés Chamorro Martínez*[1], *Patrick Nigri Happ*[1], *José David Bermúdez Castro*[1], *Laura Elena Cué La Rosa*[1], *Raul Queiroz Feitosa*[1,2]

[1] Pontifical Catholic University of Rio de Janeiro, Rio de Janeiro, Brazil - {jchamorro,patrick,bermudez,lauracue,raul}@ele.puc-rio.br
[2] State University of Rio de Janeiro, Rio de Janeiro, Brazil

***ABSTRACT***

*With population and food consumption continuously growing, the demand for efficient agricultural crop monitoring systems has been increasing in the last years. Crop dynamics are inherently complex and to model them both spatial and temporal context have to be considered. The increasing availability of timely, precise and cost-effective Remote Sensing data along with the recent development of deep learning techniques for image analysis open up new possibilities for crop monitoring. Motivated by this context, this work presents a comparative analysis of three deep learning architectures for crop recognition: Fully Convolutional Networks, Recurrent Neural Networks and Convolutional Recurrent Neural Networks. The paper reports the results of experiments performed over two datasets: a temperate region near Hanover, Germany and a sub-tropical region in Campo Verde, Brazil. Only SAR data from Sentinel-1 satellite was considered because it is marginally affected by atmospheric conditions. The experiments showed that the tested models achieved state-of-the-art performance.*

***Key words –*** *Convolutional Recurrent Neural Networks, Fully Convolutional Neural Networks, Recurrent Neural Networks, Crop Recognition, Multi-Temporal Analysis.*

## 1. INTRODUCTION

The demand for efficient, comprehensive and precise agriculture intelligence has significantly increased during the past years due to several reasons. For instance, it is necessary to increase the production, in order to feed the nine-billion people predicted by mid-century, as well as to reduce the environmental impact. In this context, crop production information is very important, since it can be used to develop commercial plans, regulate internal stocks and perform customized management decisions [1]. Remote sensing imagery has increasingly been applied for this task, being considered as a cost-effective way for gathering timely, detailed and reliable information over large areas [2].

Crop recognition is challenging because some fields are covered with different types of crops during the year and such practice may be influenced by multiple factors including phenological, ecologic or economic changes. Thus, agricultural areas are characterized by their temporal dynamics as well as their typical spatial patterns [3]. A commonly used method consists of stacking the multi-temporal sequence of images together and training a classifier using information from each individual pixel, but it generally neglects any spatial relationship among neighboring pixels. Alternatively, probabilistic graphical models, such as Conditional Random Fields (CRF), have been applied to crop recognition [4]. They are able to capture spatio-temporal context. However, the methods based on CRF tested so far rely on hand-crafted features.

In the recent years, deep learning models have made breakthroughs in several fields such as speech recognition and computer vision [5]. In remote sensing, these models have also been successfully tested in diverse applications [6]. Such models can be roughly grouped in two main categories: Recurrent Neural Networks (RNN), mostly to model temporal data sequences, and Convolutional Neural Networks (CNN) for understanding spatial context.

Two different RNN models, Long short-term memory (LSTM) and Gated Recurrent Unit (GRU), were applied in [7] for crop classification upon multi-temporal Sentinel-1 data. A similar work is presented in [8], where a Recurrent Convolutional Neural Network was used for the same purpose. In this case, a RNN and CNN were combined by applying a convolutional layer at each image of the sequence. Another type of CNN and RNN combination, known as Convolutional RNN, was applied for crop recognition in [9]. However, this work was based on optical images and tested only data from a temperate climate. In another work ( [10]) a variant of the prior method, called Convolutional LSTM networks (ConvLSTM), were designed to model jointly the spatial and temporal context by replacing the LSTM input-to-state and state-to-state layers by convolutions. In [11], a different network model, known as Fully Convolutional Networks (FCN), was applied for multi-temporal crop recognition from SAR images.

The present work fits in this research line and aims to assess the performance of three deep learning architectures for multi-temporal crop recognition upon SAR data: FCNs, LSTMs and ConvLSTMs. As baseline we present the results obtained by the commonly used image stacking approach [12]. The experiments were conducted over two study areas with very different weather conditions: the first one located in Hanover, Germany with a temperate climate; the second one from a sub-tropical region in Campo Verde municipality, Brazil. To the best of our knowledge, this is the first time a ConvLSTM is used for multitemporal crop recognition in a sub-tropical region.

The remainder of this paper is organized as follows: Section II explains the concepts of LSTM, ConvLSTM and FCN, while describing the assessed methods for crop recognition. In section III, the study areas and the

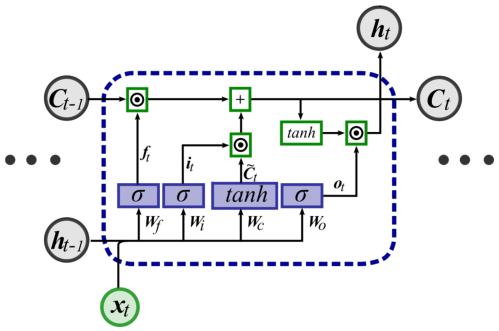


**Figure 1: LSTM structure diagram (Taken from [8])**

experimental protocol are described. Then, the experimental results are discussed in Section IV and, finally, conclusions are summarized in Section V.

## 2. METHODS

This section describes the methods based on LSTM and FCN as well as the traditional image stacking (IS) approach used in this analysis as a baseline. We briefly address the main concepts of LSTM, ConvLSTM and FCN models. For further details about deep learning networks, including the basics of CNNs and RNNs, we refer to [13].

### 2.1. Image stacking (IS)

This method consists of computing hand-crafted features for each image in the multitemporal sequence and stacking them all together, to obtain a feature vector for each pixel that comprises the whole data sequence at that location. The resulting representation is then used to train a classifier that assigns a class label to each pixel.

### 2.2. LSTM for patch classification (LSTM-PC)

Recurrent Neural Networks (RNN) are a type of neural network designed for processing sequential data. These type of models are considered the state-of-the-art in temporal modeling tasks [14]. In particular, LSTMs are a special type of RNN that are capable of modeling both long and short term time dependencies. The main improvement in relation to traditional RNNs is a memory cell $C_t$, which can be accessed, written and cleared by trainable gates (see Figure 1). Specifically, the model uses an information gate $i_t$ to select which information is added to the cell, a forget gate $f_t$ to discard useless previous knowledge and an output gate $o_t$ to decide whether the cell contents will be propagated to the final state $h_t$.

Inspired by the RNN model presented in [8], we designed a LSTM-based architecture for patch classification (LSTM-PC). Each pixel at a given date was represented by a $w \times w \times c$ dimensional feature vector that results from flattening the $w$ by $w$ by $c$ patch centered at that pixel, where $w$ refers to the spatial dimensions and $c$ stands for the number of polarizations. First, the sequence of feature vectors representing a pixel location along all dates in the sequence passes through a basic LSTM cell. Then its last output is gathered and passed to an intermediate Fully Connected (FC) layer followed by a softmax layer. In each case, the predicted class is assigned to the input's central pixel and the inferred image is constructed by concatenating spatially the classification results.

### 2.3. ConvLSTM for patch classification (ConvLSTM-PC)

LSTM's major drawback in handling spatial data is the usage of FC layers for its input-to-state and state-to-state transitions, which do not take spatial context into account. To overcome this limitation, a ConvLSTM cell takes the original LSTM (Figure 1) and replaces the input $x_t$, hidden state $h_t$ and cell output $C_t$ with 3D tensors whose first two dimensions are spatial dimensions (rows and columns), as opposed to feature vectors from LSTM [10].

Similarly to LSTM-PC, each pixel from a given date is represented as an image patch of dimensions $w \times w \times c$ which is centered around that pixel. At first, the sequence of image patches representing a pixel location is passed through a ConvLSTM cell. Then the sequence's last image is selected and *max. pooling* is applied. The result is flattened and taken to the following FC and softmax layers. As in the previous model, inference is achieved with a spatial concatenation from the predicted values.

### 2.4. FCN for patch labeling (FCN-PL)

Typical CNNs contain fully connected (FC) layers that don't consider the spatial information, producing non-spatial outputs. FCN removes the final classification layer from the CNN and converts all the fully connected layers into convolutional ones. In this way, the final output becomes a classification map with spatial dimensions.

A fully Convolutional DenseNet [15] implements a downsampling path, which extracts coarse semantic features, followed by an upsampling path responsible for recovering the input spatial resolution in the final output (Figure 2). In this architecture, Dense Blocks (*DB*) are of a sequence of convolutional layers with multiple bypassing connections among them. Transition Down (*TD*) blocks are composed of a convolution and a downsampling operation, while a Transition Up (*TU*) block performs an upsampling operation, typically a transposed convolution. Skip connections are used between downsampling and upsampling stages.

The input for FCN-PL the stack of spatially corresponding patches through all dates of the multi-temporal sequence, resulting in a tensor of size $(w, w, c \times T)$, being $T$ the sequence length. The network is called full patch labeling (PL) because the output of the network is the set of labels from the whole patch and not only the central pixel. At inference time, patch-wise outputs are spatially concatenated to form the final classification mosaic.

## 3. EXPERIMENTS

### 3.1. Study Areas

Two study areas with different agricultural practices and crop dynamics were selected for our experiments. The first area is located in the surroundings of Hanover city, in Germany. It has an extension of 1728 $km^2$ and consists of a sequence of

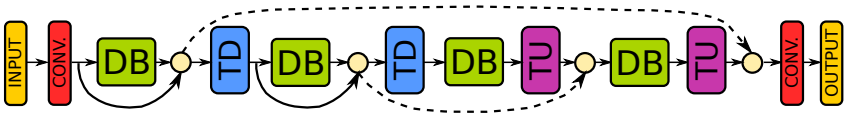


**Figure 2: FCN-PL architecture. Circles represent concatenation. (DB: Dense block, TD: Transition down, TU: Transition up, Conv.: Convolutional layer)**

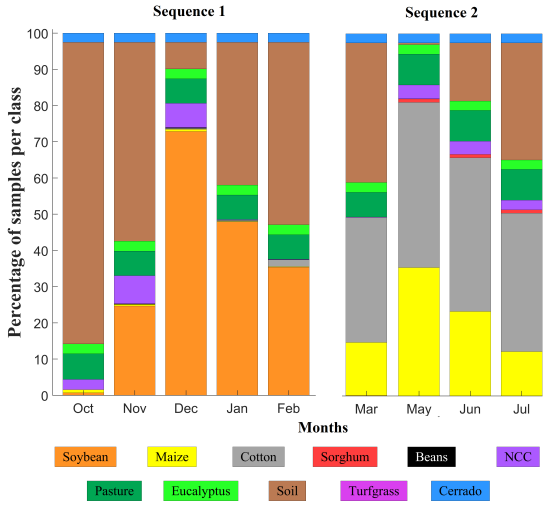


**Figure 3: Class distribution from Campo Verde dataset.**

24 SAR images, dual-polarized from Sentinel-1 satellite taken from October 2014 to October 2016. A key characteristic of this area located in a temperate region is that each parcel has single crop class throughout the entire agricultural year [16].

The second test area is located in Campo Verde municipality, in the state of Mato Grosso, Brazil with an extension of 4782 $km^2$. It consists of a sequence of 14 SAR images from Sentinel-1, dual polarized, acquired from October 2015 to July 2016. In contrast to the first area, it is in a tropical region where crop rotation and different agricultural practices are adopted. Its highly dynamic behaviour makes the modelling of different crops more challenging than in the Hanover site [17]. In our experiments the Campo Verde data set was split in two sequences, each corresponding to a different crop cycle (See Figure 3).

### 3.2. Experimental Protocol

The results presented in the next section refer to the last date of the temporal sequence, using the data of all previous dates. The parcels were randomly separated in training and testing groups, each one having 50% of all pixels. Each feature vector component was normalized to zero-mean and unit variance. The patch sizes were empirically selected as $5 \times 5$ and $15 \times 15$ for Hanover and Campo Verde, respectively.

For the IS model we used a Random Forest with 250 trees and a maximum depth of 25. As handcrafted features for the IS approach we used the correlation, homogeneity, mean and variance extracted from GLCM matrices computed for four directions (0, 45, 90 and 135 degrees).

Different from [8], we chose to work with the original values in each polarization as input for the RNNs, instead of the GLCM features. This decision was based on preliminary experiments results. Additionally, as explained in Section 2, an intermediate Fully Connected layer was added in order to further improve the performance.

For LSTM-PC and ConvLSTM-PC, 100 and 16 filters were used in the recurrent layers respectively. In both cases, the intermediate FC layer was configured with 100 filters for Campo Verde and 300 for Hanover. In ConvLSTM-PC, *max. pooling* was not used for Hanover. All the input patches were extracted with stride 1.

For FCN-PL, input patches of size $8 \times 8$ and $32 \times 32$ were empirically selected for Hanover and for Campo Verde, respectively. The DenseNet was configured with a growth rate of 16, two convolutional layers per block, *average pooling* as downsampling operator and 20% dropout. The DenseNet architecture is described in Table 1, where $w$ is the patch width/height for each database. This model was trained with non-overlapping patches.

Data augmentation was applied to minority classes through rotation, and horizontal and vertical flip. In Campo Verde and Hanover, respectively, 500 and 300 samples per class were used for FCN-PL, while 50000 and 30000 samples per class were used for IS, LSTM-PC and ConvLSTM-PC.

Early stopping regularization was adopted for training. Adam optimizer with learning rate of 0.001 and mini-batches of size 128 were used for the recurrent networks. Adagrad with 0.01 learning rate and mini-batches of size 32 were used for FCN-PL.

**Table 1: FCN-PL architecture.**

| Layer | Output Dimensions | #Filters |
|---|---|---|
| **Input** | $w \times w$ | 48 |
| **DB (2 layers)** | $w \times w$ | 80 |
| **Downsampling** | $w/2 \times w/2$ | 80 |
| **DB (2 layers)** | $w/2 \times w/2$ | 112 |
| **Downsampling** | $w/4 \times w/4$ | 112 |
| **DB (2 layers)** | $w/4 \times w/4$ | 32 |
| **Upsampling** | $w/2 \times w/2$ | 144 |
| **DB (2 layers)** | $w/2 \times w/2$ | 32 |
| **Upsampling** | $w \times w$ | 112 |
| **Conv.** | $w \times w$ | $\#classes$ |

$w$ stands for the input patch width/height: 8 for Hanover and 32 for Campo Verde.

## 4. RESULTS

Results are shown in Table 2, in terms of Overall Accuracy (OA) and Average class Accuracy (AA). Values in bold correspond to the best accuracies for each dataset. The baseline model (IS) achieved significantly better results than the ones reported in [12] for Campo Verde dataset. This can be explained by the use of different patch sizes. However, even with these improvements, the deep learning variants outperformed the baseline in both metrics for both datasets.

**Table 2: Results from both datasets in terms of Overall Accuracy (OA) and Average class Accuracy (AA)**

| Dataset | Campo Verde | | | | Hanover | |
|---|---|---|---|---|---|---|
| | Sequence 1 | | Sequence 2 | | | |
| Layer | OA | AA | OA | AA | OA | AA |
| FCN-PL | **81** | 75.6 | **73.9** | **69** | 91.9 | 88.5 |
| ConvLSTM-PC | 80.5 | **75.9** | 70.4 | 66.4 | **93.7** | **90.2** |
| LSTM-PC | 80.1 | 74 | 72.2 | 69.2 | 91.9 | 85.9 |
| IS | 79.1 | 68.1 | 71.1 | 65.9 | 86.1 | 77.4 |

By and large, the FCN-PL architecture delivered the best accuracies among the tested approaches for Campo Verde dataset, while the ConvLSTM-PC attained the highest scores for Hanover. Parcels are larger in Campo Verde than in Hannover. Thus, spatial context tends to be more important in Campo Verde, which might be beneficial for FCN-PL. On the other hand, the number of temporal images in Hanover dataset is more than three times larger than in the Campo Verde dataset. This might be an indication that this approach tends perform better as the sequence length increases.

Both recurrent networks presented similar performance values both in terms of OA. As for AA, the results do not allow identifying any clear superiority between them. Notice that ConvLSTM-PC presents the highest performance for Campo Verde sequence 1 and for Hanover, but is outperformed by LSTM-PC for Campo Verde sequence 2.

Finally, both LSTM-PC and ConvLSTM-PC presented better results than the ones reported in [8] for Campo Verde. This might also be due to architecture modifications, such as the use of larger patches, and to an additional FC layer.

## 5. CONCLUSIONS

In this work, some of the most successful deep learning architectures for multi-temporal crop recognition were implemented; tailored to the specific dataset requirements; and compared. Their performance was assessed over two very different datasets.

One of the main contributions of this work is the classification improvement for IS, LSTM-PC and ConvLSTM-PC achieved by finding their most appropriate parameter values in each study area. Results indicate that the abilities to represent spatial semantics from FCN-PC might be better harnessed in areas with larger crop tile sizes like the ones from Campo Verde. Likewise, the spatio-temporal modeling properties from ConvLSTM might be more relevant for datasets with larger time sequences.

Future works will focus on combining both architectures into a fully convolutional recurrent network, and the use of other FCN and RNN structures.

## ACKNOWLEDGEMENTS

The authors acknowledge the funding provided by CAPES and CNPq.

## 6. REFERENCES

[1] LEITE, P. B. C. et al. Hidden markov models for crop recognition in remote sensing image sequences. *Pattern Recognition Letters*, Elsevier, v. 32, n. 1, p. 19–26, 2011.

[2] THENKABAIL, P. S. *Land resources monitoring, modeling, and mapping with remote sensing*. [S.l.]: CRC Press, 2015.

[3] LOHMANN, P. et al. Multi-temporal classification for crop discrimination using terrasar–x spotlight images. *Proceedings IntArchPhRS*, v. 38, 2008.

[4] ACHANCCARAY, P. et al. Spatial-temporal conditional random field based model for crop recognition in tropical regions. In: IEEE. *Geoscience and Remote Sensing Symposium (IGARSS), 2017 IEEE International*. [S.l.], 2017. p. 3007–3010.

[5] LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436, 2015.

[6] AUDEBERT, N. et al. *Deep Learning for Remote Sensing*. [S.l.].

[7] NDIKUMANA, E. et al. Deep recurrent neural network for agricultural classification using multitemporal sar sentinel-1 for camargue, france. *Remote Sensing*, v. 10, n. 8, 2018. ISSN 2072-4292. Avaliable at: <http://www.mdpi.com/2072-4292/10/8/1217>.

[8] BERMUDEZ, J.; FEITOSA, R. Q.; HAPP, P. An hybrid recurrent convolutional neural network for crop type recognition based on multitemporal sar image sequences. In: IEEE. *Geoscience and Remote Sensing Symposium (IGARSS), 2018 IEEE International*. [S.l.], 2018.

[9] RUSSWURM, M.; KÖRNER, M. Multi-temporal land cover classification with sequential recurrent encoders. *ISPRS International Journal of Geo-Information*, Multidisciplinary Digital Publishing Institute, v. 7, n. 4, p. 129, 2018.

[10] XINGJIAN, S. et al. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2015. p. 802–810.

[11] CUÉ, L.; HAPP, P.; FEITOSA, R. Q. Dense fully convolutional networks for crop recognition from multitemporal sar image sequences. In: IEEE. *Geoscience and Remote Sensing Symposium (IGARSS), 2018 IEEE International*. [S.l.], 2018.

[12] CASTRO, J. D. B. et al. A comparative analysis of deep learning techniques for sub-tropical crop types recognition from multitemporal optical/sar image sequences. In: IEEE. *Graphics, Patterns and Images (SIBGRAPI), 2017 30th SIBGRAPI Conference on*. [S.l.], 2017. p. 382–389.

[13] GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <http://www.deeplearningbook.org>.

[14] MA, C.-Y. et al. Ts-lstm and temporal-inception: Exploiting spatiotemporal dynamics for activity recognition. *arXiv preprint arXiv:1703.10667*, 2017.

[15] JÉGOU, S. et al. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In: IEEE. *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*. [S.l.], 2017. p. 1175–1183.

[16] BARGIEL, D. A new method for crop classification combining time series of radar images and crop phenology information. *Remote Sensing of Environment*, Elsevier, v. 198, p. 369–383, 2017.

[17] SANCHES, I. D. et al. Campo verde database: Seeking to improve agricultural remote sensing of tropical areas. *IEEE Geoscience and Remote Sensing Letters*, IEEE, v. 15, n. 3, p. 369–373, 2018.