

# USING TIME SERIES IMAGE DATA TO IMPROVE THE GENERALIZATION CAPABILITIES OF A CNN - THE EXAMPLE OF DEFORESTATION DETECTION WITH SENTINEL-2

M. X. Ortega<sup>1,2\*</sup>, D. Wittich<sup>2</sup>, F. Rottensteiner<sup>2</sup>, C. Heipke<sup>2</sup>, R. Q. Feitosa<sup>1</sup>

<sup>1</sup> Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Brazil  
mortega@aluno.puc-rio.br, raul@ele.puc-rio.br

<sup>2</sup> Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Germany  
(wittich, rottensteiner, heipke)@ipi.uni-hannover.de

**KEY WORDS:** Deforestation detection, Convolutional neural networks, Time series, Transferability.

## ABSTRACT:

Deforestation is considered one of the main causes of global warming and biodiversity reduction. Therefore, early detection of deforestation processes is of paramount importance to preserve environmental resources. Currently, there is plenty of research focused on detecting deforestation from satellite imagery using Convolutional Neural Networks (CNNs). Although these works yield remarkable results, most of them employ pairs of images and detect changes which occurred between the two image acquisition epochs only. Furthermore, these models tend to produce poor results when applied to new data in real-world scenarios. In this regard, an interesting research topic deals with the generalization capacity of the classifiers. CNN-based approaches combined with time series data can be a suitable framework to obtain classifiers that generalize better to new data. Image time series contain complementary information, representing different imaging conditions over time. This work addresses the transferability for detecting deforestation in different areas of the Amazon region, using Sentinel-2 time series and reference maps from PRODES project, which are not required to be synchronized. The results indicate that the classifier with time series data brings a substantial improvement in accuracy by taking advantage of the temporal information.

## 1. INTRODUCTION

Over the past decades, major technological advances have given access to massive amounts of remote sensing (RS) data. This availability has allowed to develop various applications that are essential for understanding environmental processes, e.g. the identification of risks, the analysis of urban development and deforestation (Gomes et al., 2020; Tasar et al., 2019). The application considered in this paper is bi-temporal deforestation detection in tropical rain forests, e.g. those in the Brazilian Amazon region, which is particularly important because deforestation is one of the major causes of global warming (Kpienbaareh et al., 2022). The goal is to analyse a pair of co-registered satellite images acquired at different points in time in order to predict at pixel level whether deforestation occurred between the two dates or not, which can be seen as a special case of pixel-wise classification. For the Brazilian Amazon region, every year the Brazilian National Institute for Space Research (INPE) performs a manual analysis to detect deforestation activities in the context of the Program for Deforestation Monitoring in the Brazilian Legal Amazon (PRODES)<sup>1</sup>. As this manual process is tedious and time-consuming, it would be advantageous to automate it.

Convolutional Neural Networks (CNNs) have brought about a substantial improvement in various classification tasks. In order to train a CNN that works well on new images, one requires a large amount of labelled training data that is representative for the task. For the application addressed in this paper, there are two aspects which may make it difficult to meet this

requirement: the location of a scene and the acquisition dates of the images. This is because forests and deforested areas look different in different geographic regions and because the appearance also changes due to seasonal or atmospheric effects. Thus, in order to train a classifier that generalizes well to new data, one requires training data from a variety of regions and acquired at multiple points in time.

For forest monitoring, such data is available in principle: the reference for deforestation generated on a yearly basis in the context of PRODES is freely available for the entire Brazilian rain forest, and satellite image time series with an even higher temporal resolution are available, often as open data. However, using such data for training poses several problems. Firstly, the acquisition dates of the satellite images typically do not correspond to the dates for which the manual annotations are available. Secondly, even if the satellite images and the reference labels are synchronised, using only images corresponding to the dates for which information about a deforestation event is available in the reference may not be optimal, because it would prevent the use of other images that might otherwise increase the pool of training data under the assumption that the change process can be modelled. Due to these problems, it is a common strategy to use data from a subset of a single image pair that is coarsely synchronised with the reference to train a classifier before applying the latter to the remaining areas of the same image pair. Although this yields a high performance, such classifiers often result in poor results when applied to another image pair acquired at a different time or showing a different location (Vega et al., 2021).

Another important aspect to consider is that there may or may not have been different regrowth and deforestation cycles in the investigated scene. However, only primary deforestation events

\* Corresponding author

<sup>1</sup> <http://terrabrasilis.dpi.inpe.br/map/deforestation>  
(accessed 27/06/2023).

are recorded in PRODES. This means that areas which were already labelled as being affected by deforestation in the past are automatically considered not to be affected by (primary) deforestation any more (INPE, 2021). Practically, those areas are ignored in the manual analysis and, thus, information about regrowth is not contained in the database, so that situations in which a second deforestation event occurs after regrowth are not represented in the training data. In addition, there are barely any areas in the reference for the class *No Deforestation* where there is no forest at both, the earlier and the later date, as those areas were usually affected by deforestation in the past and, thus, are ignored in the manual labelling process. The lack of such training samples does not allow to train a classifier that detects arbitrary deforestation events.

In this work, we propose a strategy for training a CNN for bi-temporal deforestation detection in the Amazon region that leverages the available data described above with the aim that the classifier generalises better to unseen data. The proposed training strategy relies on the availability of time series of satellite and (potentially incomplete) reference label images, which need not be synchronized. The core idea is to use knowledge about the class transitions in the time series of reference labels to infer a valid set of reference labels given an arbitrary image pair. We aim to train a classifier to detect arbitrary deforestation events, i.e. areas where there is no forest at a specific date but where there was forest at an earlier date. We assume that training on many different image pairs from the time series helps the classifier to generalize better to new image pairs, thus improving the prospects of an automatic workflow for the task at hand. Furthermore, using the proposed strategy, we obtain a classifier that is capable of detecting not only primary deforestation, but also secondary deforestation occurring after regrowth. In our work we consider small regions in the Amazon region, which, however, are big enough, to enable us to investigate how the proposed strategy influences the classification performance of a CNN trained on a time series of one such region when it is applied to another region. Whereas we focus on the application of bi-temporal deforestation detection, in principle the proposed training strategy is transferable to similar scenarios in which existing but imperfect label data should be used to train a classifier to improve its generalization capacity.

The main contributions of this work are as follows: (1) We propose a strategy to leverage an existing database with a series of reference label maps and a time series of satellite images to train a CNN so that the latter generalizes well to unseen data. The labels in the database and the images need not be synchronized. (2) We propose and compare different variants of our method, relying on different ways for generating training labels from the available information. (3) We also show that the CNNs trained using the proposed strategy achieve a much higher classification performance compared to the standard procedure of training on a single image pair synchronised with the reference. (4) In our experiments, we evaluate by how much the performance of the classifier is improved when considering training and test data from different areas and / or epochs, indicating the improvement of the generalization capability. (5) Furthermore, we show that using a slight modification of the way in which the training labels are obtained, we can train a classifier capable of predicting secondary deforestation events even though those are not explicitly contained in the database. However, this contribution is evaluated only qualitatively, as we do not have a reference for secondary deforestation events.

## 2. RELATED WORK

Addressing bi-temporal deforestation detection, several Fully Convolutional Networks (FCN) were assessed in (Torres et al., 2021). Besides comparing different architectures, the authors evaluated the performance of classifiers trained on image pairs from different satellite missions, namely Landsat-8 and Sentinel-2, which have a different spatial resolution. The experiments were carried out in a region of the Amazon forest, using reference labels from the PRODES dataset. More recently, in (Andrade et al., 2022) a variant of the DeepLabv3+ architecture was suggested for deforestation detection using Sentinel-2 image pairs. This method was also evaluated on a region within the Amazon forest. The results of DeepLabv3+ were compared with two patch-wise semantic segmentation methods, and it outperformed the baselines. Although all these works achieved satisfactory classification performance, they are focused on bi-temporal image pairs, considering only test samples which come from the same image pair that was used for training. As the Amazon region is characterized by a complex forest cover, when these models are applied to other datasets with temporal and spatial changes, there is a significant drop in performance in the classification results. For instance, in (Vega et al., 2021), the classifiers achieved only a poor accuracy when they were trained on one dataset and evaluated on another one. In many such cases, this drop in accuracy was more than 50% in terms of F1-score compared to results achieved when training and testing on subsets of the same dataset.

In this regard, a related group of methods which are interesting for the application of bi-temporal deforestation detection are those from Transfer Learning (TL). Here, the data is assumed to be available in different domains which are different but related, e.g. data from different epochs or from different geographical locations. The goal of TL is to leverage the data available in all domains to achieve a good performance in the domain which is to be classified. In the default setting, one considers a source domain (SD) for which plenty of training data is available and a target domain (TD) which is to be classified but for which an insufficient amount of training labels is available. It is also assumed that training in the SD only does not yield satisfactory performance in the TD due to the so called domain gap (Soto et al., 2022; Tasar et al., 2020).

One setting in TL is unsupervised domain adaptation (UDA). It is particularly interesting, because one does not require any training labels in the target domain. Such techniques address the domain gap problem by transferring information from a labeled SD to an unlabeled TD. In particular, for deforestation detection, different methods based on domain adaptation have been proposed (Soto et al., 2022; Noa et al., 2021; Vega et al., 2021). The authors analysed the domain gap between several regions in the Amazon with specific time and geographical locations. Their analysis started by evaluating the performance of the classifiers in the naive scheme, where the classifier is trained on the SD and tested on different TD; the results showed poor performance in all scenarios. In this regard, the UDA methods are proposed as a solution to mitigate the domain gap. Although the authors reported encouraging results, when compared to the performance of a classifier trained on the TD, there is still a drop in performance, motivating the exploration of other strategies to bridge the domain gap.

In this work, we aim to prevent a potential domain gap by leveraging additional time series data for training. In

the literature, there exist some works which consider time series in combination with CNNs to analyze forest dynamics and other environmental processes (Carrillo-Niquete et al., 2022). In (Matosak et al., 2022), a CNN-based model using a combination of recurrent neural networks (RNN) and U-Net is proposed to monitor deforestation and forest degradation in the Brazilian Cerrado region. In that work, Landsat-8 and Sentinel-2 time series with labels from PRODES were used, and the method was compared in three different scenarios, starting with training and evaluating on a labeled dataset, considered as SD. The authors also evaluated a classifier trained on a SD and tested on TD from different epochs and geographical locations. In the first scenario, satisfactory results were achieved. However, in the other two scenarios, when the model is evaluated in different TD, a drop in performance is observed, especially when the TD differs in geographical location. Likewise, some variants of CNN and RNN with Landsat-8 time series were evaluated in (Masolele et al., 2021). The authors also assessed a hybrid model to combine spatio-temporal information of the data. The target application was also deforestation detection in tropical regions. Even though these works, taking advantage of the spatio-temporal information contained in the time series, reported satisfactory results, they require synchronized images and reference data, which can be problematic in real-world applications.

In this work, we propose a training strategy to bridge the domain gap for deforestation detection. This strategy relies on the availability of a satellite image time series and a series of reference label maps, which need not be synchronized. Instead, by using a set of rules to infer valid labels from a reference label series for an arbitrary epoch, it is possible to train a classifier that generalises well to unseen data and it is also possible to detect secondary deforestation events even though they are not contained in the reference.

### 3. METHODOLOGY

We start this section with a formal introduction of bi-temporal deforestation detection based on satellite imagery. The goal of deforestation detection is to automatically determine at pixel level whether deforestation has occurred between an earlier date  $t_e$  and a later date  $t_l$  based on satellite images. The data to be classified consist of a pair of co-registered images that show the same region, the earlier image  $X_e$  acquired at  $t_e$  and the later image  $X_l$  acquired at  $t_l$ . The desired output is a binary label map that indicates for each pixel whether deforestation has occurred between  $t_e$  and  $t_l$  or not. We propose a strategy that uses asynchronous time series of images and label data to train a classifier that performs well on an unseen image pair.

For training, let us consider a satellite image time series  $\mathcal{X} = \{X_{I_1}, X_{I_2}, \dots, X_{I_i}, \dots, X_{I_N}\}$ , where  $N$  is the number of time steps of  $\mathcal{X}$ , and  $X_{I_i}$  represents an image at time step  $t_{I_i}$ ; the notation  $I_i$  indicates that the index  $i$  refers to the image time series. Furthermore, we assume a label map time series  $\mathcal{Y} = \{Y_{L_1}, Y_{L_2}, \dots, Y_{L_j}, \dots, Y_{L_M}\}$ , where  $M$  is the number of time steps of  $\mathcal{Y}$ , and  $Y_{L_j}$  represents a label map at time step  $t_{L_j}$ , and refers to deforestation that occurred between  $t_{L_{j-1}}$  and  $t_{L_j}$ ; here, the notation  $L_j$  indicates that the index  $j$  refers to the label map time series. Note that identical indices do not imply identical time stamps if the indices refer to different time series, i.e.  $i = j$  does not imply  $t_{I_i} = t_{L_j}$ . Typically, the time series  $\mathcal{X}$  will contain more images than  $\mathcal{Y}$ . In this paper, the label maps in  $\mathcal{Y}$  are generated based on the information available

in PRODES, i.e. information about recently deforested areas published once a year. This information is derived from Landsat-8, Sentinel-2 and CBERS data acquired in the dry season, when the cloud coverage is minimal.

In principle, the task of bi-temporal deforestation detection requires a reference in which two classes are differentiated: *Deforestation (DF)* and *No Deforestation (NDF)*, where this information is related to what happened between the epochs  $t_e$  and  $t_l$  at which the images were acquired. In particular, PRODES only contains information about primary deforestation, i.e. areas that were labelled as deforested at some point in time in the past are ignored in the yearly manual labelling process. As there may or may not have happened another regrowth and deforestation cycle, those areas cannot be used to train a model for deforestation detection as the reference labels are unknown. To deal with this problem, a third label, *Past Deforestation (PD)*, is assigned to areas in a label map  $Y_{L_j}$  that were labelled as *DF* at any point in time earlier than  $t_{L_{j-1}}$ . Such areas are supposed not to carry any information for bi-temporal deforestation classification between epochs  $t_{L_{j-1}}$  and  $t_{L_j}$ , and they are commonly disregarded in the training procedure (INPE, 2021).

Our goal is to use the two time series  $\mathcal{X}$  and  $\mathcal{Y}$  to train a classifier  $C$  that predicts the label map  $\hat{Y}$  differentiating the classes *DF* and *NDF* for an arbitrary unseen image pair  $\tilde{X}^T = \{X_e^T, X_l^T\}$ , where the superscript  $T$  indicates that the image pair is a test pair not contained in the training set. In our strategy, the time series  $\mathcal{X}$  and  $\mathcal{Y}$  to be used for training are not required to be synchronised. This is different from the common approach (Wang et al., 2023; De Bem et al., 2020; Torres et al., 2021), which is to choose a pair of training images  $X_{I_l}$  and  $X_{I_e}$  and label maps  $Y_{L_j}$  and  $Y_{L_{j-1}}$  such that the acquisition date  $t_{I_l}$  of the later training image  $X_{I_l}$  is as close as possible to the date  $t_l = t_{L_j}$  of the label map  $Y_{L_j}$  and the acquisition date  $t_{I_e}$  of the earlier image is as close as possible to the date  $t_e = t_{L_{j-1}}$  of  $Y_{L_{j-1}}$ . This commonly used approach has two disadvantages. Firstly, the resultant classifier cannot be expected to be well transferable to new image pairs, because only a single image pair is used for training. Secondly, there are barely samples for the subcategory of the class *NDF* in which there is no forest in  $t_e$  and no forest in  $t_l$ , either, because this is a case that usually only occurs in the areas marked as *Past Deforestation*, which are not to be used for training. Consequently, in such areas, the classification performance is poor. This is why usually the pixels marked as *Past Deforestation* in the label maps are not considered in the evaluation, e.g. (Torres et al., 2021).

We propose an alternative training strategy (cf. Figure 1). We start by randomly sampling overlapping patches in arbitrary image pairs from the time series  $\mathcal{X}$  and generate the required training labels online using information about the time  $t_d$  at which the area related to a pixel was identified as being deforested. This information could be derived from the (non-synchronous) time series of label maps  $\mathcal{Y}$  by identifying the time  $t_{L_j}$  of the label map  $Y_{L_j}$  in which this pixel is marked as belonging to class *DF*. In the particular case of PRODES, the database does not only contain the label images. For every area affected by deforestation, it also contains the acquisition date of the image in which the corresponding *DF* polygon was digitized. In other words, for every *DF* pixel in any of the images in the time series  $\mathcal{Y}$ , PRODES also contains the time stamp  $t_{DF}$  of the image in which that pixel was found to be deforested. In this work, we use this attribute to define  $t_d = t_{DF}$  and, consequently, the label maps describing the

deforestation between  $t_e$  and  $t_l$ . The way in which this is done is the core of the proposed strategy and is described in Section 3.1. The resultant training data are used to train a CNN to predict the pixel-wise class scores for  $DF$  and  $NDF$ . The CNN structure is presented in Section 3.2, whereas Section 3.3 describes the training procedure.

### 3.1 Generation of training samples

In this work, we introduce three different procedures to create the training samples. All of them create a label map  $Y$  by computing the label  $y$  for each pixel using the time  $t_d$  at which the deforestation occurred, defined in the way described above; pixels that were never deforested (i.e., do not appear as belonging to class  $DF$  in any of the label images in the time series  $\mathcal{Y}$ ) are marked by a special value  $t_d = NEVER$ .

The first procedure serves as our baseline and corresponds to the common strategy in which two fixed dates are used for training a classifier; it was also used for bi-temporal deforestation detection in (Torres et al., 2021; Andrade et al., 2022). Here, we select a time  $t_d$  for areas marked as deforested in a label map  $Y_{L_j}$  and then search for a pair of images  $X_{I_l}$  and  $X_{I_e}$  from  $\mathcal{X}$ , acquired at times  $t_l = t_{I_l}$  and  $t_e = t_{I_e}$ , such that  $t_l$  is the date closest to  $t_d$  and  $t_e$  is selected to correspond most closely to the date  $t_{L_{j-1}}$  (i.e., given the strategy used in PRODES, the one of the label map produced for the year before  $t_{L_j}$ ). Then, a new label map  $Y$  representing the deforestation that occurred between  $t_e$  and  $t_l$  is generated. This is done based on a set  $R1$  of rules to determine the label map  $Y$  to be used for training by computing the label  $y$  for each pixel:

$$R1 : y = \begin{cases} DF & \text{if } t_e \leq t_d \leq t_l, \\ NDF & \text{if } (t_d > t_l) \vee (t_d = NEVER), \\ UK & t_d < t_e. \end{cases} \quad (1)$$

The last row in equ. 1 corresponds to what was introduced as *Past Deforestation* above. For such regions it is unknown ( $UK$ ) whether the correct class is  $DF$  or  $NDF$ .

For the other two training strategies, we use all the available data, i.e. we do not restrict the selection of  $t_e$  and  $t_l$  by requiring them to correspond to consecutive years. To construct a training sample, first, a random pair of images  $X_{I_l}$  and  $X_{I_e}$  is sampled from  $\mathcal{X}$ , and again we have  $t_l = t_{I_l}$  and  $t_e = t_{I_e}$ . Here, the label  $y$  for each pixel is determined based on a set  $R2$  of rules:

$$R2 : y = \begin{cases} DF & \text{if } t_e + \rho \leq t_d \leq t_l, \\ NDF & \text{if } (t_d > t_l) \vee (t_d = NEVER), \\ UK & \text{otherwise.} \end{cases} \quad (2)$$

In eq. 2,  $\rho$  is a hyper-parameter corresponding to a buffer time-span. The rationale behind  $R2$  is that for an arbitrary image pair, we assume deforestation to have happened between the two dates if the deforestation was recorded before  $t_l$  and after  $t_e$ . However, if it was recorded only shortly after  $t_e$ , the actual deforestation might have happened before  $t_e$  and, thus, not between  $t_e$  and  $t_l$ . This is the reason why we add the buffer  $\rho$ , which should roughly correspond to the frequency in which the manual reference is generated.

The third variant considers additional cases, using knowledge about the class transitions to define a set  $R3$  of rules for determining the reference label  $y$  for every pixel:

$$R3 : y = \begin{cases} DF & \text{if } t_e + \rho \leq t_d \leq t_l, \\ NDF & \text{if } (t_d = NEVER) \vee \\ & (t_d > t_l + \rho_a) \vee \\ & (t_e - \rho_r < t_d < t_e), \\ UK & \text{otherwise.} \end{cases} \quad (3)$$

The main difference to the rule set  $R2$  (eq. 2) is in the definition of class  $NDF$ , which is assigned to  $y$  if one of three constraints is fulfilled. The first one is considered in  $R2$ , too. The second one considers the case in which deforestation that happened after  $t_l + \rho_a$ , from which we infer that the deforestation event did not happen in between  $t_e$  and  $t_l$ . Again, a time buffer is required, because if the deforestation was detected only shortly after  $t_l$ , the event may or may not fall into the interval between  $t_e$  and  $t_l$ . The third constraint is very interesting in the context of the addressed application. Here, recent deforestation, i.e. deforestation that was recorded before  $t_e$ , but not earlier than  $t_e - \rho_r$ , is considered as *No Deforestation* with respect to the interval between  $t_e$  and  $t_l$ . As noted earlier, areas which were recorded as deforestation in the past could, in principle, belong to the class  $DF$  or to  $NDF$ , because another regrowth cycle may have happened. However, if an area was recorded as deforestation only shortly before  $t_e$ , we assume that no forest has regrown in the meantime and thus, the area is considered as  $NDF$  with respect to the interval  $t_e$  to  $t_l$ . This is beneficial when training the model, because it corresponds to a  $NDF$  case in which there is neither a forest in  $t_e$  nor in  $t_l$ . This case is barely represented in the labels obtained when using the other rules, where  $NDF$  corresponds to the case in which there is forest in  $t_e$  and also in  $t_l$  in most cases.

Based on these considerations, we assume that classifiers trained using the rules  $R1$  or  $R2$  will tend to make wrong predictions for  $NDF$  areas where there is no forest in both images. This is why the predictions of these classifiers are only usable in combination with the past deforestation maps, which basically mask out the regions in which there is no forest in the earlier image although deforestation occurred in the past ( $t_d < t_e$ ). On the other hand, we assume that a classifier that was trained using the third set of rules (eq. 3) can correctly predict such regions and, thus, the predictions can be used without knowledge about past deforestation.

It has to be noted that the label maps generated using the rules defined above may contain some errors related to errors in  $t_d$ . These errors are caused by the fact that in areas covered by clouds, deforestation areas might be manually labelled on the basis of images that were not acquired immediately after the deforestation event, but at a later point in time.

### 3.2 Network architecture

In this work, an architecture similar to U-Net (Ronneberger et al., 2015) is used, having a Xception backbone (Chollet, 2017), a CNN architecture for image categorisation that builds upon the concept of residual learning (He et al., 2016). The main modification is the integration of depth-wise separable convolutions into the residual blocks. Chollet (2017) argues that, when using the same amount of parameters, this

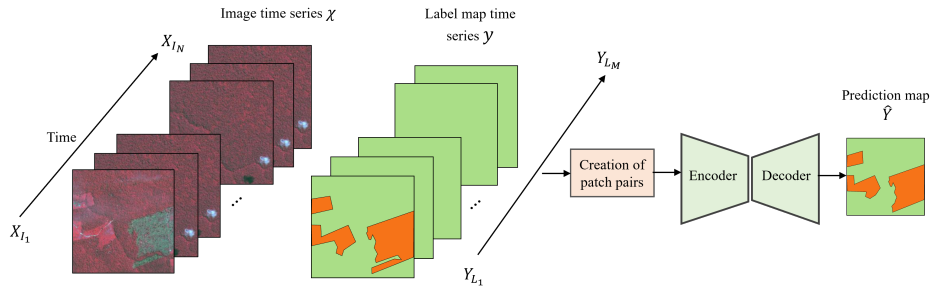


Figure 1. Overview of the suggested method. The training set is created from non-synchronized time series of Sentinel-2 images  $\mathcal{X}$  and reference label maps  $\mathcal{Y}$ .

modification leads to an increased learning capacity compared to using residual blocks with regular convolutions.

Figure 2 shows an overview of the classification network, and Table 1 provides an overview of all layers. The encoder of the network processes input patches with dimension  $p = H \times W$  and  $n_C$  channels, which are passed through the layers of the Xception network (Chollet, 2017) (layers 1-14 in Table 1). In the decoder of the network, nearest neighbour interpolation is used for upsampling (layers 15-23 in Table 1). All convolutions in the decoder use  $3 \times 3$  kernels and zero padding with  $1 px$ . The exceptions are the last three convolutions (layers 24, 25 and 26), that use  $1 \times 1$  kernels and no padding. For the last convolutional layer, we use the softmax activation function with two outputs, associated to the classes  $DF$  and  $NDF$ . Overall, this network has about 15.5 M parameters and a theoretical receptive field of  $907 \times 907 px$ .

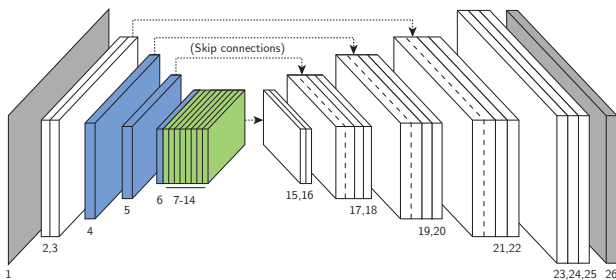


Figure 2. Illustration of the Xception U-Net architecture. The layer numbers correspond to those in Table 1. Blue and green layers correspond to Xception blocks of type A and B, respectively. The dotted lines represent skip connections.

### 3.3 Loss Function and Training Procedure

As this application contains highly imbalanced class distributions, we used the Adaptive Cross-Entropy loss (ACE) for training, proposed in (Wittich and Rottensteiner, 2021). It is a variation of the conventional weighted cross entropy loss, where instead of setting a fixed weight for each class, the weights are adapted during each epoch according to the classification performance of each class  $c$ . The adapted weights  $w_c$  are calculated according to:

$$w_c = (1 - \Delta IoU_c)^\kappa = \left[ 1 - \left( IoU_c - \frac{1}{2} \sum_c IoU_c \right) \right]^\kappa, \quad (4)$$

where  $c \in \{DF, NDF\}$ ,  $IoU_c$  symbolizes the intersection over union for each class,  $\Delta IoU_c$  is the difference between the

	Layer(s)	Layer type	$h, w$	Depth
Encoder	1	Input layer	$p$	$n_C$
	2	StrConv, BN, ReLU	$p/2$	32
	3	Conv, BN, ReLU	$p/2$	64
	4	Xception block A	$p/4$	128
	5	Xception block A	$p/8$	256
	6	Xception block A	$p/16$	728
	7-14	$8 \times$ Xception block B	$p/16$	728
Decoder	15, 16	$2 \times \{\text{Conv, BN, ReLU}\}$	$p/8$	128
	17	Upsample, Concat(4)	$p/4$	256
	18, 19	$2 \times \{\text{Conv, BN, ReLU}\}$	$p/4$	64
	20	Upsample, Concat(3)	$p/2$	128
	21,22	$2 \times \{\text{Conv, BN, ReLU}\}$	$p/2$	32
	23	Upsample	$p$	32
	24,25	$2 \times \{\text{Conv, BN, ReLU}\}$	$p$	16
	26	Conv, softmax	$p$	2

Table 1. Layers of the architecture of the network used in this paper. Conv: Convolutional layer. StrConv: Strided convolutional layer. BN: 2D batch normalization; ReLU: Rectified linear unit. Concat( $L_X$ ): Depth-wise concatenation of the output of layer  $L_X$  and the current layer.  $h, w$ , depth: Output dimensions.  $p$  is the predefined patch size. For a detailed explanation of the Xception blocks, cf. (Chollet, 2017).

class-wise  $IoU$  score and the mean  $IoU$  of classes  $DF$  and  $NDF$ , and the hyper-parameter  $\kappa$  scales the influence of classes with a lower  $IoU$ . The loss for the images in a batch  $B$  and with height  $H$  and width  $W$  is defined by:

$$\mathcal{L} = -\frac{1}{N_p} \sum_{b=1}^B \sum_{i=1}^H \sum_{j=1}^W \sum_c \bar{y}_b(i, j, c) \cdot \log(r_b^S(i, j, c)) \cdot w_c, \quad (5)$$

where  $b$  is the index of an image in the batch,  $i, j$  are the indices of a pixel in an image, and  $c$  is the class  $DF$  or  $NDF$ .  $N_p = H \cdot W \cdot B$  represents the total number of pixels in a batch. The symbol  $\bar{y}_b(i, j, c)$  indicates whether pixel  $(i, j)$  in the  $b^{th}$  label map belongs to class  $c$ ,  $r_b^S(i, j, c)$  denotes the softmax output for the class with index  $c$  at pixel  $(i, j)$ . We extend this loss formulation by defining  $w_c = 0$  for those pixels for which the reference class is unknown ( $UK$ ; cf. eqs. 1-3), i.e. those pixels do not contribute to the loss.

The Xception backbone is pre-trained on the ImageNet dataset (Deng et al., 2009), except for the first layer. That layer is initialized randomly because ImageNet only consists of RGB-images, whereas in our application the number  $n_C$  of

input channels is different from 3. The parameters of the decoder are also initialized randomly; random initialization is based on (He et al., 2015). In each training iteration, a batch of training samples is constructed following the strategy introduced in Section 3.1. Next, the loss is calculated for the batch according to eq. 5 and the parameters of the network are updated using gradient descent with momentum. Furthermore, to prevent over-fitting, we use early stopping, i.e. training is stopped if the performance of the classifier on the validation set did not improve for 10 epochs.

#### 4. EXPERIMENTS

In this section, we present the experiments conducted to evaluate our method. We start by presenting the dataset used in the experiments. Then, we describe the experimental setup, and finally, we present and discuss the results.

##### 4.1 Dataset

The dataset is composed of three different domains located in the Pará and Rondônia states of the Brazilian Legal Amazon (BLA; cf. Figure 3). These domains were suggested by experts for deforestation monitoring and cover both, areas with and without deforestation. The time series of images and deforestation reference maps are related to the deforestation which occurred between 2017 and 2021. These label maps were downloaded from the PRODES project, which provides an open-access database.

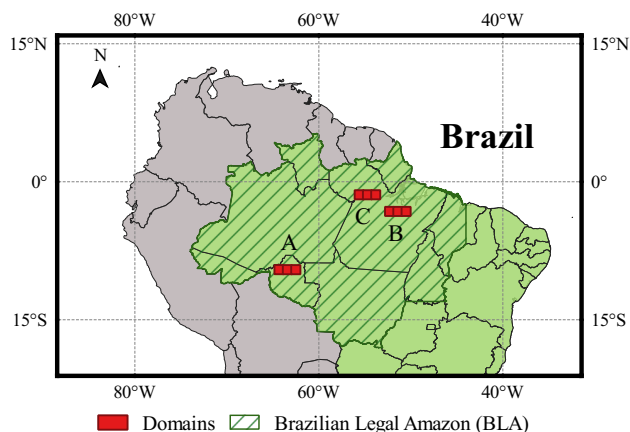


Figure 3. Geographical locations of the datasets. Domain A is located in Rondônia, domains B and C in Pará.

The dataset consists of 100 Sentinel-2 Level-2A images. Each domain is a mosaic of three Sentinel-2 scenes, which were acquired from the Copernicus Open Access Hub<sup>2</sup> provided by the European Space Agency (ESA). Multiple images from different acquisition dates from late May to September are considered, depending on the cloud coverage and data availability (Assis et al., 2019). The images were selected based on the requirement for a small cloud coverage (less than 5%). Table 2 presents detailed information about the acquired images per domain and year, as well as the percentage of deforestation in each scene. All images were processed by ESA’s Sen2Cor v2.9 software<sup>3</sup> to apply bottom-of-atmosphere corrections. For

<sup>2</sup> <https://scihub.copernicus.eu/> (accessed 27/06/2023)

<sup>3</sup> <https://step.esa.int/main/snap-supported-plugins/sen2cor/sen2cor-v2-9> (accessed 27/06/2023)

the classification, the multi-spectral composite of near infrared, red, green and blue was used. These channels are available with a ground sampling distance (GSD) of 10 m.

Figure 4 illustrates some examples of the training set from each domain. Each column represents a different domain (A, B, or C). The first two lines represent the co-registered pair of multi-spectral images (MSI) for the earlier image  $X_e$  and the later image  $X_l$  as RGB composites. The third line shows the reference label maps.

Domain	Sentinel-2 Tile ID	Images per year					Def. [%]	
		2016	2017	2018	2019	2020		2021
A	T20-LLQ	1	2	2	2	2	2	9.3
	T20-LMQ	1	2	3	3	2	2	6.6
	T20-LNQ	1	2	2	2	2	2	4.0
B	T22-MCB	1	2	2	2	2	2	2.6
	T22-MDB	1	2	2	2	2	1	7.6
	T22-MEB	1	2	2	2	2	1	9.3
C	T21-MXU	1	2	2	3	2	2	2.1
	T21-MYU	1	2	2	2	2	2	1.6
	T21-MZU	1	2	2	2	2	2	2.6

Table 2. Number of Sentinel-2 images per year for each domain. The last column (Def.) gives the relative amount of deforestation areas in each tile.

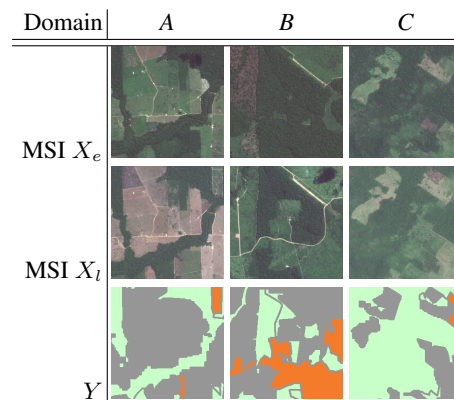


Figure 4. Data samples from the datasets for deforestation detection; side length of the patches is 256 px. MSI: RGB composite of the earlier ( $X_e$ ) and the later multispectral images ( $X_l$ ). Y: Reference label maps derived from PRODES. Colour-codes: *Unknown* (grey), *Deforested* (orange), *Non-deforested* (green)

##### 4.2 Experimental setup

Each domain was divided into 75 tiles, with a distribution of 40%:10%:50% for training, validation and evaluation, respectively. The classifier was trained using image patches with size of  $256 \times 256$  px; patches that do not contain any deforestation were removed from the training set to increase the relative amount of pixels that show deforestation and partially compensate for problems due to the class imbalance.

In pre-processing, the deforestation reference was rasterized with a GSD of 10 m to generate the label maps, also including the time stamp  $t_d$  for every pixel marked as *DF* in a label map. Although only images with low cloud-coverage were queried, there are still some clouds in the images. As a correct prediction is impossible for the areas which are occluded

by the cloud in one or both of the images, such areas are ignored during training and inference. Following the PRODES methodology, two-pixel-wide inner and outer boundaries of polygons identified as *Deforested* in the reference are also ignored. This buffering is done because the manual delineation was partially carried out using images of lower resolution, resulting in a low spatial precision of the outlines of some deforestation areas, and therefore possibly incorrect labels.

The implementation of the Xception U-Net and the pre-trained weights are taken from the *Segmentation Models Pytorch* repository<sup>4</sup> (Iakubovskii, 2019). In addition, during training, the loss function was minimized using Stochastic Gradient Descent with a learning rate of  $2e^{-3}$ , momentum of 0.9, and weight decay of  $1e^{-5}$ . Furthermore, data augmentation was applied by randomly cropping the image patches. Each experiment was run five times using different random initializations. The classification results reported in the next section represent the average of the five runs. We report the F1-scores for the class *Deforestation*, defined as harmonic mean of precision and recall for that class.

For the experiments, we trained different classifiers for each domain (A, B, and C) following the rules described in Section 3.1. For *R2* and *R3*, the time-span parameters  $\rho$ ,  $\rho_a$ , and  $\rho_r$  were defined as one year. The performance of the classifiers was evaluated in two different scenarios. In the first scenario, for training and evaluation, we selected the images acquired at two specific epochs for each domain, i.e.  $t_e = 07/25/2019$  and  $t_l = 08/08/2020$  for domain A,  $t_e = 06/23/2019$  and  $t_l = 06/22/2020$  for domain B, and  $t_e = 07/29/2019$  and  $t_l = 09/06/2020$  for domain C. It roughly represents the real scenario implemented in PRODES, which is related to the deforestation occurring within one year; the results are presented in Section 4.3. In the second scenario, we randomly selected an image pair from the entire time series, which contains images from 05/24/2016 to 10/11/2021, and the corresponding references were defined in the way described in Section 3.1. Note that in this second scenario, the image pair for a patch was selected randomly only once so that in different test runs, the same image pairs were used. The results of the evaluation in this scenario are discussed in Section 4.4. In both scenarios, we analysed the performance of the classifiers applied on the same domains (intra-domain setting) and on different domains (cross-domain setting).

### 4.3 Evaluation on a fixed image pair

For these experiments, we evaluated each classifier trained on domains A, B, and C, and following the rules *R1*, *R2* and *R3* for an image pair acquired at two specific epochs. Table 3 summarizes the classification results in terms of the average F1-scores for the class *Deforestation* (*DF*). As expected, the F1-scores achieved in the intra-domain settings are higher than those in the cross-domain setting: a classifier performs better if trained on data from the same domain as the test data, because then it is more likely for the data in the training and test sets to follow similar distributions. Comparing the performance of the classifiers trained using labels generated according to rule sets *R1* and *R2* in the intra-domain setting, we can notice that the inclusion of time series for training results in a better performance in all cases, demonstrating that the classifiers can use the knowledge about class transition in different epochs to

<sup>4</sup> [https://github.com/qubvel/segmentation\\_models\\_pytorch](https://github.com/qubvel/segmentation_models_pytorch) (accessed 27/06/2023)

gain a more comprehensive view of the appearance of changes over time. The improvement in the intra-class setting is up to 6% in the F1-score for *DF* (area C). Similarly, using the entire time series and defining the training labels according to rule set *R3*, a better performance was achieved compared to *R1*. In this case, additional samples of class *NDF* are included. For area A, where the variant based on *R1* performs best among all areas, *R3* leads to a small decrease in the F1-scores and for area B the increase is smaller than for *R2*, but in the intra-domain setting in area C, the variant based on *R3* performs best, with an increase in the F1-score of 3.7%.

Train on / Rule	Test on 2019-2020		
	A	B	C
A / <i>R1</i>	72.4	58.1	26.3
A / <i>R2</i>	76.2 (+3.8)	68.2 (+10.1)	28.6 (+2.3)
A / <i>R3</i>	71.8 (-0.6)	64.3 (+6.2)	28.0 (+1.7)
B / <i>R1</i>	67.8	68.3	21.1
B / <i>R2</i>	71.7 (+3.9)	74.3 (+6.0)	23.7 (+2.6)
B / <i>R3</i>	68.6 (+0.8)	71.1 (+2.8)	22.3 (+1.2)
C / <i>R1</i>	54.8	43.5	40.9
C / <i>R2</i>	67.0 (+12.2)	69.9 (+26.4)	43.5 (+2.6)
C / <i>R3</i>	68.8 (+14)	70.8 (+27.3)	44.6 (+3.7)

Table 3. Mean F1-scores [%] for the class *DF* achieved in five test runs in the evaluation scenario based on a fixed image pair (2019-2020). The classifiers were trained following the rules defined in Section 3.1. Values in parentheses show the gain in F1-score of the classifiers based on rule sets *R2* and *R3* compared to the baseline (*R1*). The standard deviation of the F1 scores is  $\pm 1.4\%$ .

Regarding the cross-domain results achieved in the variants based on *R1*, there is a drop in performance compared to the intra-domain results. In this case, the classifiers are evaluated in another domain than they were trained on, which may have different distributions due to the different geographic locations and vegetation types. The drop of the F1-scores when only using a single image pair for training (*R1*) can be up to 25% (e.g., testing in domain B after training in domain C). However, when time series data and the additional cases for *NDF* are included (rule sets *R2* and *R3*), better results are achieved, demonstrating that the generalization capability is improved when the information from the time series is exploited. The improvement is most pronounced when training in domain C and evaluating in domain B (+26% for *R2*, +27% for *R3*); however, a small improvement can be observed in all cross-domain settings.

### 4.4 Evaluation on random image pairs

In this setting, we evaluated the classifiers in a more general scenario. Instead of selecting a fixed image pair, an arbitrary random image pair was selected from the entire time series for every patch to be classified, while  $t_e < t_l$  in all cases. For comparison purposes, the random image pairs were the same for all of these experiments. Table 4 summarizes the classification results in terms of average F1-scores for the class *DF*. The results suggest that using rule sets *R2* and *R3* improves the F1-score of the classifier significantly compared to the baseline *R1* in all three domains. Looking at the intra-domain results based on rule set *R1*, a lower performance was achieved than for the evaluation scenario based on fixed dates (cf. Table 3). We believe that this is due to the

fact that in the first scenario, the classifier was trained using samples in the same year (and, thus, in the same images) that were also used for testing. In contrast, in the evaluation using different (randomly selected) image pairs, the distribution of the data may be different compared to the single image pair used in training, thus potentially leading to a decreased performance. On the other hand, when the time series data ( $R2$ ) and the additional cases for the class  $NDF$  ( $R3$ ) are considered in training, the performance of the classifiers is significantly improved compared to the baseline ( $R1$ ), with a margin of up to 22% in the intra-domain setting. This shows the advantage of using time series for training to identify patterns and trends that may not be apparent in a single image pair: with the inclusion of time series data, the classifiers can learn from a larger set of changes occurring over time, possibly characterized by different changes of the appearance in the data, leading to a better performance when classifying unseen data.

Train on / Rule	Test on Random dates		
	A	B	C
A / $R1$	71.8	56.3	16.4
A / $R2$	82.7 (+10.9)	76.6 (+20.3)	38.6 (+22.2)
A / $R3$	81.2 (+9.4)	72.2 (+15.9)	32.7 (+16.3)
B / $R1$	69.8	68.1	20.2
B / $R2$	78.1 (+8.3)	80.2 (+12.1)	27.9 (+7.7)
B / $R3$	79.0 (+9.2)	78.5 (+10.4)	38.1 (+17.9)
C / $R1$	55.3	43.1	35.3
C / $R2$	75.3(+20.0)	76.8 (+33.7)	57.7 (+22.4)
C / $R3$	75.9 (+20.6)	76.6 (+33.5)	57.7 (+22.4)

Table 4. Mean F1-scores [%] for the class  $DF$  achieved in five test runs in the evaluation scenario based on random epochs (2016-2021). The classifiers were trained following the rules defined in Section 3.1. Values in parentheses show the gain in F1-score of the classifiers based on rule sets  $R2$  and  $R3$  compared to the baseline ( $R1$ ). The standard deviation of the F1 scores is  $\pm 1.1\%$ .

Regarding the cross-domain results, even larger improvements in the performance can be achieved when using rule sets  $R2$  and  $R3$  compared to the baseline (up to 34% in F1-scores when training in domain C and testing in domain B). In these cases, the highest F1-scores for the class  $DF$  were achieved, demonstrating that the generalization capability of the classifiers is improved when the information contained in the entire time series is used for training. In some cases, the F1-scores in the cross-domain setting were very similar to those the intra-domain results. For instance, when a classifier is trained and evaluated on domain A, F1-scores of approximately 83% were achieved. When independent classifiers were trained using domains B and C and evaluated on domain A, the approximate F1-score values obtained were 79% and 75%, respectively, and, thus, of a similar order of magnitude. A similar behavior was observed when the classifiers were evaluated on domain B. In general, for the domain C, lower F1-scores were obtained in both, the intra-domain and the cross-domain settings, which may be due to the high percentage of cloud coverage in the images, possibly affecting the classifiers in a negative way.

#### 4.5 Visual analysis

A visual example of deforestation prediction of the classifiers trained using rule sets  $R1$ ,  $R2$ , and  $R3$  is shown in Figure 5.

It shows the false-color multi-spectral images for the earlier image  $X_e$  and the later image  $X_l$  from domain A along with the reference  $Y$  and the output predictions for that image pair for different variants of the classifier in an intra-domain and a cross-domain scenario, i.e., when trained in domain A and B, respectively. We present the output predictions without completely masking out the past deforestation areas to analyze how the approach deals with regrowth and secondary deforestation events that happen only in these areas.

For the visual analysis we focus on three areas in the past deforestation region. The region marked by the red outline (north-west) corresponds to an area which shows no forest in both images. Thus, this area should be predicted as no deforestation. A region that does show secondary deforestation is marked by a blue outline (middle of the image). Even though the area is marked as past deforestation, there is forest in the earlier image and no forest in the later image, which indicates a secondary deforestation event. In the region with the green outline, there some deforestation occurred prior to the  $t_e$ , but also in the period between  $t_e$  and  $t_l$ . Whether this case should be predicted as deforestation is unclear and depends on the definition of that class.

A comparison of the three classifiers trained on domain A (second row in Figure 5) shows that the classifiers trained using  $R1$  and  $R3$  make correct predictions for the areas with red and blue outlines. However, the classifier trained using  $R1$  erroneously predicts deforestation in the north-east and south of the image, where the classifier trained using  $R3$  correctly predicts no deforestation. The classifier trained using rule set  $R2$  performs a bit better than rule set  $R1$  in those regions but erroneously predicts deforestation in the area marked by the red outline. Interestingly, the classifiers trained using rule sets  $R1$  and  $R3$  predict no deforestation in the area marked by the green outline, whereas the classifier trained using rule set  $R2$  predicts deforestation. To summarize, considering only those regions in the past deforestation area which can clearly be classified as deforestation or no deforestation, the classifier trained using rule set  $R3$  performs best in the intra-domain scenario.

In the cross-domain scenario (last row in Figure 5), the classifier trained using rule sets  $R1$  and  $R2$  erroneously predicts deforestation in the area marked by the red outline. The classifier trained using rule set  $R3$  correctly predicts no deforestation in that area. Looking at the remaining areas, again the classifier trained using rule set  $R3$  is considered to perform best. It is to be noted that the predictions made by the two classifiers trained using rule sets  $R3$  in the intra-domain and cross-domain scenario are very similar except for the area marked by the green outline, while the predictions made by the classifiers trained using rule sets  $R1$  and  $R2$  differ more. We consider this as an indication that the classifier trained using rule set  $R3$  is more robust to changes in the domain.

Although we do not have reference data for regrowth and secondary deforestation cycles, we conclude from the visual interpretation of the results that the use of time series and the inclusion of additional cases of the  $NDF$  class improved the generalization capacity of the classifier.

## 5. CONCLUSIONS

We have proposed a training strategy to improve the generalisation capability of a classifier to detect deforestation



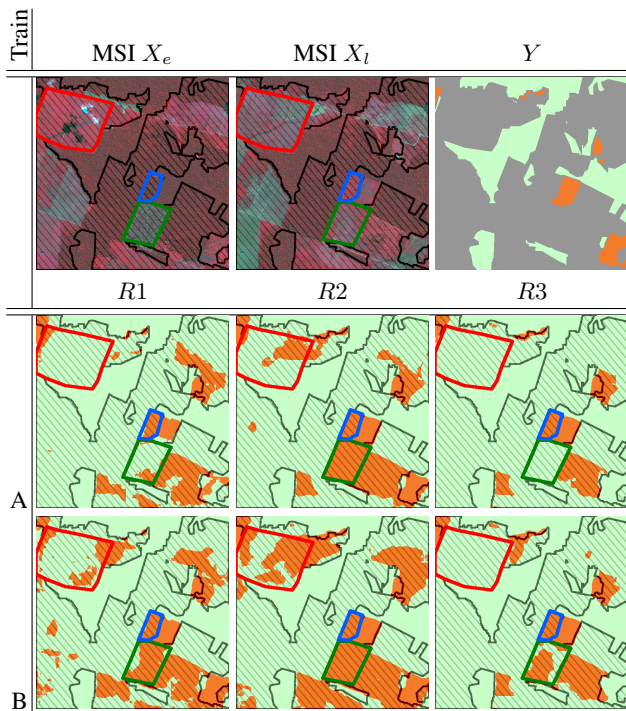


Figure 5. Sample predictions from the classifiers using rules R1, R2, and R3 for an image pair in domain A. Side length of the patches is 256 px. MSI: False-colour multi-spectral image (infra red, red, green) for the earlier image  $X_e$  and the later image  $X_l$ .  $Y$ : Reference label map. Colour-codes: *Unknown* (grey), *Deforested* (orange), *Non-deforested* (green)

using Sentinel-2 time series. Unlike conventional methods selecting a single image pair with dates very close to the beginning and the end of the time interval reflected in a reference map for deforestation, we used arbitrary images pairs from the entire time series and potentially non-synchronized reference maps to build a training set that helps a classifier to generalise better to unseen data. We defined different sets of rules for deriving training labels from the existing reference in order to exploit the available (but not necessarily optimal) information as good as possible. One of these rule sets was designed to identify additional samples for the class *No Deforestation* to improve the classification in areas where there is no forest in both images. This allows the classifier to be more accurate in predicting deforestation events and to reduce its dependency on information about past deforestation. The performance of this strategy was evaluated on three datasets from different geographic locations and showing different types of vegetation. Experimental results demonstrated the superiority of the proposed strategy over the baseline (training with a single image pair), leading to a more precise identification of deforested areas in nearly all cases. In particular, the generalization capability could be improved considerably in a cross-domain setting, with an improvement of up to 22% in the F1-scores of the class *DF* in the scenario considered in the PRODES project (evaluation using images acquired at two subsequent years). Thus, the performance gap due to the differences between the domains could be reduced by leveraging the time series and our definition of the corresponding training labels. We believe that this strategy can potentially be extended in different areas of the Amazon region to automatically detect deforestation and alleviate the manual

annotation task.

As far as potential future work is concerned, we see a high potential of combining the proposed training strategy with unsupervised domain adaptation in order to improve the generalization capabilities in the cross-domain scenario. These would be an attractive extension to mitigate the domain gap between data from different geographic regions even further. In addition, the inclusion of more Sentinel-2 images and self-supervised approaches can be explored as an alternative to deal with the lack of intermediary labels in the time series.

## ACKNOWLEDGEMENTS

This work was conducted in the scope of ForstCARE project, supported by and the Federal Ministry for Economic Affairs and Climate Action, Germany (Bundesministerium für Wirtschaft und Klimaschutz, Funding codes 50EE2017A and 50EE2017B). The authors would also like to acknowledge the support provided by the DAAD (Deutscher Akademischer Austauschdienst) through the funding program 57588368.

## References

- Andrade, R., Mota, G. L. A., da Costa, G. A. O. P., 2022. Deforestation Detection in the Amazon Using DeepLabv3+ Semantic Segmentation Model Variants. *Remote Sensing*, 14(19), 4694.
- Assis, L. F., Ferreira, K. R., Vinhas, L., Maurano, L., Almeida, C., Carvalho, A., Rodrigues, J., Maciel, A., Camargo, C., 2019. TerraBrasilis: a spatial data analytics infrastructure for large-scale thematic mapping. *ISPRS International Journal of Geo-Information*, 8(11), 513.
- Carrillo-Niquete, G. A., Andrade, J. L., Valdez-Lazalde, J. R., Reyes-García, C., Hernández-Stefanoni, J. L., 2022. Characterizing spatial and temporal deforestation and its effects on surface urban heat islands in a tropical city using Landsat time series. *Landscape and Urban Planning*, 217, 104280.
- Chollet, F., 2017. Xception: Deep learning with depthwise separable convolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1251–1258.
- De Bem, P. P., de Carvalho Junior, O. A., Fontes Guimarães, R., Trancoso Gomes, R. A., 2020. Change detection of deforestation in the Brazilian Amazon using landsat data and convolutional neural networks. *Remote Sensing*, 12(6), 901.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 248–255.
- Gomes, V. C., Queiroz, G. R., Ferreira, K. R., 2020. An overview of platforms for big earth observation data management and analysis. *Remote Sensing*, 12(8), 1253.
- He, K., Zhang, X., Ren, S., Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *IEEE International Conference on Computer Vision (ICCV)*, 1026–1034.

- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.
- Iakubovskii, P., 2019. Segmentation models pytorch. [https://github.com/qubvel/segmentation\\_models.pytorch](https://github.com/qubvel/segmentation_models.pytorch).
- INPE, 2021. National institute for space research. earth observation general coordination. monitoring program of the amazon and other biomes. deforestation - legal amazon -. <http://terrabrasil.is.dpi.inpe.br>.
- Kpienbaareh, D., Luginaah, I., Bezner Kerr, R., Wang, J., Poveda, K., Steffan-Dewenter, I., Lupafya, E., Dakishoni, L., 2022. Assessing local perceptions of deforestation, forest restoration, and the role of agroecology for agroecosystem restoration in northern Malawi. *Land Degradation & Development*, 33(7), 1088–1100.
- Masolele, R. N., De Sy, V., Herold, M., Marcos, D., Verbesselt, J., Gieseke, F., Mullissa, A. G., Martius, C., 2021. Spatial and temporal deep learning methods for deriving land-use following deforestation: A pan-tropical case study using Landsat time series. *Remote Sensing of Environment*, 264, 112600.
- Matosak, B. M., Fonseca, L. M. G., Taquary, E. C., Maretto, R. V., Bendini, H. d. N., Adami, M., 2022. Mapping Deforestation in Cerrado Based on Hybrid Deep Learning Architecture and Medium Spatial Resolution Satellite Time Series. *Remote sensing*, 14(1), 209.
- Noa, J., Soto, P., Costa, G., Wittich, D., Feitosa, R., Rottensteiner, F., 2021. Adversarial discriminative domain adaptation for deforestation detection. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3, 151–158.
- Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation. *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 9351, 234–241.
- Soto, P. J., Costa, G. A., Feitosa, R. Q., Ortega, M. X., Bermudez, J. D., Turnes, J. N., 2022. Domain-Adversarial Neural Networks for Deforestation Detection in Tropical Forests. *IEEE Geoscience and Remote Sensing Letters*, 19, 1–5.
- Tasar, O., Giros, A., Tarabalka, Y., Alliez, P., Clerc, S., 2020. Dagnet: Unsupervised, multisource, multitarget, and life-long domain adaptation for semantic segmentation of satellite images. *IEEE Transactions on Geoscience and Remote Sensing*, 59(2), 1067–1081.
- Tasar, O., Tarabalka, Y., Alliez, P., 2019. Incremental learning for semantic segmentation of large-scale remote sensing data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(9), 3524–3537.
- Torres, D. L., Turnes, J. N., Soto Vega, P. J., Feitosa, R. Q., Silva, D. E., Marcato Junior, J., Almeida, C., 2021. Deforestation detection with fully convolutional networks in the Amazon Forest from Landsat-8 and Sentinel-2 images. *Remote Sensing*, 13(24), 5084.
- Vega, P. J. S., da Costa, G. A. O. P., Feitosa, R. Q., Adarme, M. X. O., de Almeida, C. A., Heipke, C., Rottensteiner, F., 2021. An unsupervised domain adaptation approach for change detection and its application to deforestation mapping in tropical biomes. *ISPRS Journal of Photogrammetry and Remote Sensing*, 181, 113–128.
- Wang, Z., Liu, D., Liao, X., Pu, W., Wang, Z., Zhang, Q., 2023. SiamHRnet-OCR: A Novel Deforestation Detection Model with High-Resolution Imagery and Deep Learning. *Remote Sensing*, 15(2), 463.
- Wittich, D., Rottensteiner, F., 2021. Appearance based deep domain adaptation for the classification of aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 180, 82–102.