

## An unsupervised domain adaptation approach for change detection and its application to deforestation mapping in tropical biomes

Pedro Juan Soto Vega <sup>a,\*</sup>, Gilson Alexandre Ostwald Pedro da Costa <sup>b</sup>, Raul Queiroz Feitosa <sup>a</sup>, Mabel Ximena Ortega Adarme <sup>a</sup>, Claudio Aparecido de Almeida <sup>c</sup>, Christian Heipke <sup>d</sup>, Franz Rottensteiner <sup>d</sup>

<sup>a</sup> Computer Vision Lab, Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Rio de Janeiro, Brazil

<sup>b</sup> Institute of Mathematics and Statistics, State University of Rio de Janeiro (UERJ), Rio de Janeiro, Brazil

<sup>c</sup> Amazon Program Coordination, National Institute for Space Research (INPE), São José dos Campos, Brazil

<sup>d</sup> Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover (LUH), Germany



### ARTICLE INFO

**Keywords:**  
Deforestation detection  
Change detection  
Domain adaptation  
CycleGAN  
Deep learning  
Remote sensing

### ABSTRACT

Changes in environmental conditions, geographical variability and different sensor properties typically make it almost impossible to employ previously trained classifiers for new data without a significant drop in classification accuracy. Domain adaptation (DA) techniques have been proven useful to alleviate that problem. In particular, appearance adaptation techniques may be used to adapt images from a specific dataset in such a way that the generated images have a style that is similar to the images from another dataset. Such techniques are, however, prone to creating artifacts that hinder proper classification of the adapted images. In this work we propose an unsupervised DA approach for change detection tasks, which is based on a particular appearance adaptation method: the Cycle-Consistent Generative Adversarial Network (CycleGAN). Specifically, we extend that method by introducing additional constraints in the training phase of the model components, which make it preserve the semantic structure and class transitions in the adapted images. We evaluate the proposed approach on a deforestation detection application, considering different sites in the Amazon rain-forest and in the Brazilian Cerrado (savanna) using Landsat-8 images. In the experiments, each site corresponds to a domain, and the accuracy of a classifier trained with images and references from one (source) domain is measured in the classification of another (target) domain. The results show that the proposed approach is successful in producing artifact-free adapted images, which can be satisfactorily classified by the pre-trained source classifiers. On average, the accuracies achieved in the classification of the adapted images outperformed the baselines (when no adaptation was made) by 7.1% in terms of mean average precision, and 9.1% in terms of F1-Score. To the best of our knowledge, the proposed method is the first unsupervised domain adaptation approach devised for change detection.

### 1. Introduction

As a direct result of anthropogenic activities, environmental changes have reached unprecedented levels, threatening food production, natural resources, and many forms of life. In this respect, deforestation of natural forests constitutes one of the largest sources of greenhouse gas emissions, being responsible for the reduction of carbon storage and for invaluable losses in biodiversity (De Sy et al., 2015). The Amazon forest,

for instance, which contains approximately 10% of all the biomass on the planet, and a similar proportion of all living species, has been facing serious threats as a result of the impact of unsustainable economic activities (Goodman et al., 2019; Malingreau et al., 2012; Nogueron et al., 2006).

Therefore, monitoring Earth's surface changes, especially those that adversely impact the environment, has become a priority for a number of authorities around the world. Nevertheless, the efficient detection of

\* Corresponding author.

E-mail addresses: [psoto@ele.puc-rio.br](mailto:psoto@ele.puc-rio.br) (P.J. Soto Vega), [gilson.costa@ime.uerj.br](mailto:gilson.costa@ime.uerj.br) (G.A.O.P. Costa), [raul@ele.puc-rio.br](mailto:raul@ele.puc-rio.br) (R.Q. Feitosa), [mortega@ele.puc-rio.br](mailto:mortega@ele.puc-rio.br) (M.X. Ortega Adarme), [claudio.almeida@inpe.br](mailto:claudio.almeida@inpe.br) (C.A. Almeida), [heipke@ipi.uni-hannover.de](mailto:heipke@ipi.uni-hannover.de) (C. Heipke), [rottensteiner@ipi.uni-hannover.de](mailto:rottensteiner@ipi.uni-hannover.de) (F. Rottensteiner).

URL: <http://www.lvc.ele.puc-rio.br/wp/> (P.J. Soto Vega).

such changes is a difficult task, which requires large amounts of Remote Sensing (RS) data.

Fortunately, the improvements in RS technology in the past decades, notably in orbital Earth observation systems, have contributed to a vast increase in the availability of RS data for environmental monitoring applications. More recently, deep learning techniques for image analysis, particularly those based on Convolutional Neural Networks (CNNs), have evolved in such a way that they now represent the state-of-the-art in many application fields, including RS (Zhu et al., 2017b; LeCun et al., 2015; Szegedy et al., 2015).

Deep Learning models, however, are known to demand large amounts of labeled data for proper training, which is a problem for many RS applications mostly because of the costs involved in the field surveys and visual interpretation required to produce such reference data.

Moreover, changes in environmental conditions, geographical variability and different sensor properties typically makes it almost impossible to employ previously trained classifiers for new data without a significant decrease in classification accuracy. While this phenomenon represents a problem for any supervised classifier, considering the aforementioned high demand for labeled data, it may seriously impair the operationalization of deep learning-based classification approaches in real-world applications.

Transfer learning techniques (Weiss et al., 2016; Hu et al., 2015; Pan and Yang, 2010) can be used to mitigate the problem. In transfer learning, data is assumed to be available in different *domains*, each of them characterized by a feature space  $X$  and a corresponding feature distribution  $P(X)$ . In each domain, a learning *task* is to be solved, which involves a definition of the set  $C$  of object classes to be differentiated and a function that predicts the class label for a feature vector. In particular, one usually differentiates a *source domain* in which a large amount of labeled training samples is assumed to be available and a *target domain* in which only few or no labelled samples are available. The goal of transfer learning is to use the information in the source domain to improve the performance of a classifier in the target domain. Whereas the domains and the tasks may be different, they have to be related for transfer learning to be applicable (Pan and Yang, 2010). In the context of deep learning, the standard strategy for transfer learning is to pre-train a CNN using labelled data in a source domain and then fine-tune the pre-trained classifier using a small amount of labelled data in the target domain. This strategy can improve classification accuracy on that dataset, but the potential improvement would depend on the amount of available labeled data and the level of discrepancy between the datasets both in terms of the features and the classes to be differentiated (Yosinski et al., 2014).

*Domain Adaptation* (DA) is a specific setting of transfer learning in which the feature spaces and the class labels to be differentiated are considered to be identical in both domains, but the joint distribution  $P(X, C)$  of the features and the class labels is expected to be different; this discrepancy between the domain is frequently referred to as *domain shift* (Wang and Deng, 2018; Schenkel and Middelmann, 2019; Tuia et al., 2016; Sun and Saenko, 2016; Ganin and Lempitsky, 2015; Tzeng et al., 2014). We are interested in a variant called *unsupervised* DA in Computer Vision (Wang and Deng, 2018), in which labelled data are only available in the source domain. In this setting, the classifier has to be trained to perform well in the target domain without being able to use labelled data in that domain. Note that in remote sensing, this setting is sometimes referred to as *semi-supervised* DA (Tuia et al., 2016); we stick to the nomenclature of (Wang and Deng, 2018) for the remainder of this paper.

In change detection, the application discussed in this paper, a domain is associated with a pair of images of the same geographical location, taken at different epochs and using the same sensor. The domain shift is due to the fact that the source and the target domains correspond to different geographical regions, so that the objects of the same type look different in the images; in addition, there may also be a different distribution of object classes.

Among the previously proposed DA solutions, those based on

Generative Adversarial Network (GAN) (Goodfellow et al., 2014) concepts represent the current state-of-the-art. GAN-based DA techniques have been proposed for many RS applications such as urban land cover mapping (Tasar et al., 2020; Schenkel and Middelmann, 2019; Wittich and Rottensteiner, 2019), cloud detection (Mateo-García et al., 2019) and change detection in urban areas (Deng et al., 2019).

Such DA approaches can be broadly divided into *representation matching* (Wittich and Rottensteiner, 2019; Ganin and Lempitsky, 2015) and what could be called *appearance adaptation* techniques (Wittich and Rottensteiner, 2021; Tasar et al., 2020; Zhu et al., 2017a). Representation matching aims at aligning the features extracted from the domains in a common (latent) feature space, and involves either creating a different classifier for each domain, or a single classifier that is able to properly classify data from both domains. Appearance adaptation is mainly supported by so-called Image-to-Image (I2I) translation methods (Zhu et al., 2017a; Isola et al., 2017), which aim at transforming images from a target domain so that the generated images have a style (appearance) that is similar to the images from a source domain. An interesting characteristic of appearance adaptation approaches is that they can be regarded as *task-agnostic*, i.e. the adaptation is not bound to any classification problem. Such approaches are, however, prone to creating artifacts in the generated images, which severely limits the performance of classifiers trained using source domain images. Other approaches combine concepts of representation matching and image translation, e.g., (Murez et al., 2018; Hoffman et al., 2017), but they also involve training a particular classifier for the target domain.

In this work we propose a deep learning-based, unsupervised domain adaptation approach specialized for change detection. It can be regarded as an appearance adaptation approach and is based on Cycle-Consistent Generative Adversarial Networks (CycleGANs) (Zhu et al., 2017a). Our goal is to transform a target domain image pair so that the generated images look similar to the ones from a source domain, and, at the same time, preserve the semantic structure and class transitions observed in the original (target) image pair, thus creating artifact-free adapted images that can be properly classified using a classifier trained solely using labelled source domain data. To achieve this goal, we devised a way to extend the CycleGAN model, introducing additional constraints in the training phase of the model components.

We further evaluated the proposed approach on a deforestation detection application, considering different sites in the Amazon rainforest and in the Brazilian Cerrado (savanna). Whereas we show results for a specific change detection task (deforestation mapping), the principles proposed in this paper are more general in the sense that they can also be applied to other change detection applications. The contribution of this paper is thus threefold:

- We propose a new unsupervised domain adaptation approach for change detection applications based on the CycleGAN model. To the best of our knowledge, ours is the first work to address unsupervised DA for change detection and considering domains to consist of pairs of images from different epochs.
- We adapt the CycleGAN model to perform image translation over input image pairs, and introduce new regularization constraints in the training of the model, which prevent it from creating artifacts that may hinder the subsequent change classification. This regularization is based on the assumption that the change in appearance between epochs is less affected by a domain shift than the appearance of the objects itself and, thus, leverages the availability of image pairs in each domain.
- We evaluate the proposed approach on a deforestation detection application, considering three different domains associated with images from different sites in the Amazon and Brazilian Cerrado biomes.

Moreover, we provide the code of the proposed method, as well as the datasets used in the experiments carried out in this work,<sup>1</sup> thus enabling further research and grounds for comparative evaluation of other domain adaptation techniques devised for change detection.

In the next section we summarize previously proposed change detection approaches as well as some of the most recent domain adaptation solutions for remote sensing applications. After that, we describe the proposed approach, the experimental analysis, and discuss the results of the experiments. Finally, we present conclusions and directions for further research.

## 2. Related work

In the context of RS, change detection refers to the process of identifying differences on the Earth surface by observing a particular location at different times. Several change detection models have been proposed in the literature, and employed in a variety of applications.

Algebra-based models rely on features computed with band-wise mathematical operations (Asokan and Anitha, 2019). They include image difference methods (Bruzzone and Prieto, 2000), image ratio (Afify, 2011) and Change Vector Analysis (CVA) (Liu et al., 2015; Malila, 1980), which has been the basis for more advanced approaches, e.g., (Thonfeld et al., 2016).

Transformation-based algorithms (Asokan and Anitha, 2019) involve mapping intensity values into a new feature space through methods such as Principal Component Analysis (PCA) (Sadeghi et al., 2016; Pearson, 1901), Tasseled Cap Transformation (Han et al., 2007; Kauth and Thomas, 1976) and Slow Feature Analysis (SFA) (Wu et al., 2013; Wiskott and Sejnowski, 2002; Wiskott, 1999). More advanced methods include probability graph models (Zhang et al., 2012; Koller and Friedman, 2009), Markov Random Fields (Gu et al., 2017; Li, 2009), Conditional Random Fields (Zhou et al., 2016; Lafferty et al., 2001), Wavelets (Celik and Ma, 2010), among many others. All of the above-mentioned methods explore shallow features, most of which are hand-crafted, potentially deficient in representing crucial change information, which, according to Chen et al. (2020), might limit their performance.

### 2.1. Deep learning for change detection

As in many other areas, deep learning-based methods represent the current state-of-the-art in change detection problems. Most of the recently proposed approaches follow Patch Wise Classification (PWC) or Fully Convolutional (FC) schemes: PWC produces a global decision by considering two patches of the same image locations taken at different times (Andrade et al., 2020), while FC models return dense pixel level decisions. Remarkable PWC-based methods can be found in the literature. Chu et al. (2016) propose a CD method that uses a pair of Deep Belief Networks (DBNs) (Hinton, 2009), one for each patch, trained to minimize their output distances for non-changed patches, and maximize them for the changed ones. Afterwards, the representations learned by the DBNs are submitted to PCA based k-means clustering, which produces the final outcome. Methods like (Daudt et al., 2018b and Zagoruyko and Komodakis, 2015) are based on a similar idea by using Siamese Convolutional Neural Networks (S-CNN) (Chopra et al., 2005). The models use the last layers of two convolutional networks with shared weights as input of a fully connected layer that delivers the final decision. Alternatively, Daudt et al. (2018b) employ an Early Fusion (EF) scheme, in which the input data is the concatenation of images taken at different epochs. Specifically focusing on deforestation detection in tropical forests, Ortega Adarme et al. (2020) propose an approach which employs EF and PWC.

Fully convolutional architectures for change detection have been

proposed more recently, generally delivering higher accuracies than the ones based on PWC. For instance, Daudt et al. (2018a) use a fully convolutional S-CNN and the EF scheme, obtaining results that are superior to those reported in (Daudt et al., 2018b). Other works rely on different FC architectures, such as U-net (Alexakis and Armenakis, 2020; Li et al., 2019) and DeepLabv3+ (Andrade et al., 2020). The latter work compared deep learning PWC and FC models applied to deforestation detection.

### 2.2. Domain adaptation for change detection

Among the previously proposed DA approaches, those based on the selection of invariant features have been the most commonly employed ones in remote sensing (Saha et al., 2020; Deng et al., 2019; Mateo-García et al., 2019; Saha et al., 2019; Schenkel and Middelmann, 2019; Wittich and Rottensteiner, 2019) and, in general, in most of the related computer vision applications (Gholami et al., 2020; Ganin and Lempitsky, 2015). Specifically for change detection, Chen et al. (2020) propose a DA procedure that employs a siamese architecture to extract spatial-spectral features. In sequence, the method computes a difference vector that serves as the input of several fully connected layers, which produce the final change detection map. Similarly, (Song et al., 2019) aims at aligning features in a common subspace. A CNN pre-trained on the source domain extracts features from source and target patches, followed by the selection of eigenvectors that serve as bases of a feature subspace. Subsequently, a subspace alignment procedure is carried out by learning a transformation through the Bregman matrix divergence (Dhillon and Tropp, 2008) algorithm to minimize the discrepancy between source and target domain. Finally, the source data, projected into the newly aligned subspace, is used to fine-tune the classification layer of the pre-trained CNN. The comparison of DA-based representation learning models was the main objective of (Elshamli et al., 2017), which evaluated, among other architectures, Denoising Autoencoders and Domain Adversarial Neural Networks (Ganin and Lempitsky, 2015). A common problem of the aforementioned feature space-based models is that they may perform poorly when employed in scenarios characterized by imbalanced quantities of (unlabeled) samples per class in the target domains (Ming Harry Hsu et al., 2015).

Another class of domain adaptation approaches are based on image-to-image translation methods, such as the Cycle-Consistent Generative Adversarial Network (CycleGAN) (Zhu et al., 2017a). Those approaches generally aim at generating fake images that have a style that is similar to the source or target domain images. The generated images can then be used to fine-tune a classifier, e.g., using images from the source domain adapted to the style of the target domain, and labeled data from the source domain. Alternatively, target images adapted to the source domain can be evaluated using a classifier trained with source domain images and labeled samples. The main problem of I2I methods is that the generated (adapted) target domain images tend to match the class distributions of the source domain (or vice versa when the domains are swapped). This means that objects in the adapted images may have their appearances changed to incorrect classes, severely hindering further classification. Despite that difficulty, some works have evaluated the application of the CycleGAN model in remote sensing. For instance, Benjdira et al. (2019) propose a CycleGAN-based method to generate fake data that is used to fine-tune a classifier previously trained with the original source images. However, due to the artifacts created by the adaptation process, the quality of the generated data was poor. In another Cycle GAN-based DA method (Soto et al., 2020), this time for deforestation detection, the authors introduce a new regularization loss term in the attempt to reduce the generation of such artifacts. The method, however, has a limited adaptation capacity.

In the attempt to avoid the class distribution matching problem, Tasar et al. (2020) proposed the so-called ColorMapGAN. The authors designed a generator that learns a color mapping transformation for each color present in the source and target domain images.

<sup>1</sup> <http://www.lvc.ele.puc-rio.br/wp/?cat=41>.

Unfortunately, the ColorMapGAN (Tasar et al., 2020) has a computation complexity that grows exponentially with the number of image bands, which makes the method not suited for images with more than three bands, such as the ones from Landsat 8, used in the experiments carried out in this work.

### 3. Method

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) have caught the computer vision community's attention due to their ability to learn data distributions through an unsupervised procedure. Consequently, a variety of methods and applications based on GANs have been proposed to date, including DA methods for remote sensing data.

The method presented in this paper relies on the Cycle-Consistent Generative Adversarial Network (CycleGAN) approach (Zhu et al., 2017a) and exploit its ability to perform image-to-image translation between different domains (Deng et al., 2019; Murez et al., 2018; Hoffman et al., 2017). We extend the CycleGAN approach so that it can be employed in unsupervised DA for change detection based on remote sensing imagery. To make this paper self-contained, we start with a concise review of the original CycleGAN approach (Section 3.1). The proposed extension for change detection applications and, thus, our main methodological contribution, is presented in Section 3.2.

#### 3.1. Cycle-Consistent Generative Adversarial Network (CycleGAN)

CycleGANs try to learn how to adapt the appearance of images  $x$  from one domain  $X$  so that they look like images  $y$  from another domain  $Y$  or vice versa. A feature that distinguishes CycleGAN from other GAN models is that it does not require pairs of corresponding images from both domains for training.

CycleGAN training aims at learning mapping functions  $\mathcal{G} : X \rightarrow Y$  and  $\mathcal{F} : Y \rightarrow X$  so that the produced images  $\mathcal{G}(x)$  and  $\mathcal{F}(y)$  are indistinguishable from the real set of images in  $Y$  and  $X$ , respectively. These mapping functions are implemented as convolutional neural networks (CNN). In order to learn appropriate mappings, two additional networks referred to as *discriminators*  $\mathcal{D}_X$  and  $\mathcal{D}_Y$  are required. It is the task of the discriminator  $\mathcal{D}_X$  to predict whether an input image corresponds to a real sample of domain  $X$  or whether it was produced by the mapping  $\mathcal{F}(y)$ . Thus,  $\mathcal{D}_X$  is trained such that its output  $\mathcal{D}_X(x)$  for an image  $x \in X$  is close to one, while its output  $\mathcal{D}_X(\mathcal{F}(y))$  for an image produced by the mapping  $\mathcal{F}(y)$  of an image  $y \in Y$  should be close to zero. Similarly,  $\mathcal{D}_Y$  is trained to predict whether its input is a sample from the domain  $Y$  or whether it was produced by the mapping  $\mathcal{G}(x)$ . In the training procedure, the optimal mapping functions  $\mathcal{G}^*$  and  $\mathcal{F}^*$  are determined according to:

$$\mathcal{G}^*, \mathcal{F}^* = \arg \min_{\mathcal{G}, \mathcal{F}} \max_{\mathcal{D}_X, \mathcal{D}_Y} \mathcal{L}(\mathcal{G}, \mathcal{F}, \mathcal{D}_X, \mathcal{D}_Y), \quad (1)$$

based on the CycleGAN objective function  $\mathcal{L}(\cdot)$ . This optimization process, known as the *minimax game*, is performed in a sequence of two alternating steps. In the first step, the parameters of the mappings  $\mathcal{G}$  and  $\mathcal{F}$  are kept constant while the parameters of the discriminators  $\mathcal{D}_X$  and  $\mathcal{D}_Y$  are updated so as to maximize  $\mathcal{L}(\cdot)$ . In the second step, the discriminator parameters remain fixed while the parameters of the mappings are updated so as to minimize  $\mathcal{L}(\cdot)$ . As usual in adversarial training, in this way, the two mappings should learn to produce images that can fool the discriminator, whereas the discriminators should be trained to make this task difficult for the two mapping networks (Goodfellow et al., 2014).

The CycleGAN objective function  $\mathcal{L}(\cdot)$  is defined as (Zhu et al., 2017a):

$$\begin{aligned} \mathcal{L}(\mathcal{G}, \mathcal{F}, \mathcal{D}_X, \mathcal{D}_Y) &= \mathcal{L}_{GAN}(\mathcal{G}, \mathcal{D}_Y, X, Y) \\ &+ \mathcal{L}_{GAN}(\mathcal{F}, \mathcal{D}_X, Y, X) \\ &+ \lambda_{cyc} \mathcal{L}_{cyc}(\mathcal{G}, \mathcal{F}) + \lambda_{idt} \mathcal{L}_{idt}(\mathcal{G}, \mathcal{F}), \end{aligned} \quad (2)$$

In Eq. (2), the first two terms,  $\mathcal{L}_{GAN}(\mathcal{G}, \mathcal{D}_Y, X, Y)$  and  $\mathcal{L}_{GAN}(\mathcal{F}, \mathcal{D}_X, Y, X)$ , also illustrated in Fig. 1a, represent the *adversarial losses* of the model, as in the basic GAN approach (Goodfellow et al., 2014). Each of these terms is associated with the capacity of the respective generators to match the data distributions observed in the respective target domains. Following the least squares GAN (Mao et al., 2017) formulation and observing that  $\mathbb{E}_{z \sim p_d(z)}[\cdot]$  denotes the expectation value of the argument for images  $z$  drawn from the data distribution  $p_d(z)$  of the corresponding domain, with  $z \in \{x, y\}$ , these terms are defined as:

$$\begin{aligned} \mathcal{L}_{GAN}(\mathcal{G}, \mathcal{D}_Y, X, Y) &= \mathbb{E}_{y \sim p_d(y)}[(\mathcal{D}_Y(y))^2] \\ &+ \mathbb{E}_{x \sim p_d(x)}[(\mathcal{D}_Y(\mathcal{G}(x)) - 1)^2], \end{aligned} \quad (3)$$

$$\begin{aligned} \mathcal{L}_{GAN}(\mathcal{F}, \mathcal{D}_X, Y, X) &= \mathbb{E}_{x \sim p_d(x)}[(\mathcal{D}_X(x))^2] \\ &+ \mathbb{E}_{y \sim p_d(y)}[(\mathcal{D}_X(\mathcal{F}(y)) - 1)^2], \end{aligned} \quad (4)$$

Note that the loss in Eq. (3) will become a minimum (zero) if  $\mathcal{D}_Y(y) = 0$  and  $\mathcal{D}_Y(\mathcal{G}(x)) = 1$ , i.e. if the discriminator  $\mathcal{D}_Y$  is fooled. Minimizing this loss with respect to the mappings will adapt them to fool the discriminator (step 1 of the minimax game), whereas this loss has to be maximized for the discriminator to do a good job (step 2 of the minimax game). Eq. (4) follows the same logic for  $\mathcal{D}_X(x)$  and  $\mathcal{D}_X(\mathcal{F}(y))$ .

The third term in Eq. (2),  $\mathcal{L}_{cyc}(\mathcal{G}, \mathcal{F})$ , is the *cycle consistency loss*, which enforces the sequence of mappings  $\mathcal{G}(\mathcal{F}(y))$  and  $\mathcal{F}(\mathcal{G}(x))$  to deliver a result close to the input, as illustrated in Figs. 1(b) and 1(c);  $\lambda_{cyc}$  is a weighting coefficient. The loss  $\mathcal{L}_{cyc}(\mathcal{G}, \mathcal{F})$  is given by (Zhu et al., 2017a):

$$\begin{aligned} \mathcal{L}_{cyc}(\mathcal{G}, \mathcal{F}) &= \mathbb{E}_{x \sim p_d(x)}[\|x - \mathcal{F}(\mathcal{G}(x))\|_1] \\ &+ \mathbb{E}_{y \sim p_d(y)}[\|y - \mathcal{G}(\mathcal{F}(y))\|_1], \end{aligned} \quad (5)$$

where  $\|\cdot\|_1$  denotes the  $L_1$  norm.

The fourth term in Eq. 2 is the *identity loss*  $\mathcal{L}_{idt}(\mathcal{G}, \mathcal{F})$ :

$$\begin{aligned} \mathcal{L}_{idt}(\mathcal{G}, \mathcal{F}) &= \mathbb{E}_{y \sim p_d(y)}[\|y - \mathcal{G}(y)\|_1] \\ &+ \mathbb{E}_{x \sim p_d(x)}[\|x - \mathcal{F}(x)\|_1]. \end{aligned} \quad (6)$$

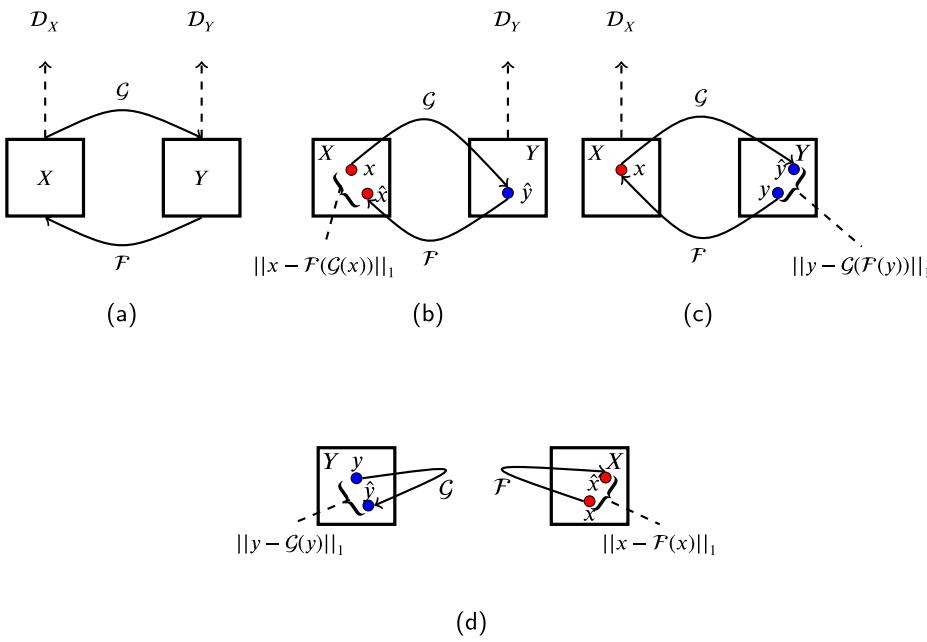
This loss enforces the generated target mappings to preserve some of the characteristics of the corresponding source images, e.g., the structure of the objects or the range of intensity values present in the original images (Fig. 1(d)). Scaled by the coefficient  $\lambda_{idt}$ ,  $\mathcal{L}_{idt}(\mathcal{G}, \mathcal{F})$  is a regularization term that forces the functions  $\mathcal{G}$  and  $\mathcal{F}$  to be close to identity mapping, i.e.,  $\mathcal{G}(y) \approx y$  and  $\mathcal{F}(x) \approx x$ .

#### 3.2. Proposed method

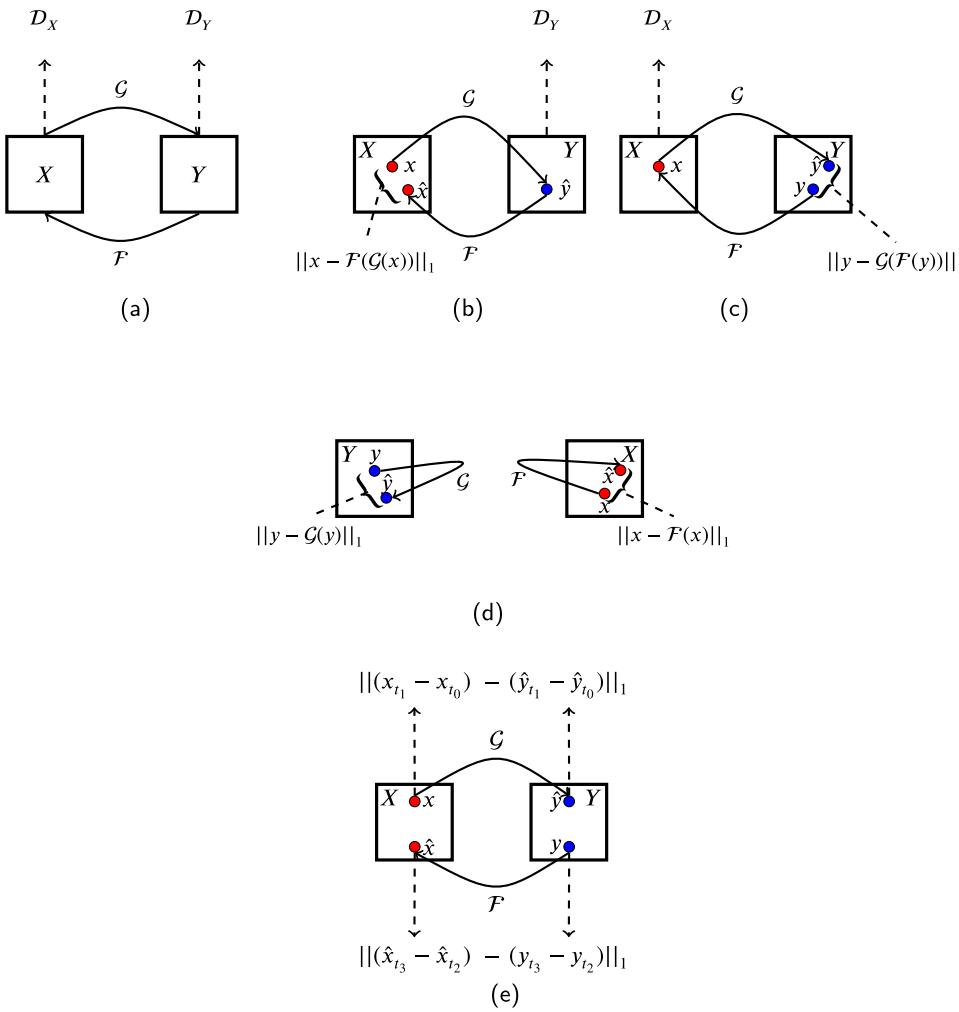
The proposed method aims at alleviating the domain shift phenomenon in the context of change detection by using a DA approach based on CycleGANs. The general strategy of the proposed DA approach requires the adaptation of the appearance of pairs of images from a target domain so that a change detection method trained using image pairs and labeled samples from a source domain could properly classify the adapted image pairs without any further training.

For this purpose, we propose a CycleGAN-based method which learns nonlinear mapping functions that take a pair of remote sensing images of a particular geographic location as input, and generates corresponding, translated images, adapted to the conditions and characteristics of a second pair of images, of a different geographic location. For the purpose of DA, only the mapping from the target domain to the source domain is relevant; the other mapping is only important in the context of training, because it allows for the use of the cycle consistency loss in Eq. (2) for regularization.

Let  $x = [x_{t_0} : x_{t_1}]$  and  $y = [y_{t_2} : y_{t_3}]$  denote two co-registered, concatenated pairs of multispectral remote sensing images, where  $[::]$  is the concatenation operator,  $x_{t_0}, x_{t_1}$  and  $y_{t_2}, y_{t_3}$  represent co-registered image pairs taken at epochs  $t_0 < t_1, t_2 < t_3$ . Let  $X$  and  $Y$  denote the source and target domains, respectively, with  $x \in X$  and  $y \in Y$ . It is worth to note that  $t_0 = t_2$  and  $t_1 = t_3$  may occur if  $X$  and  $Y$  refer to different geographic locations. Moreover, we denote as  $\hat{y} = [\hat{y}_{t_0} : \hat{y}_{t_1}]$  the result of computing



**Fig. 1.** Overview of the CycleGAN model. In the figure, each box represents a domain,  $X$  or  $Y$ , and the colored circles represent samples/images from each domain:  $x$  and  $y$  represent real samples, and  $\hat{x}$  and  $\hat{y}$  represent adapted samples. Solid arrows indicate the mapping functions, and dashed arrows, the processing flow. Dashed lines joined with braces specify terms of the CycleGAN's loss function. As represented in (a), the CycleGAN model is composed of two generators, which are trained to learn mapping functions from one domain to the other, i.e.,  $G: X \rightarrow Y$  and  $F: Y \rightarrow X$ , and two discriminators,  $D_X$  and  $D_Y$ , which are trained to discern whether or not an image is real or generated through  $F$  or  $G$ , respectively. The CycleGAN's loss function contains cycle-consistency terms:  $\|x - F(G(x))\|_1$  (b), and  $\|y - G(F(y))\|_1$  (c), which enforce the sequence of mappings  $G(F(y))$  and  $F(G(x))$  to deliver results which are similar to the respective inputs, aiming at regularizing the possible mappings associated with each generator. The loss function also contains identity loss terms  $\|y - G(y)\|_1$  and  $\|x - F(x)\|_1$  (d), which enforce the respective mappings to preserve some of the characteristics of the input images. Illustration adapted from (Zhu et al., 2017a).



**Fig. 2.** Overview of the proposed CycleGAN-based domain adaptation framework for change detection. As in Fig. 1, the boxes represent the domains  $X$  and  $Y$ . The samples, represented by red and blue circles, are composed of two concatenated remote sensing images, i.e.,  $x = [x_{t_0} : x_{t_1}]$  and  $y = [y_{t_2} : y_{t_3}]$ . As in the original CycleGAN framework, the training procedure involves two mapping functions  $G: X \rightarrow Y$  and  $F: Y \rightarrow X$ , as well as the associated discriminators  $D_Y$  and  $D_X$  (a). The proposed model loss function also contains cycle consistency terms:  $\|x - F(G(x))\|_1$  (b), and  $\|y - G(F(y))\|_1$  (c), which enforce the sequence of mappings  $G(F(y))$  and  $F(G(x))$  to deliver results which are similar to the respective inputs, aiming at regularizing the possible mappings associated with each generator. The loss function also contains identity loss terms:  $\|y - G(y)\|_1$  and  $\|x - F(x)\|_1$  (d), which enforce the respective mappings to preserve some of the characteristics of the input images. To further regularize the mappings  $F$  and  $G$ , we introduced a *difference* loss term, which aims at preserving the change transitions from both domains in the adapted image pairs:  $(x_{t_1} - x_{t_0}) \approx (\hat{x}_{t_1} - \hat{x}_{t_0})$  and  $(y_{t_3} - y_{t_2}) \approx (\hat{y}_{t_3} - \hat{y}_{t_2})$ , as represented in (e). Illustration adapted from (Zhu et al., 2017a).

$\mathcal{G}(x)$ , and as  $\hat{x} = [\hat{x}_{t_2} : \hat{x}_{t_3}]$  the result of computing  $\mathcal{F}(y)$ .

Considering that the inputs and outputs of the image adaptation approach consist of stacked pairs of images, the adapted image pair should preserve the spatial structures (i.e., boundaries of image objects) observed in the individual images that comprise the input of the corresponding mapping ( $\mathcal{G}$  or  $\mathcal{F}$ ). That would be an effect of the *identity* constraint.

Moreover, as an effect of the adversarial training of the mappings, it is expected that the prior distribution of class transitions observed in the target domain be mimicked in the images adapted from the source domain (and vice versa). That behaviour, however, is not generally desirable considering a change detection application, the main problem being that the mapping functions would not be constrained to create class transitions where no change was present in the images to be adapted, nor to freely adapt the spatial distribution of transitions in the resultant images.

In essence, our approach relies on the assumption that for change detection to be performed properly on the adapted image pairs, changes present in the input (source) image pair must be properly mapped and matched in the generated (target) image pairs. That is, change observed from  $x_{t_0}$  to  $x_{t_1}$ , or from  $y_{t_2}$  to  $y_{t_3}$ , must be properly represented in the pair  $\hat{y}_{t_0}$  and  $\hat{y}_{t_1}$ , or in the pair  $\hat{x}_{t_2}$  and  $\hat{x}_{t_3}$ , respectively.

Therefore, we propose to add a new regularization term to the CycleGAN objective function, as illustrated in Fig. 2(e). The new term is meant to direct the attention of the model to map not only global or object level spectral information during training, but also change structures between domains. Moreover, the proposed term is intended to provide the model with a spatial prior, indicating where the mapping functions should place class transitions, and giving them clues about the intensities and orientations of those changes. Thus, the CycleGAN objective function  $\mathcal{L}(\cdot)$  given in Eq. (2) is redefined as:

$$\begin{aligned} \mathcal{L}(\mathcal{G}, \mathcal{F}, D_X, D_Y) &= \mathcal{L}_{GAN}(\mathcal{G}, \mathcal{D}_Y, X, Y) \\ &+ \mathcal{L}_{GAN}(\mathcal{F}, \mathcal{D}_X, Y, X) \\ &+ \lambda_c \mathcal{L}_{cyc}(\mathcal{G}, \mathcal{F}) + \lambda_d \mathcal{L}_{idt}(\mathcal{G}, \mathcal{F}) \\ &+ \lambda_{diff} \mathcal{L}_{diff}(\mathcal{G}, \mathcal{F}), \end{aligned} \quad (7)$$

where the proposed *difference constraint*  $\mathcal{L}_{diff}$  is devised to compel the mapping functions  $\mathcal{G}$  and  $\mathcal{F}$  to preserve the difference vector computed over the pixels of the input image pairs in its adapted version, and  $\lambda_{diff}$  is the weight of the new loss term.

The design of the new loss term  $\mathcal{L}_{diff}$  entails the main contributions of the present paper. In the next sections we present two different formulations for  $\mathcal{L}_{diff}$  which are investigated in this work.

### 3.2.1. CycleGAN D

The first formulation for  $\mathcal{L}_{diff}$ , which we call *CycleGAN D*, enforces the preservation of magnitude and orientation of the difference vector for each pixel location, formally:

$$\begin{aligned} x_{t_1}(p) - x_{t_0}(p) &= \hat{y}_{t_1}(p) - \hat{y}_{t_0}(p) \\ y_{t_3}(q) - y_{t_2}(q) &= \hat{x}_{t_3}(q) - \hat{x}_{t_2}(q), \end{aligned} \quad (8)$$

where  $x_{t_0}(p), x_{t_1}(p), y_{t_2}(q)$ , and  $y_{t_3}(q)$  represent the spectral vectors at pixel locations defined by the coordinate vectors  $p$  and  $q$ . Accordingly,  $\hat{x}_{t_2}(q), \hat{x}_{t_3}(q), \hat{y}_{t_0}(p)$ , and  $\hat{y}_{t_1}(p)$  represent pixel spectral vectors at locations  $q$  and  $p$  in the corresponding adapted image. We can generalize Eqs. (8) to express the vector differences over all pixels of real and generated images as follows:

$$\begin{aligned} x_{t_1} - x_{t_0} &= \hat{y}_{t_1} - \hat{y}_{t_0} \\ y_{t_3} - y_{t_2} &= \hat{x}_{t_3} - \hat{x}_{t_2}, \end{aligned} \quad (9)$$

and consequently,  $\mathcal{L}_{diff}$  takes the form:

$$\begin{aligned} \mathcal{L}_{diff}(\mathcal{G}, \mathcal{F}) &= \mathbb{E}_{x \sim p_d(x)} [\|(x_{t_1} - x_{t_0}) - (\hat{y}_{t_1} - \hat{y}_{t_0})\|_1] \\ &+ \mathbb{E}_{y \sim p_d(y)} [\|(y_{t_3} - y_{t_2}) - (\hat{x}_{t_3} - \hat{x}_{t_2})\|_1]. \end{aligned} \quad (10)$$

### 3.2.2. CycleGAN DN

We also propose and investigate an alternative formulation of the difference loss term that relaxes the preservation of the magnitude of the difference vector. We denote this formulation as *CycleGAN DN*.

In this second formulation of  $\mathcal{L}_{diff}$ , we force the mapping functions  $\mathcal{G}$  and  $\mathcal{F}$  to keep the orientation of the difference vectors between the image being adapted and its adapted version unchanged, but preserve the magnitude only up to a scale factor. In this case, Eqs. (8) are replaced by:

$$\begin{aligned} x_{t_1}(p) - x_{t_0}(p) &= a(\hat{y}_{t_1}(p) - \hat{y}_{t_0}(p)) \\ y_{t_3}(q) - y_{t_2}(q) &= b(\hat{x}_{t_3}(q) - \hat{x}_{t_2}(q)), \end{aligned} \quad (11)$$

where  $a$  and  $b$  are scalars. The general formulation for Eqs. (11), which encompasses the difference vectors over all pixels of real and adapted images, becomes:

$$\begin{aligned} x_{t_1} - x_{t_0} &= a(\hat{y}_{t_1} - \hat{y}_{t_0}) \\ y_{t_3} - y_{t_2} &= b(\hat{x}_{t_3} - \hat{x}_{t_2}), \end{aligned} \quad (12)$$

By dividing both sides of Eqs. (12) the average magnitude of the difference vectors of the entire corresponding image we obtain:

$$\begin{aligned} \frac{x_{t_1} - x_{t_0}}{\text{Norm}_{x_{t_1}t_0}} &= \frac{\hat{y}_{t_1} - \hat{y}_{t_0}}{\text{Norm}_{y_{t_1}t_0}} \\ \frac{y_{t_3} - y_{t_2}}{\text{Norm}_{y_{t_3}t_2}} &= \frac{\hat{x}_{t_3} - \hat{x}_{t_2}}{\text{Norm}_{x_{t_3}t_2}}, \end{aligned} \quad (13)$$

where:

$$\text{Norm}_{x_{t_1}t_0} = \frac{1}{H_x W_x} \sum_p \|x_{t_1}(p) - x_{t_0}(p)\|_2 \quad (14)$$

$$\text{Norm}_{y_{t_1}t_0} = \frac{1}{H_y W_y} \sum_p \|\hat{y}_{t_1}(p) - \hat{y}_{t_0}(p)\|_2,$$

$$\text{Norm}_{y_{t_3}t_2} = \frac{1}{H_y W_y} \sum_q \|y_{t_3}(q) - y_{t_2}(q)\|_2 \quad (15)$$

$$\text{Norm}_{x_{t_3}t_2} = \frac{1}{H_x W_x} \sum_q \|\hat{x}_{t_3}(q) - \hat{x}_{t_2}(q)\|_2,$$

In Eqs. (14) to (15),  $\|\cdot\|_2$  denotes the  $L_2$  norm,  $H_x, W_x, H_y$  and  $W_y$  are the height and width of the images  $x_{t_0}$  and  $x_{t_1}$ , and the height and width of the images  $y_{t_2}$  and  $y_{t_3}$ , respectively.

So, the term of the objective function that accounts for the preservation of the change transitions takes the form:

$$\begin{aligned} \mathcal{L}_{diff}(\mathcal{G}, \mathcal{F}) &= \mathbb{E}_{x \sim p_d(x)} \left[ \left\| \frac{x_{t_1} - x_{t_0}}{\text{Norm}_{x_{t_1}t_0}} - \frac{\hat{y}_{t_1} - \hat{y}_{t_0}}{\text{Norm}_{y_{t_1}t_0}} \right\|_2 \right] \\ &+ \mathbb{E}_{y \sim p_d(y)} \left[ \left\| \frac{y_{t_3} - y_{t_2}}{\text{Norm}_{y_{t_3}t_2}} - \frac{\hat{x}_{t_3} - \hat{x}_{t_2}}{\text{Norm}_{x_{t_3}t_2}} \right\|_2 \right], \end{aligned} \quad (16)$$

We remark that in Eqs. (13) we make a data-dependent choice for the otherwise free parameters  $a$  and  $b$  in Eqs. (12), which are replaced by the ratio of the average magnitudes computed over the whole extents of the image pairs. We observe that the norm of the difference vectors' magnitudes can be interpreted as the mean intensity (or brightness) change between two epochs. Thus, when normalizing by the ratio of the mean brightness changes, the model should compensate for some radiometric differences between the domains, e.g., due to lighting conditions or seasonal effects.

## 4. Experiments

The experiments conducted in this work aimed at verifying the effectiveness of the proposed domain adaptation approach in the context

of a particular change detection problem, namely, deforestation detection in tropical forests. The selected domains represent forests of different types, affected by various deforestation practices. Two domains are associated with regions in the Brazilian Legal Amazon (BLA) containing Dense Ombrophyl Forest areas, and Open Ombrophyl Forest areas. The third domain represents a site in the transition between the Brazilian Cerrado and the Amazon Rainforest, which contains Seasonal Deciduous and Semi-Deciduous Forest areas. Six different combinations of the domains were evaluated, considering each of the three areas as source and target domains in the adaptation task. For each of those six scenarios a deep learning-based, fully convolutional classifier was trained with data from the source domain, and evaluated in the original and adapted images from the target domain, considering CycleGAN as a baseline and the alternative implementations of the new regularization loss term.

#### 4.1. Datasets

The image pairs from the BLA cover regions in the Brazilian states of Rondônia (RO) and Pará (PA), and have the following geographic extents: 9°36'51" S - 10°18'35" S latitude, and 62°56'41" W - 64°20'51" W longitude; and 3°08'21" S - 3°26'16" S latitude, and 50°34'04" W - 51°16'12" W longitude, respectively. The site in the Cerrado/Amazon transition zone is located in the Brazilian state of Maranhão (MA), covering an area given by the following coordinates: 4°44'52" S - 5°12'48" S latitude, and 43°37'58" W - 44°01'23" W longitude.

As for the forest typologies, the selected domains represent a gradation that goes from a very dense forest, with little variability of the canopy structure (PA), to a seasonal forest with high canopy variability (MA). The open rainforest (RO) is at an intermediate point between these two formations in terms of canopy variability. Fig. 3 shows the location of the study areas, as well as a RGB composites of the most recent images of the corresponding image pairs. Table 1 shows the acquisition dates and the forest typologies (IBGE, 2012) in the respective domains, plus the number of pixels in each domain labeled as deforestation, no-deforestation, and previous deforestation. As can be seen the class distribution is highly imbalanced.

The images were acquired by the Landsat 8-OLI sensor with 30 m resolution and 7 spectral bands. The images were acquired with minimum cloud cover at dates in the dry season. They have the following dimensions: 2550 × 5120 pixels (RO); 1100 × 2600 pixels (PA); and 1700 × 1440 pixels (MA). All images underwent Level-1 data processing and were downloaded from the Earth Explorer web service from the United States Geological Survey (USGS).<sup>2</sup> In all experiments, the individual image bands were normalized to zero mean and to a variance of one.

The ground truth for deforestation was produced by the PRODES Deforestation Mapping project of the Brazilian National Institute for Space Research (INPE), which, according to (Pinheiro Maurano et al., 2019; Parente et al., 2021), have accuracies of approximately 93% in both the Amazon and Cerrado biomes. The data is freely available at the Terrabrasilis website.<sup>3</sup> We observe that the images that compose the datasets for this study were also used in PRODES for deforestation mapping, for the respective sites and epochs (Almeida et al., 2021). Additionally, as shown in Table 1, we observe that all images were acquired in the months of July and August, in which the acquisition conditions are optimum with respect to cloud coverage. Although it would be interesting to conduct experiments considering different periods of the year, this is not possible because according with PRODES's methodology deforestation mapping in the Amazon is carried out only in the dry season, which happens from late May to September. The reason for that lies in the difficulty of obtaining cloud-free images during the rest of

the year.

Fig. 4 shows the deforestation reference for the respective image pairs (dark orange), which represent the deforestation that happened between the acquisition of the two images. The figure also shows the accumulated deforestation (light gray), which happened between 1988 and the year of acquisition of the first image of the pair.

#### 4.2. Classifier training setup

For the deforestation detection accuracy assessment, we used the Early Fusion strategy adopted in (Daudt et al., 2018b; Andrade et al., 2020; Ortega Adarme et al., 2020) and a fully convolutional classifier based on the U-net architecture (Ronneberger et al., 2015). Following (Ortega Adarme et al., 2020) the image space in each domain was divided into tiles (i.e., large image subsets); specifically, 100 tiles in the RO image, and 15 in the PA and MA images. Approximately 20% of the RO tiles were used to extract training patches/samples (i.e., small image subsets, of sizes equal to the classifier input), 5% of the tiles to extract validation patches, and the remaining 75% for testing. In the case of the PA and MA images, 26% of the tiles were selected for training, 13% for validation, and 60% for testing. Fig. 4 indicates the tiles selected for training by red tile identifiers and those selected for validation by dark green ones in each study area. Tiles without identifiers are in the test sets.

The input to the U-net classifier is a tensor of size 128 × 128 × 14, which represents an image patch. The patches used during training were extracted using a sliding window procedure with an overlap of 98% in each direction. To be selected for training (and validation) at least 2% of the total number of pixels in the patch needed to be labeled as deforestation. During training and testing, image areas with the following characteristics were masked out and thus not further considered in the training and evaluation of the classifier: (i) areas belonging to regions that were affected by deforestation prior to the date of the first image of the image pair (bright green areas in Fig. 4); (ii) areas inside a buffer of a width of two pixels outside the deforestation polygons; (iii) areas inside deforestation polygons smaller than 6.25 ha (equivalent to 69 pixels) in the Amazon sites and in deforestation polygons smaller than 1 ha (equivalent to 11 pixels) for the Cerrado site. The first restriction is applied because the reference data provide no information about change between the two processed epochs in areas that were deforested in earlier years. After deforestation is first detected by the PRODES program, the corresponding regions remain marked as deforestation, regardless of any future change. The second condition aims at avoiding the impact of small inaccuracies in the deforestation reference produced by the rasterization process. As for the third constraint, we adopted the criteria also used in PRODES, because no deforestation polygons smaller than the given thresholds are mapped and, consequently, no small polygons are available in the reference data.

Data augmentation was applied to all extracted patches: a 90° rotation, as well as vertical and horizontal flips. Additionally, aiming at alleviating the highly imbalanced distribution of deforestation and no-deforestation pixels, we adopted a *weighted cross-entropy* cost function to train the U-net. The weights were determined empirically, and set to 2 and 0.4 for the deforestation and no-deforestation class, respectively.

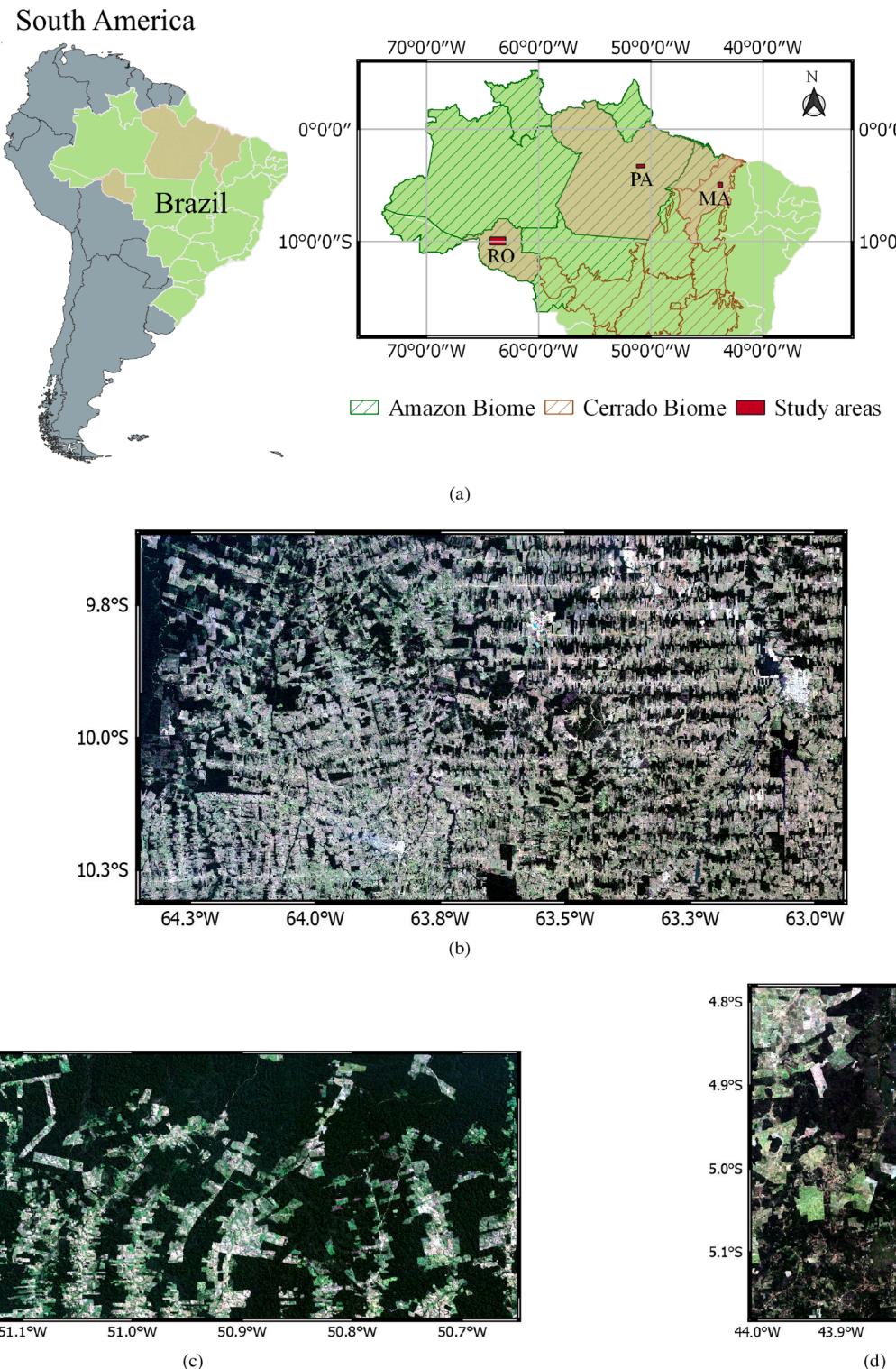
During training, the cost function was minimized using the Adam optimizer (Kingma and Ba, 2014), with a learning rate  $\gamma$  and momentum  $\beta_1$  equal to 0.0001 and 0.9, respectively. The batch size was 32, and the early stopping procedure was used to avoid over-fitting. The patience parameter, which controls the number of epochs without improvements in the validation loss, was set to 10. The classifier was executed 10 times, each time with a different (random) initialization of the trainable parameters.

#### 4.3. Domain adaptation training setup

To train the CycleGAN-based image translation variants, patches of

<sup>2</sup> <https://earthexplorer.usgs.gov/>.

<sup>3</sup> <http://terrabrasilis.dpi.inpe.br/map/deforestation>.



**Fig. 3.** Visual representation and localization of each domain used in the experiments carried out in this work. (a) Geographical localization of the respective sites. True color composites of the images covering the sites, corresponding to the acquisition date 2017 (b) Rondônia (RO), (c) Pará (PA), and (d) Maranhão (MA).

256 × 256 pixels were used, extracted from the entire extents of the images in both domains. The patches were extracted using a sliding window procedure with strides equal to 50, 21, and 19, for the RO, PA, and MA images, respectively. We randomly shuffled the samples in each domain to produce unpaired training samples in each pair of domains. Moreover, following the same scheme applied in (Zhu et al., 2017a), each training pair of samples was resized to 286 × 286, randomly

cropped to 256 × 256, and randomly flipped.

Additionally, the batch size was set to 1, and the Adam optimizer (Kingma and Ba, 2014) was used, with learning rate  $\gamma$  and momentum  $\beta_1$  set to 0.002 and 0.5, respectively. The coefficient  $\lambda_{cyc}$  of the cycle consistency loss term was set to 10;  $\lambda_{idt}$  of the identity loss term to 5; and  $\lambda_{diff}$  of the difference loss term to 10. The models were trained for 200 epochs, employing a linear learning rate decay from the 100<sup>th</sup> epoch,

**Table 1**

Characteristics of the different domains: vegetation typology; image acquisition dates; and number of pixels labeled as deforestation, no-deforestation, or covering previous deforested areas. Values inside parentheses represent the proportions of pixels of each class, for the respective sites.

Domains	Rondônia (RO)	Pará (PA)	Maranhão (MA)
Vegetation	Open Ombrophyl Forest	Dense Ombrophyl Forest	Seasonal Deciduous and Semi-Deciduous Forest
Date 1	July 18, 2016	August 2, 2016	August 18, 2017
Date 2	July 21, 2017	July 20, 2017	August 21, 2018
# of pixels labeled as deforestation	225 635 (3%)	82 970 (3%)	71 265 (3%)
# of pixels labeled as no-deforestation	3 816 981 (29%)	1 867 929 (65%)	1 389 844 (57%)
# of pixels covering previous deforestation	9 013 384 (69%)	903 901 (32%)	986 891 (40%)

also following (Zhu et al., 2017a).

#### 4.4. Network architectures

The network architecture of the U-net classifier, CycleGAN generators, generators' Resnet blocks, and discriminators are described in Tables 2–4, respectively. In the tables, the operations performed at each layer are described as convolution ( $C$ ), transposed convolution ( $D$ ), instance normalization ( $I$ ), ReLU ( $R$ ), Leaky ReLU ( $L$ ), MaxPooling ( $M$ ), and skip connections [:]. The values in parenthesis after the operation types correspond to the number of filters, filter dimensions and stride, for convolutional layers. In the case of MaxPooling, they correspond to the kernel dimension and stride. For Reflection Padding, the values indicate the number of rows and columns to be reflected. Following (Zhu et al., 2017a), the CycleGAN model uses instance normalization (Ulyanov et al., 2017) instead of batch normalization. While in batch normalization all image patches/samples are considered in the normalization procedure, which is more appropriate for classification problems, in instance normalization each patch is normalized individually.

The discriminators, described in Table 4, follow the architecture proposed in (Zhu et al., 2017a), namely, a  $70 \times 70$  PatchGAN (Isola et al., 2017). The parameter  $\alpha$  in the Leaky ReLU activation function was set to 0.2. The CycleGAN model was implemented in Pytorch,<sup>4</sup> while the U-net classifier was implemented with Tensorflow.<sup>5</sup> We conducted all experiments on an NVIDIA Titan XP GPU with 11 GB of RAM.

#### 4.5. Metrics

The performance of the U-net classifier in all scenarios is expressed in terms of average F1-score (F1) and mean average precision (mAP) considering the positive (deforestation) class obtained in each classifier trial (See Section 4.2). Specifically, the F1-score is expressed by the harmonic mean of Precision and Recall as follows:

$$F1\text{-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (17)$$

where

$$\begin{aligned} \text{Precision} &= \frac{t_p}{t_p + f_p} \\ \text{Recall} &= \frac{t_p}{t_p + f_n}. \end{aligned} \quad (18)$$

<sup>4</sup> <https://pytorch.org/>.

<sup>5</sup> <https://www.tensorflow.org/>.

In Eq. (18),  $t_p$  is the number of pixels correctly assigned to the deforestation class (true positives),  $f_p$  represents the number of pixels erroneously classified as deforestation (false positives). Similarly,  $f_n$  corresponds to the number of pixels incorrectly classified as no-deforestation (false negatives).

The mAP metrics represents the area under the curve obtained when computing pairs of Precision and Recall values for different classification thresholds in the range of 0 to 1, over the average of probability maps delivered in each classifier execution.

#### 5. Results and discussion

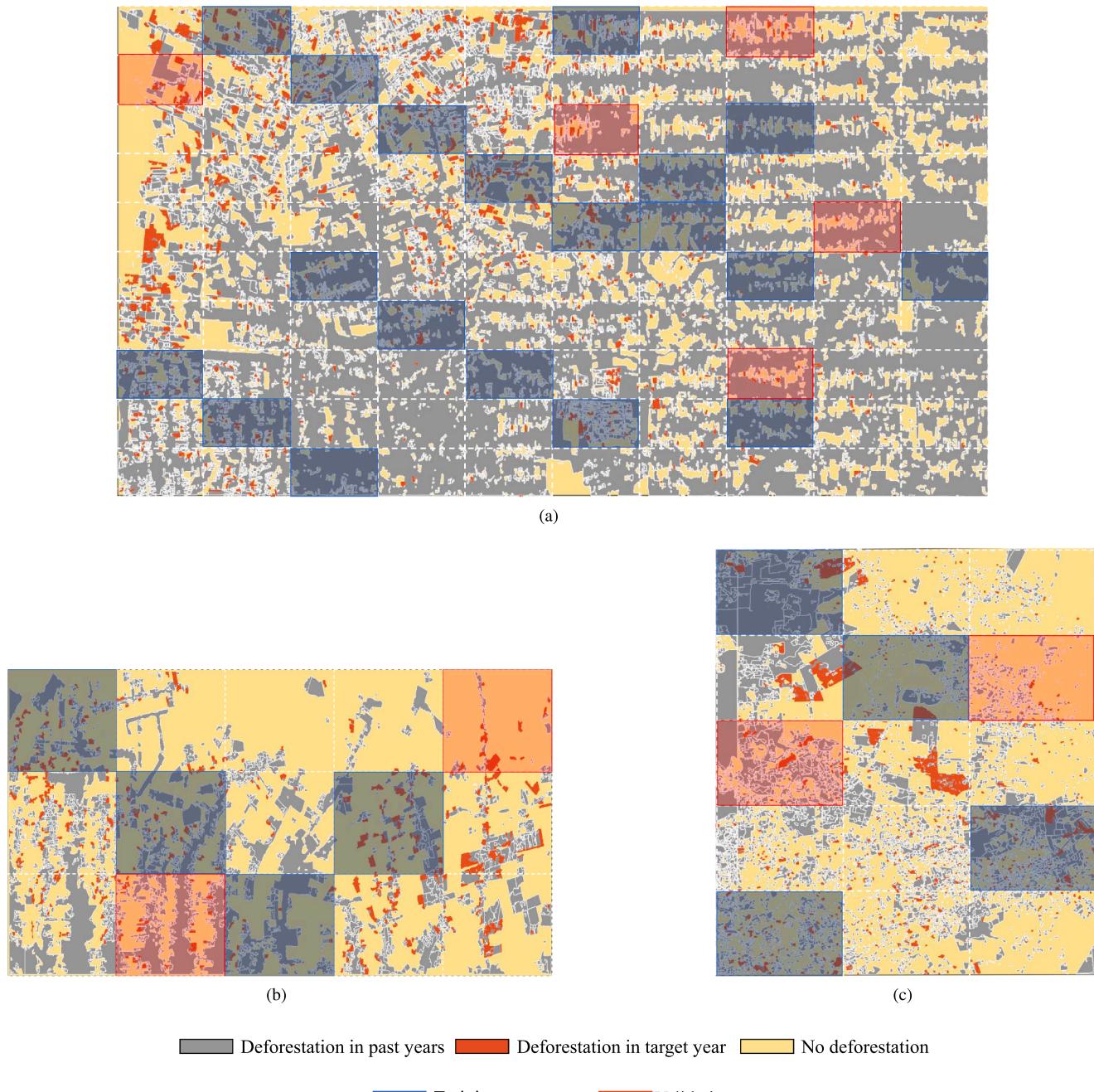
Fig. 5 shows the U-net classifier results for the different domain adaptation combinations. The plots show precision versus recall curves. In the figure,  $X$  corresponds to the source domain,  $Y$  to the target domain, and  $Y \rightarrow X$  represents the target domain adapted to the source domain. The different curves in each graph correspond to five classification schemes: in the first two schemes no adaptation was performed, the other three schemes refer to domain adaptation using the different variants of the image translation model. The first scheme, [Train( $Y$ )|Test( $Y$ )] (dark blue curves), corresponds to a situation where the source and target domains are the same, i.e., where no adaptation is needed. In the second scheme, [Train( $X$ )|Test( $Y$ )] (light blue curves), regarded as the baseline, the classifier is trained with the image pair and deforestation references from the source domain, and evaluated on the target domain data, with no adaptation. The other three schemes correspond to the classifier trained with source domain data and evaluated on the target domain images adapted to the source domain [Train( $X$ )|Test( $Y \rightarrow X$ )]. We considered three variants of the image translation model: original CycleGAN (green curves), CycleGAN D (red curves), and CycleGAN DN (orange curves). CycleGAN D corresponds to a variant that uses the formulation of the *difference loss* given by Eq. (10), while CycleGAN DN corresponds to the formulation given by Eq. (16).

Tables 5 and 6 show the mean average precisions (mAP), the F1 scores (F1), and the precision and recall values obtained in the experiments. The No-Domain Adaptation rows show the results associated with the first two classification schemes described in the previous paragraph. The other values represent the performance obtained with all possible domain adaptation configurations, considering the three different domains. The values in parenthesis represent the improvements achieved with the image translation variants relative to the second scheme, which involves different domains with no adaptation. In the No-Domain Adaptation rows, the values in bold represent the upper-bound for domain adaptation. The values in bold in the other rows in Table 5 represent the best domain adaptation configuration, considering the (target) domain associated with the respective table column.

As expected, high classification accuracies were obtained when the U-net classifier was trained and tested on the same domain, and significant drops in performance were observed when the classifier was evaluated on a domain different than the one used for training. However, in the majority of the cases, the results show that the proposed domain adaptation approaches brought notable performance improvements in the cross-domain classification scheme.

By inspecting Tables 5 and 6, and Fig. 5(f), it can be observed that the highest absolute accuracies in the cross-domain evaluations are associated with the [Train(MA)|Test(PA)] domain combination. Moreover, the results reveal that the U-net trained on the PA domain generalizes worse when it is tested on the RO or MA domains (observe the No-Domain Adaptation row in Table 5, and the light blue curves in Fig. 5). To find a reason for this behavior, we examined the complexity of the deforestation and non-deforestation classes in each domain.

For this purpose, we adopted the number of clusters in the difference images as an indicator of class complexity by computing the pixel-wise spectral differences between the images that compose the image pair of each domain. In this analysis we computed the optimal number of clusters  $k$  considering all pixel locations associated with the



**Fig. 4.** Distribution of image tiles for training, validation and testing in the respective study areas: (a) Rondônia (RO); (b) Pará (PA); and (c) Maranhão (MA). Please note that the tiles that are not shaded correspond to the ones selected for testing. The figure also shows the polygons associated with the deforestation that occurred during the image acquisition dates of the respective domains, the polygons associated with the deforestation that occurred prior to the date of the first image of the respective image pairs, and the areas labeled as not deforested.

deforestation and no-deforestation classes in each domain, as shown in Table 7. To determine  $k$  we executed the k-means algorithm together with the Calinski-Harabasz criterion (Calinski and Harabasz, 1974) for different numbers of clusters, and for each difference image. Pixels located in areas affected by deforestation prior to the first date of each image pair, or lying within the buffer around the deforestation polygons were not included in the analysis, as they were not considered in the classification, either (see Section 4.2). The number of clusters is related to the diversity of changes in the respective domains. It indicates the complexity of the landscape associated with deforestation and non-deforestation (forest) regions.

Accordingly, the complexity of forested regions in PA and MA is

minimum and maximum, respectively, among the three domains. These numbers are consistent with the complexity of the canopies of the respective forest typologies, which in PA is related to the forest homogeneity in that site. The higher complexity in MA is a consequence of the more significant inter-annual variability of forests. As longer and more marked dry periods can be present, the seasonal forest can suffer leaf loss that is not related to deforestation, thus increasing the forest variability. This may explain why the classifier trained with data from the MA domain delivers better results in the cross evaluation schemes in comparison to the one trained on PA data. We hypothesize that the classifier trained on MA is more efficient in discerning changes that are not associated with deforestation.

**Table 2**

U-net and CycleGAN's generator architecture details. At each layer, operations are represented by the following symbols: convolution (C); transpose convolution (D); instance normalization (I); max-pooling (M); ReLU (R); and skip-connections [:]. The values in parenthesis after the operation types correspond to the number of filters, filter dimensions and stride. In the case of MaxPooling, they correspond to the kernel dimension and stride. For Reflection Padding, the values indicate the number of rows and columns to be reflected in each dimension.

	U-net		CycleGAN's generator		
	Layer name	Layer structure	Output shape	Layer structure	Output shape
Encoder		Input	(128, 128, 14)	Input	(256, 256, 14)
	$E_1$	$CR(32, 3, 1)$	(128, 32)	Reflection Padding (3, 3)	(262, 262, 14)
	$E_2$	$M(2, 2)$	(64, 64, 32)	$CIR(64, 7, 1)$	(256, 256, 64)
	$E_3$	$CR(64, 3, 1)$	(64, 64, 64)	$CIR(128, 3, 2)$	(128, 128, 128)
	$E_4$	$M(2, 2)$	(32, 32, 64)	$CIR(256, 3, 2)$	(64, 64, 256)
	$E_5$	$CR(128, 3, 1)$	(32, 32, 128)	ResNet block 1	(64, 64, 256)
	$E_6$	$M(2, 2)$	(16, 16, 128)	:	:
	$E_7$	$CR(256, 3, 1)$	(16, 16, 256)	ResNet block 9	(64, 64, 256)
	$E_8$	$M(2, 2)$	(8, 8, 256)		
	$E_9$	$CR(512, 3, 1)$	(8, 8, 512)		
Decoder	$U_1$	$DR(256, 3, 2)$	(16, 16, 256)	$DIR(128, 3, 2)$	(128, 128, 128)
	$U_2$	$[U_1 : E_7]$	(16, 16, 512)	$DIR(64, 3, 2)$	(256, 256, 64)
	$U_3$	$DR(128, 3, 2)$	(32, 32, 128)	Reflection Padding (3, 3)	(262, 262, 64)
	$U_4$	$[U_3 : E_5]$	(32, 32, 256)	$C(14, 7, 1)$	(256, 256, 14)
	$U_5$	$DR(64, 3, 2)$	(64, 64, 64)		
	$U_6$	$[U_5 : E_3]$	(64, 64, 128)		
	$U_7$	$DR(32, 3, 2)$	(128, 128, 32)		
	$U_8$	$[U_7 : E_1]$	(128, 128, 64)		
	$U_9$	$C(2, 1, 1)$	(128, 128, 2)		
	O	Softmax	(128, 128, 2)		

**Table 3**

Generator's ResNet block architecture details. At each layer operations are represented by the following symbols: convolution (C); instance normalization (I); and ReLU (R). The values in parenthesis after the operation types correspond to the number of filters, filter dimensions and stride. For Reflection Padding, the values indicate the number of rows and columns to be reflected in each dimension.

Layer name	Layer Structure
$O_1$	Input Reflection Padding(1,1)
$O_2$	$CIR(64, 3, 1)$
$O_3$	Reflection Padding(1,1)
$O_4$	$CI(64, 3, 1)$
O	Input + $O_4$

**Table 4**

CycleGAN discriminator architecture details. At each layer operations are represented by the following symbols: convolution (C); instance normalization (I); and LeakyReLU (L). The values in parenthesis after the operation types correspond to the number of filters, filter dimensions and stride.

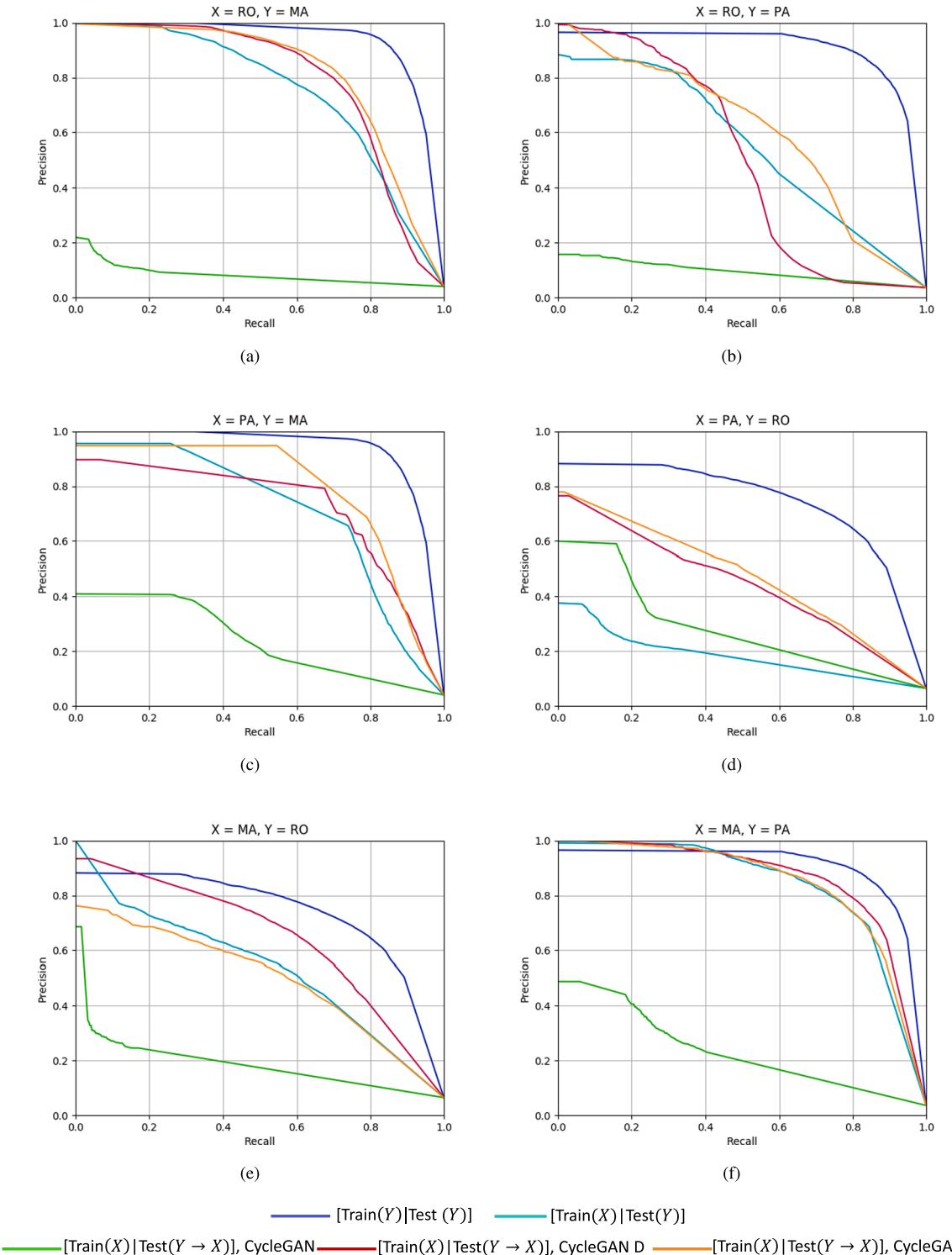
Layer Structure	Output shape
Input	(256, 256, 14)
$CL(64, 4, 2)$	(128, 128, 64)
$CIL(128, 4, 2)$	(64, 64, 128)
$CIL(256, 4, 2)$	(32, 32, 256)
$CIL(512, 4, 1)$	(31, 31, 512)
$C(1, 4, 1)$	(30, 30, 1)

Following the same line of thought, Table 7 indicates how the original forest areas have a more complex structure/density on the PA site, followed by RO, and finally by MA, although the differences are less pronounced. Different forest densities can lead to different degrees of difficulty in carrying out the deforestation process. The process consists of cutting the trees, leaving the deadwood on the ground to dry, and burning the remaining debris until the terrain is clear. The cleaning process can take several years. Our hypothesis is that in areas with lower forest density (such as MA), the process of cleaning the area is easier and faster, resulting in more uniform deforestation footprints. In the opposite situation, as found in the case of the PA site, the higher forest density can lead to incomplete deforestation processes with the presence of debris that will possibly take various years to disappear.

Nevertheless, the comparatively higher deforestation diversity in PA could lead the classifier trained with data from that site to misclassifying changes in other domains that are not associated with deforestation, thus generating a higher number of false positives, which is expressed by the lower precision values presented in Table 6. Similarly, the highest diversity of forested areas in MA may lead the classifier trained on that domain to misclassify changes associated with deforestation in the other domains. Referring to Table 6, this is expressed by the lower recall values in the No-Domain Adaptation row. On the other hand, the classifier trained on RO reached performances that are generally in between the ones obtained with the classifiers trained on PA and MA, which can be explained by the fact that the number of deforestation and no-deforestation clusters found in RO are in the middle of the other domains.

Regarding the classification results obtained by the adaptation approaches using the regularization constraints proposed in this work, i.e., CycleGAN D and CycleGAN DN, the new approach performed consistently better than the plain CycleGAN model. The proposed variants also outperformed the [Train(X)|Test(Y)] baseline classification in most cases. In terms of mAP (Table 5), the highest improvements were recorded when evaluating the RO image pair adapted to PA, i.e., [Train (PA)|Test(RO→PA)], which were of + 26.1% and + 28.8%, respectively for the CycleGAN D and CycleGAN DN variants. The lowest average improvements were associated with the [Train(MA)|Test (PA→MA)] adaptation (+2.5% and +0.6%), which is not surprising because in this combination, the gap between the upper-bound and baseline classifications was also narrow, limiting the scope for possible improvement (see Fig. 5(f)). Both variants of the proposed domain adaptation method favored the precision of the classifier trained in PA and tested in the RO images adapted for PA (RO→PA), and MA to PA (MA→PA). At the same time, classifiers trained in RO and MA achieved an increase in recall when evaluated on PA images adapted to these domains. Low precision for PA and low recall for RO and MA were the major drawbacks of such classifiers before the domain adaptation (see Table 6).

Additionally, the proposed image adaptation variants led to a negative transfer in [Train(RO)|Test(PA→RO)] (-5.2% for CycleGAN D) and [Train(MA)|Test(RO→MA)] (-3.6% for CycleGAN DN), in terms of



**Fig. 5.** U-net mean classification results for each domain adaptation combination. (a)-(e) RO  $\leftrightarrow$  MA, (b)-(d) RO  $\leftrightarrow$  PA, (c)-(f) PA  $\leftrightarrow$  MA. In the figure, X corresponds to the source domain, Y to the target domain, and Y-X to the target domain adapted to the source domain. Each plot shows precision versus recall curves for five classification schemes. The dark blue curves correspond to the scheme [Train(Y)|Test(Y)]. The light blue curves correspond to [Train(X)|Test(Y)]. The other three curves correspond to the scheme [Train(X)|Test(Y→X)], for three adaptation models: CycleGAN (green curves), CycleGAN D (red curves), and CycleGAN DN (orange curves).

mAP. However, when the translation was carried out in the opposite direction, it yielded substantial improvements of + 26.1% and + 6.2%, respectively. This asymmetry is probably related to the issues previously raised regarding the different patterns in deforested and non-deforested areas among the study sites.

Moreover, in general the CycleGAN DN variant produced better re-

sults than the CycleGAN D variant. In terms of mAP and F1-Scores, the CycleGAN DN variant outperformed its counterparts in four out of the six domain combinations, whereas CycleGAN D was better in two combinations: [Train(MA)|Test(PA→MA)] and [Train(MA)|Test(RO→MA)] for mAP. In the first case, however, the accuracy difference was relatively small (86.1% vs. 84.2%), and the F1-Score associated

**Table 5**

U-net average performance in terms of F1-Scores (F1) and mean average precisions (mAP) (%). The columns with the accuracy values correspond to the domains the classifier was evaluated on, and the rows correspond to the domains the U-net was trained on. Each group of rows represents a domain adaptation model (including no adaptation). Values in bold represent the highest accuracies for the respective domain combinations.

DA Models	Domains	Evaluating on:						
		RO		PA		MA		
		F1	mAP	F1	mAP	F1	mAP	
No-Domain Adaptation	Training on:	RO	<b>67.0</b>	<b>74.3</b>	35.5	55.3	60.3	75.0
		PA	20.5	18.4	<b>83.2</b>	<b>90.4</b>	41.7	71.2
		MA	47.9	53.6	70.7	83.6	<b>85.5</b>	<b>93.7</b>
CycleGAN	CycleGAN	RO	—	—	14.6(–20.9)	9.6(–45.7)	10.7(–49.6)	8.3(–66.7)
		PA	27.8(+7.3)	28.4(+10.0)	—	—	32.2(–9.4)	24.1(–47.1)
		MA	12.4(–35.5)	18.8(–34.8)	28.8(–41.9)	23.5(–59.8)	—	—
CycleGAN D	CycleGAN D	RO	—	—	34.0(–1.5)	50.1(–5.2)	58.3(–2.0)	79.2(+4.1)
		PA	46.6(+26.1)	44.5(+26.1)	—	—	<b>52.0(+10.3)</b>	72.0(+1.0)
		MA	<b>61.0(+13.1)</b>	<b>64.7(+11.1)</b>	70.8(+0.1)	<b>86.1(+2.5)</b>	—	—
CycleGAN DN	CycleGAN DN	RO	—	—	51.0(+15.0)	<b>60.2(+4.9)</b>	<b>67.0(+6.7)</b>	<b>81.2(+6.2)</b>
		PA	<b>48.0(+27.5)</b>	<b>47.2(+28.8)</b>	—	—	48.0(+6.3)	<b>79.2(+8.0)</b>
		MA	45.4(–2.5)	50.0(–3.6)	<b>72.0(+1.3)</b>	84.2(+0.6)	—	—

**Table 6**

U-net average performance in terms of Precision and Recall in (%). The columns with the accuracy values correspond to the domains the classifier was evaluated on, and the rows correspond to the domains the U-net was trained on. Each group of rows represents a domain adaptation model (including no adaptation). Values in bold represent the highest accuracies for the respective domain combinations.

DA Models	Domains	Evaluating on:						
		RO		PA		MA		
		Precision	Recall	Precision	Recall	Precision	Recall	
No-Domain Adaptation	Training on:	RO	<b>70.5</b>	<b>64</b>	74.7	23.6	57.7	60.3
		PA	24.4	17.8	<b>84.2</b>	<b>82.3</b>	27.9	84.3
		MA	59.1	40.7	85.8	60.3	<b>85.3</b>	<b>86.4</b>
CycleGAN	CycleGAN	RO	—	—	13.7(–61)	15.8(–7.8)	12.3(–45.4)	9.6(–50.6)
		PA	43.6(+19.2)	20.5(+2.7)	—	—	26.4(–1.5)	41.6(–42.7)
		MA	28.1(–31)	8.0(–32.7)	33.1(–52.7)	25.5(–34.8)	—	—
CycleGAN D	CycleGAN D	RO	—	—	53.8(–20.9)	33.2(+9.6)	67.6(+9.9)	55.6(–4.9)
		PA	39.6(+15.2)	56.9(+39.1)	—	—	38.0(+10.1)	84.2(–0.1)
		MA	57.7(–1.4)	64.6(+23.9)	85.6(–0.2)	62.6(+1.9)	—	—
CycleGAN DN	CycleGAN DN	RO	—	—	55.8(–18.9)	48.5(+24.9)	68.7(+11)	66.1(+6.2)
		PA	39.2(+14.8)	62.0(+44.2)	—	—	33.4(+7)	88.2(+3.9)
		MA	55.0(–4.1)	40.0(–0.7)	82.2(–3.6)	65.0(+2.7)	—	—

**Table 7**

Optimal number of clusters  $k$  found within the deforestation and no-deforestation classes, using the k-means algorithm together with the Calinski-Harabasz criterion.

Domain	Class	Optimal $k$
PA	Deforestation	4
	No-deforestation	2
RO	Deforestation	3
	No-deforestation	3
MA	Deforestation	2
	No-deforestation	10

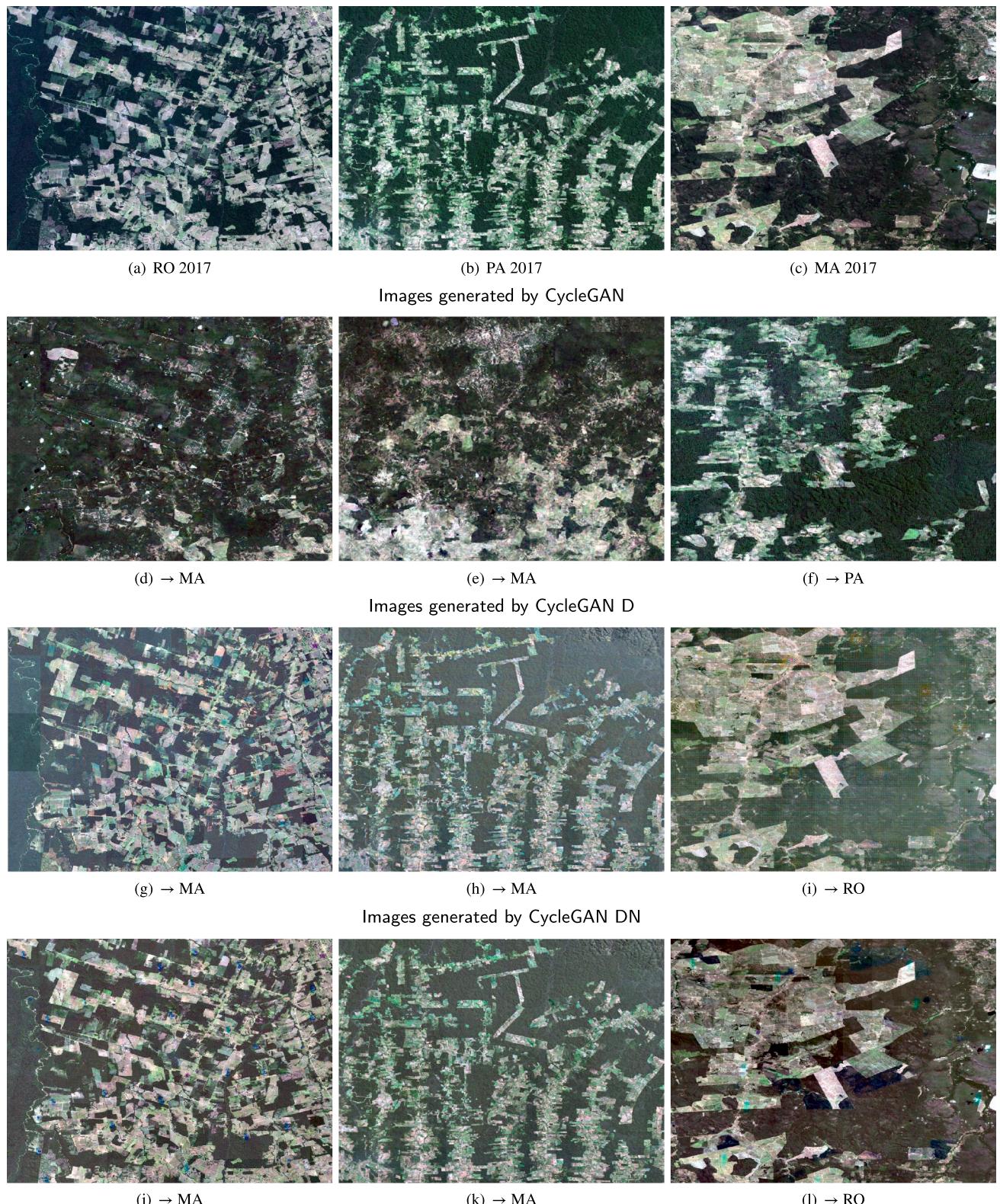
with the CycleGAN DN was higher (72.0% against 70.8%). In the second case, the accuracy difference was higher (64.7% vs. 50.0%), which is a result that deserves further investigation.

We conclude the discussion with an analysis of the images generated with the image translation model variants. Fig. 6 shows subsets of images from the year 2017, which constitute real and adapted image pairs. The image subsets on the top represent the three domains prior to adaptation: RO (Fig. 6(a)); PA (Fig. 6(b)); and MA (Fig. 6(c)). The subsets shown in the following rows show the outcome of the image translation variant models: original CycleGAN (Figs. 6(d)(e)(f)); CycleGAN D (Figs. 6(g)(h)(i)); and CycleGAN DN (Figs. 6(j)(k)(l)). Each column corresponds to one domain.

Observing the images generated with the original CycleGAN, it is clear that the model fails to preserve most of the structural character-

istics of the original domain in the adapted images. The images generated with that model indeed incorporate the style of the respective target domains, but also the class distributions of those domains, which is detrimental to the change detection task, as the classification accuracies have shown. For instance, deforestation areas have been created in standing forest regions, in the MA→PA adaptation (Fig. 6(f)). The opposite effect can be seen in RO→MA (Fig. 6(d)) and PA→MA (Fig. 6(e)), i.e., deforested regions have been reforested. However, as already mentioned, an very good translation capacity can be noted in the adapted images, which show colors and structures that match the ones in the respective target domains. For example, clouds and corresponding shadows, which are originally present only in the MA domain (Fig. 6(c)), have been inserted in the images adapted from the other domains (Figs. 6(d) and 6(e)).

The images generated by the proposed translation model variants, however, seem to better preserve object structures and class labels from the original domain. That can evidently be credited to the proposed regularization constraint (Eqs. (10) and (16)). Both variants also seem to successfully transfer the style of the target domain to the adapted images, but it can be noted that the CycleGAN DN variant, which uses the implementation of the constraint given by Eq. (16), is able to generate images with colors that better match the ones in the target domain. This behaviour seems to indicate that limiting the constraint to preserve the orientation of the difference vectors in the image to be adapted makes the model better adapt to the color intensities originally present in the target domain.



**Fig. 6.** Examples of real and generated images in each domain. The image subsets on the top represent the three domains prior to adaptation: (a) Rondônia (RO), (b) Pará (PA), (c) Maranhão (MA). The subsets shown in the following rows show the outcomes of the image translation variant models: (d)(e)(f) original CycleGAN, (g)(h)(i) CycleGAN D, (j)(k)(l) CycleGAN DN. Each column represents one domain, and the caption below each image subset indicates the respective target domain.

## 6. Conclusions

In this work we proposed a cycle-consistent image translation approach for unsupervised domain adaptation, designed for change

detection. The approach is an extension of CycleGAN, in which we modify the original objective function, so that the adaptation preserves the changes contained in the images being adapted.

The approach was evaluated on a particular remote sensing

application, namely, deforestation detection in tropical biomes. The experiments showed that the proposed approach was generally successful in preserving the changes through the domain adaptation process, while transferring the style of the source domain images to the adapted ones. The devised method enabled a classifier trained solely on source domain data to properly perform change detection on adapted target domain images, thus eliminating the cost of gathering labeled data for the target domains. Additionally, the proposed domain adaptation approach is unsupervised, as the image translation process requires no labeled samples from the source or target domains.

The experiments demonstrated that the accuracies obtained with the proposed approach depend considerably on the data distribution of the respective domains. The mere inversion of the role between source and target domains can substantially impact the results. The issues underlying such behavior demand further investigation.

In any case, the new regularization terms added to the CycleGAN objective function significantly reduced the artifacts that are commonly generated in CycleGAN results and brought accuracy improvements for the classifier in 9.1% and 7.1% according mAP and F1 respectively. The proposed approach does not distinguish between different types of changes present in the datasets, it rather treats all of them in the same way. We conjecture that an approach capable of considering the different kinds of change occurring in the source and target domains may yield better results – this is a line of investigation worth being explored in the continuation of this research.

It is also worth mentioning that, although we conducted the study with a focus on detecting deforestation in tropical forests, the proposed approach is not bound to a particular type of change. Therefore, it is reasonable to expect that the methods introduced in the presented work will also perform well in other change detection applications, opening up another direction for future investigation.

Finally, in this work we concentrated on the design and implementation of the domain adaptation process, and took as a quality metric the accuracy of a standard U-Net classifier. The experimental results suggest that there might be room to improve the generalization capacity of the classifier model. This could be achieved by better exploiting the training data, e.g., with additional data augmentation techniques, or, which we think is more promising, simplifying the classifier in terms of the number of parameters, thus reducing the risk of overfitting. Such issues will be addressed in the continuation of this research.

### Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Acknowledgements

This work was supported by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Deutsche Akademische Austauschdienst (DAAD), Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ), and NVIDIA Corporation.

### References

- Afify, H.A., 2011. Evaluation of change detection techniques for monitoring land-cover changes: A case study in new burg el-arab area. *Alexandria Eng. J.* 50, 187–195.
- Alexakis, E.B., Armenakis, C., 2020. Evaluation of unet and unet++ architectures in high resolution image change detection applications. *Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci.* 43, 1507–1514.
- Almeida, C.A., Maurano, L.E.P., Valeriano, D.D.M., Camara, G., Vinhas, L., Gomes, A.R., Monteiro, A.M.V., Souza, A.A.A., Renno, C.D., Silva, D.E., Adami, M., Escada, M.I.S., Mota, M., Kampel, S.A., 2021. Methodology for Forest Monitoring used in PRODES and DETER projects. Technical Report. INPE, São José dos Campos. <http://urlib.net/rep/8JMKD3MP3W34R/443H3RE>.
- Andrade, R., Costa, G., Mota, G., Ortega, M., Feitosa, R., Soto, P., Heipke, C., 2020. Evaluation of semantic segmentation methods for deforestation detection in the amazon. *Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci.* 43, 1497–1505.
- Asokan, A., Anitha, J., 2019. Change detection techniques for remote sensing applications: a survey. *Earth Sci. Inf.* 12, 143–160.
- Benjdira, B., Bazi, Y., Koubaa, A., Ouni, K., 2019. Unsupervised domain adaptation using generative adversarial networks for semantic segmentation of aerial images. *Remote Sens.* 11, 1369.
- Bruzzone, L., Prieto, D.F., 2000. Automatic analysis of the difference image for unsupervised change detection. *IEEE Trans. Geosci. Remote Sens.* 38, 1171–1182.
- Caliński, T., Harabasz, J., 1974. A dendrite method for cluster analysis. *Commun. Statistics-theory Methods* 3, 1–27.
- Celik, T., Ma, K.K., 2010. Multitemporal image change detection using undecimated discrete wavelet transform and active contours. *IEEE Trans. Geosci. Remote Sens.* 49, 706–716.
- Chen, H., Wu, C., Du, B., Zhang, L., 2020. Dsdanet: Deep siamese domain adaptation convolutional neural network for cross-domain change detection. arXiv preprint arXiv:2006.09225.
- Chopra, S., Hadsell, R., LeCun, Y., 2005. Learning a similarity metric discriminatively, with application to face verification. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). IEEE, pp. 539–546.
- Chu, Y., Cao, G., Hayat, H., 2016. Change detection of remote sensing image based on deep neural networks. In: 2016 2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE 2016). Atlantis Press.
- Daudt, R.C., Le Saux, B., Boulch, A., 2018a. Fully convolutional siamese networks for change detection. In: 2018 25th IEEE International Conference on Image Processing (ICIP). IEEE, pp. 4063–4067.
- Daudt, R.C., Le Saux, B., Boulch, A., Gousseau, Y., 2018b. Urban change detection for multispectral earth observation using convolutional neural networks. In: IGARSS 2018–2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 2115–2118.
- De Sy, V., Herold, M., Achard, F., Beuchle, R., Clevers, J., Lindquist, E., Verchot, L., 2015. Land use patterns and related carbon losses following deforestation in south america. *Environ. Res. Lett.* 10.
- Deng, X., Yang, H.L., Makkar, N., Lunga, D., 2019. Large scale unsupervised domain adaptation of segmentation networks with adversarial learning. In: IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 4955–4958.
- Dhillon, I.S., Tropp, J.A., 2008. Matrix nearness problems with bregman divergences. *SIAM J. Matrix Anal. Appl.* 29, 1120–1146.
- Elshamli, A., Taylor, G.W., Berg, A., Areibi, S., 2017. Domain adaptation using representation learning for the classification of remote sensing images. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* 10, 4198–4209.
- Ganin, Y., Lempitsky, V., 2015. Unsupervised domain adaptation by backpropagation. In: International conference on machine learning, PMLR. pp. 1180–1189.
- Gholami, B., Sahu, P., Rudovic, O., Bousmalis, K., Pavlovic, V., 2020. Unsupervised multi-target domain adaptation: An information theoretic approach. *IEEE Trans. Image Process.* 29, 3993–4002.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. In: Advances in neural information processing systems, pp. 2672–2680.
- Goodman, R., Aramburu, M., Gopalakrishna, T., Putz, F., Gutiérrez, N., Alvarez, J., Aguilera-Amuchastegui, N., Ellis, P., 2019. Carbon emissions and potential emissions reductions from low-intensity selective logging in southwestern amazonia. *For. Ecol. Manage.* 439, 18–27. <https://doi.org/10.1016/j.foreco.2019.02.037>.
- Gu, W., Lv, Z., Hao, M., 2017. Change detection method for remote sensing images based on an improved markov random field. *Multimedia Tools Appl.* 76, 17719–17734.
- Han, T., Wulder, M.A., White, J.C., Coops, N.C., Alvarez, M., Butson, C., 2007. An efficient protocol to process landsat images for change detection with tasseled cap transformation. *IEEE Geosci. Remote Sens. Lett.* 4, 147–151.
- Hinton, G.E., 2009. Deep belief networks. *Scholarpedia* 4, 5947.
- Hoffman, J., Tzeng, E., Park, T., Zhu, J.Y., Isola, P., Saenko, K., Efros, A.A., Darrell, T., 2017. Cycada: Cycle-consistent adversarial domain adaptation. arXiv preprint arXiv: 1711.03213.
- Hu, F., Xia, G., Hu, J., Zhang, L., 2015. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sens.* 7, 14680–14707.
- IBGE, 2012. Manual Técnico da Vegetação Brasileira. <https://www.ibge.gov.br/geociencias/downloads-geociencias.html>.
- Isola, P., Zhu, J., Zhou, T., Efros, A.A., 2017. Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1125–1134.
- Kauth, R.J., Thomas, G., 1976. The tasseled cap—a graphic description of the spectral-temporal development of agricultural crops as seen by landsat. In: LARS symposia, p. 159.
- Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- Koller, D., Friedman, N., 2009. Probabilistic graphical models: principles and techniques. MIT Press.
- Lafferty, J., McCallum, A., Pereira, F.C., 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: International conference on machine learning, pp. 282–289.
- LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature* 521, 436.
- Li, L., Wang, C., Zhang, H., Zhang, B., 2019. Residual unet for urban building change detection with sentinel-1 sar data. In: IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 1498–1501.

- Li, S.Z., 2009. Markov random field modeling in image analysis. Springer Science & Business Media.
- Liu, S., Bruzzone, L., Bovolo, F., Zanetti, M., Du, P., 2015. Sequential spectral change vector analysis for iteratively discovering and detecting multiple changes in hyperspectral images. *IEEE Trans. Geosci. Remote Sens.* 53, 4363–4378.
- Malila, W.A., 1980. Change vector analysis: an approach for detecting forest changes with landsat. In: LARS symposia, p. 385.
- Malingreau, J., Eva, H., de Miranda, E., 2012. Brazilian amazon: a significant five year drop in deforestation rates but figures are on the rise again. *Ambio* 41, 309–314. <https://doi.org/10.1007/s13280-011-0196-7>.
- Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S., 2017. Least squares generative adversarial networks. In: Proceedings of the IEEE international conference on computer vision, pp. 2794–2802.
- Mateo-García, G., Laparra, V., Gómez-Chova, L., 2019. Domain adaptation of landsat-8 and proba-v data using generative adversarial networks for cloud detection. In: IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 712–715.
- Ming Harry Hsu, T., Yu Chen, W., Hou, C.A., Hubert Tsai, Y.H., Yeh, Y.R., Frank Wang, Y.C., 2015. Unsupervised domain adaptation with imbalanced cross-domain data. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4121–4129.
- Murez, Z., Kolouri, S., Kriegman, D., Ramamoorthi, R., Kim, K., 2018. Image to image translation for domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4500–4509.
- Nogueron, R., Barreto, P., Souza Jr, C., Anderson, A., Salomão, R., 2006. Human pressure on the Brazilian Amazon forests. World Resources Institute, Washington, DC.
- Ortega Adarme, M., Queiroz Feitosa, R., Nigri Happ, P., Aparecido De Almeida, C., Rodrigues Gomes, A., 2020. Evaluation of deep learning techniques for deforestation detection in the brazilian amazon and cerrado biomes from remote sensing imagery. *Remote Sens.* 12, 910.
- Pan, S.J., Yang, Q., 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* 22, 1345–1359.
- Parente, L., Nogueira, S., Baumann, L., Almeida, C., Maurano, L., Affonso, A.G., Ferreira, L., 2021. Quality assessment of the prodes cerrado deforestation data. *Remote Sens. Appl. Soc. Environ.* 21, 100444.
- Pearson, K., 1901. Lili. on lines and planes of closest fit to systems of points in space. *Lond. Edinburgh Dublin Philos. Mag. J. Sci.* 2, 559–572.
- Pinheiro Maurano, L.E., Sobral Escada, M.I., Daleles Renno, C., 2019. Padrões espaciais de desmatamento e a estimativa da exatidão dos mapas do prodes para amazônia legal brasileira. *Ciência Florestal* 29, 01039954.
- Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. Springer, pp. 234–241.
- Sadeghi, V., Ahmadi, F.F., Ebadi, H., 2016. Design and implementation of an expert system for updating thematic maps using satellite imagery (case study: changes of lake urmia). *Arabian J. Geosci.* 9, 257.
- Saha, S., Bovolo, F., Bruzzone, L., 2019. Unsupervised multiple-change detection in vhr multisensor images via deep-learning based adaptation. In: IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 5033–5036.
- Saha, S., Solano-Correa, Y.T., Bovolo, F., Bruzzone, L., 2020. Unsupervised deep transfer learning-based change detection for hr multispectral images. *IEEE Geosci. Remote Sens. Lett.*
- Schenkel, F., Middelmann, W., 2019. Domain adaptation for semantic segmentation using convolutional neural networks. In: IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 728–731.
- Song, S., Yu, H., Miao, Z., Zhang, Q., Lin, Y., Wang, S., 2019. Domain adaptation for convolutional neural networks-based remote sensing scene classification. *IEEE Geosci. Remote Sens. Lett.* 16, 1324–1328.
- Soto, P., Costa, G., Feitosa, R., Happ, P., Ortega, M., Noa, J., Almeida, C., Heipke, C., 2020. Domain adaptation with cyclegan for change detection in the amazon forest. *Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci.* 43, 1635–1643.
- Sun, B., Saenko, K., 2016. Deep coral: Correlation alignment for deep domain adaptation. In: European conference on computer vision. Springer, pp. 443–450.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9.
- Tasar, O., Happy, S., Tarabalka, Y., Alliez, P., 2020. Colormapgan: Unsupervised domain adaptation for semantic segmentation using color mapping generative adversarial networks. In: *IEEE Trans. Geosci. Remote Sens.*
- Thonfeld, F., Feilhauer, H., Braun, M., Menz, G., 2016. Robust change vector analysis (rcva) for multi-sensor very high resolution optical satellite data. *Int. J. Appl. Earth Obs. Geoinf.* 50, 131–140.
- Tuia, D., Persello, C., Bruzzone, L., 2016. Domain adaptation for the classification of remote sensing data: An overview of recent advances. *IEEE Geosci. Remote Sens. Mag.* 4, 41–57.
- Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., Darrell, T., 2014. Deep domain confusion: Maximizing for domain invariance. arXiv preprint arXiv:1412.3474.
- Ulyanov, D., Vedaldi, A., Lempitsky, V., 2017. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6924–6932.
- Wang, M., Deng, W., 2018. Deep visual domain adaptation: A survey. *Neurocomputing* 312, 135–153.
- Weiss, K., Khoshgoftaar, T.M., Wang, D., 2016. A survey of transfer learning. *J. Big Data* 3, 9.
- Wiskott, L., 1999. Learning invariance manifolds. *Neurocomputing* 26, 925–932.
- Wiskott, L., Sejnowski, T.J., 2002. Slow feature analysis: Unsupervised learning of invariances. *Neural Comput.* 14, 715–770.
- Wittich, D., Rottensteiner, F., 2019. Adversarial domain adaptation for the classification of aerial images and height data using convolutional neural networks. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inform. Sci.* 4.
- Wittich, D., Rottensteiner, F., 2021. Appearance based deep domain adaptation for the classification of aerial images. *ISPRS J. Photogramm. Remote Sens.* 180, 82–102.
- Wu, C., Du, B., Zhang, L., 2013. Slow feature analysis for change detection in multispectral imagery. *IEEE Trans. Geosci. Remote Sens.* 52, 2858–2874.
- Yosinski, J., Clune, J., Bengio, Y., Lipson, H., 2014. How transferable are features in deep neural networks?. In: Advances in neural information processing systems, vol. 27, pp. 332–3328.
- Zagoruyko, S., Komodakis, N., 2015. Learning to compare image patches via convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4353–4361.
- Zhang, X., Chen, J., Meng, H., 2012. A novel sar image change detection based on graph-cut and generalized gaussian model. *IEEE Geosci. Remote Sens. Lett.* 10, 14–18.
- Zhou, L., Cao, G., Li, Y., Shang, Y., 2016. Change detection based on conditional random field with region connection constraints in high-resolution remote sensing images. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* 9, 3478–3488.
- Zhu, J.Y., Park, T., Isola, P., Efros, A.A., 2017a. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision, pp. 2223–2232.
- Zhu, X.X., Tuia, D., Mou, L., Xia, G.S., Zhang, L., Xu, F., Fraundorfer, F., 2017b. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geosci. Remote Sens. Mag.* 5, 8–36.