Multitask learning for large-scale semantic change detection

Rodrigo Caye Daudt, Bertrand Le Saux, Alexandre Boulch, Yann Gousseau

Multitask Learning for Large-scale Semantic Change Detection

Rodrigo Caye Daudt[a,b], Bertrand Le Saux[a], Alexandre Boulch[a], Yann Gousseau[b]

[a]*DTIS, ONERA, Université Paris-Saclay, FR-91123 Palaiseau, France*
[b]*LTCI, Télécom ParisTech, FR-75013 Paris, France*

Abstract

Change detection is one of the main problems in remote sensing, and is essential to the accurate processing and understanding of the large scale Earth observation data available. Most of the recently proposed change detection methods bring deep learning to this context, but change detection labelled datasets which are openly available are still very scarce, which limits the methods that can be proposed and tested. In this paper we present the first large scale very high resolution semantic change detection dataset, which enables the usage of deep supervised learning methods for semantic change detection with very high resolution images. The dataset contains coregistered RGB image pairs, pixel-wise change information and land cover information. We then propose several supervised learning methods using fully convolutional neural networks to perform semantic change detection. Most notably, we present a network architecture that performs change detection and land cover mapping simultaneously, while using the predicted land cover information to help to predict changes. We also describe a sequential training scheme that allows this network to be trained without setting a hyperparameter that balances different loss functions and achieves the best overall results.

1. Introduction

One of the main purposes of remote sensing is the observation of the evolution of the land. Satellite and aerial imaging enables us to keep track of the changes that occur around the globe, both in densely populated areas as well as in remote areas that are hard to reach. That is why change detection is a problem so closely studied in the context of remote sensing (Coppin et al., 2004). Change detection is the name given to the task of identifying areas of the Earth's surface that have experienced changes by jointly analysing two or more coregistered images (Bruzzone and Bovolo, 2013). Changes can be of several different types depending on the desired application, e.g. those caused by natural disasters, urban expansion, and deforestation. In this paper we treat change detection as a dense classification problem, aiming to predict a label for each pixel in an input image pair, i.e. achieving semantic segmentation.

The search for ever more accurate change detection comes from the value of surveying large amounts of land and analysing its evolution over a period of time. Detecting changes manually is a slow and laborious process (Singh, 1989) and the problem of automatic change detection using image pairs or sequences has been studied for many decades. The history of change detection algorithms and overviews of the most important methods are described in the reviews Singh (1989) and Hussain et al. (2013). Throughout the years, change detection benefited a lot from computer vision and image processing advances. In recent years, computer vision made tremendous progress thanks to machine learning techniques, and these were used for solving a wide range of problems related to image understanding (LeCun et al., 2015).

The rise of these techniques is explained by three main factors. First, the hardware required for the large amounts of calculations that are often required for machine learning techniques is becoming cheaper and more powerful.

This work was originally submitted under the title "High Resolution Semantic Change Detection", and had its title changed during the review process.

Second, new methods are being proposed to exploit the data in innovative ways. Finally, the amount of available data is increasing, which is essential for many machine learning techniques.

In this paper we propose a versatile supervised learning method to perform pixel-level change detection from image pairs based on state-of-the-art computer vision ideas. The proposed method is able to perform both binary and semantic change detection using very high resolution (VHR) images. Binary change detection attempts to identify which pixels correspond to areas where changes have occurred, whereas semantic change detection attempts to further identify the type of change that has occurred at each location. The proposed method is able to perform change detection using VHR images from sources such as WorldView-3, Plèiades and IGN's BD ORTHO. As was described by Hussain et al. (2013) and Bruzzone and Bovolo (2013), VHR change detection involves several extra challenges.

A new VHR semantic change detection dataset of unprecedented size is also presented in this paper. This dataset will be released publicly to serve as a benchmark and as a research tool for researchers working on change detection. The methods used to create this dataset, as well as the limitations of the available data, will be described later on. Until now, the most advanced ideas brought to computer vision by deep learning techniques could not be applied to change detection due to the lack of large annotated datasets. This dataset will enable the application of more sophisticated machine learning techniques that were heretofore too complex for the amount of change detection data available.

2. Related work

The work presented in this paper is based on several different ideas coming from two main research areas: change detection and machine learning. This section contains a discussion about the works hat have more heavily influenced this work, providing details about unsupervised methods, supervised learning, and fully convolutional networks for semantic segmentation.

Change detection algorithms usually comprise two main steps (Singh, 1989; Hussain et al., 2013). First, a difference metric is proposed so that a quantitative measurement of the difference between corresponding pixels can be calculated. The image generated from this step is usually called a difference image. Second, a thresholding method or decision function is proposed to separate the pixels into "change" and "no change" based on the difference image. These two steps are usually independent. Post-processing and pre-processing methods are sometimes used to improve results. Many algorithms use out-of-the-box registration algorithms and focus on the other main steps for change detection (Hussain et al., 2013). Most papers on change detection propose either a novel image differencing method (Bovolo and Bruzzone, 2005; El Amin et al., 2016, 2017; Zhan et al., 2017) or a novel decision function (Bruzzone and Prieto, 2000; Celik, 2009). A well established family of change detection methods is change vector analysis (CVA), considering the multispectral difference vector in polar or hyperspherical coordinates and attempting to characterise the changes based on the associated vectors at each pixel (Lambin and Strahlers, 1994; Bovolo and Bruzzone, 2007; Hussain et al., 2013). Most methods that propose image differencing techniques followed by thresholding assume that a threshold is chosen based on the difference image. The authors of Hussain et al. (2013) and Rosin and Ioannidis (2003) noted that the performance of such algorithms is scene dependent.

Hussain et al. (2013) categorise change detection algorithms into two main groups: pixel based and object based change detection. The former are attempts to identify whether or not a change has occurred at each pixel in the image pair, while the latter methods attempt to first group pixels that belong to the same object and use information such as the object's colour, shape and neighbourhood to help determine if that object has been changed between the acquisitions. Change detection algorithms can also be split into supervised and unsupervised groups.

As noted by Hussain et al. (2013) and Bruzzone and Bovolo (2013), change detection on low resolution images and on VHR images face different challenges. In low resolution images, pixels frequently contain information about several objects contained within its area. In such cases, a pixel in an image pair may contain both changed and unchanged surfaces simultaneously. VHR images are more susceptible to problems such as parallax, high reflectance variability for objects of the same class, and co-registration problems (Bruzzone and Bovolo, 2013). It follows that algorithms that perform change detection on

(a) Image 1. (b) LCM 1. (c) Image 2. (d) LCM 2. (e) Change map.

(f) Image 1. (g) LCM 1. (h) Image 2. (i) LCM 2. (j) Change map.

Figure 1: Examples of image pairs, land cover maps (LCM) and associated pixel-wise change maps from the HRSCD dataset. In the depicted LCMs, blue represents the "artificial surfaces" class, and orange represents the "agricultural areas" class.

very high resolution images must be aware of not only a given pixel's values, but also of information about its neighbourhood.

Machine learning algorithms, and notably convolutional neural networks (CNNs) in recent years, also have had great impact. For examples, in remote sensing, CNNs were used for road detection (Mnih and Hinton, 2010), and in computer vision, CNNs were used on the related task of comparing image pairs (Chopra et al., 2005; Zagoruyko and Komodakis, 2015). We now examine in details unsupervised and supervised machine learning approaches, the latter category being then subdivided in standard techniques, CNNs and Fully-Convolutional Neural Networks.

Unsupervised methods have been used for change detection in many different ways (Hussain et al., 2013; Vakalopoulou et al., 2015; Liu et al., 2019). In the context of change detection, annotated datasets are extremely scarce and often kept private. Thus, unsupervised methods are extremely useful, since, unlike supervised methods, they do not need labelled data for training. Many of these methods automatically analyse the data in difference images and detect patterns that correspond to changes (Bazi et al., 2005; Bruzzone and Prieto, 2000). Other methods use unsupervised learning approaches such as iterative training (Liu et al., 2016), autoencoders (Zhao et al., 2014), and principal component analysis with k-means clustering (Celik, 2009) to separate changed pixels from unchanged ones.

Supervised change detection algorithms require labelled training data from which the task of change detection can be learned. Several methods have been proposed for performing change detection using supervised learning algorithms such as support vector machines (Huang et al., 2008; Volpi et al., 2009, 2013; Le Saux and Randrianarivo, 2013), random forests (Sesnie et al., 2008), and neural networks (Gopal and Woodcock, 1996; Dai and Khorram, 1999; Zhao et al., 2014). CNN architectures have also been proposed to perform supervised change detection (Zhan et al., 2017; Chen et al., 2018b).

Convolutional neural networks (CNNs) for change detection have been proposed by different authors in the recent years. The majority of these methods avoid the problem of the lack of data by using transfer learning techniques, i.e. using networks that have been pre-trained for a different purpose on a large dataset (El Amin et al., 2016, 2017). While transfer learning is a valid solution, it is also limiting. Firstly, end-to-end training tends to achieve the best results for a given problem when possible. Transfer learning also assumes that all images are of the same type. As most large scale datasets contain RGB images, this means that extra bands contained in multispectral images must be ignored. It has however been shown that using all available multispectral bands for change detection leads to better results (Daudt et al., 2018b).

Several works have used CNNs to generate the difference image that was described earlier, followed by traditional thresholding methods on those images. El Amin et al. (2016, 2017) proposed using the activation of pretrained CNNs to generate descriptors for each pixel, and using the Euclidean distance between these descriptors to build the difference image. Zhan et al. (2017) trained a network to produce a 16-dimensional descriptor for each pixel. Descriptors were similar for pixels with no change and dissimilar for pixels that experienced change. Liu et al. (2016) used deep belief networks to generate pixel descriptors from heterogeneous image pairs, then the Euclidean distance is used to build a difference image. Zhao et al. (2014) proposed a deep belief network that takes into account the context of a pixel to build its descriptor. Mou et al. (2019) proposed using patch based recurrent CNNs to detect changes in image pairs. CNNs for change detection have also been studied outside the context of remote sensing, such as surface inspection (Stent et al., 2015).

Fully convolutional neural networks (FCNNs) are a type of CNNs that are especially suited for dense prediction of labels and semantic segmentation (Long et al., 2015). Unlike traditional CNNs, which output a single prediction for each input image, FCNNs are able to predict labels for each pixel independently and efficiently. Ronneberger et al. (2015) proposed a simple and elegant addition to FCNNs that aims to improve the accuracy of the final prediction results. The proposed idea is to connect directly layers in earlier stages of the network to layers at later stages to recover accurate spatial information of region boundaries. FCNNs currently achieve state-of-the-art results in semantic segmentation problems, including those in remote sensing (Volpi and Tuia, 2017; Maggiori et al., 2017; Chen et al., 2018a).

Fully convolutional networks trained from scratch to perform change detection were proposed for the first time by Daudt et al. (2018a). Both Siamese and early fusion architectures were compared, expanding on the ideas proposed earlier by Chopra et al. (2005) and Zagoruyko and Komodakis (2015). A similar approach was simultaneously proposed by Chen et al. (2018b) outside the context of remote sensing. To the best of our knowledge, the only other time a fully convolutional Siamese network has been proposed was by Bertinetto et al. (2016) with the purpose of tracking objects in image sequences.

3. Dataset

Research on the problem of change detection is hindered by a lack of open datasets. Such datasets are essential for a methodical evaluation of different algorithms. Benedek and Szirányi (2009) created a binary change dataset with 13 aerial image pairs split into three regions called the Air Change dataset. A dataset, called ONERA Satellite Change Detection (OSCD) dataset, composed of 24 multispectral image pairs taken by the Sentinel-2 satellites is presented in (Daudt et al., 2018b). Both of these datasets allow for simple machine learning techniques to be applied to the problem of change detection, but with these small amounts of images overfitting becomes one of the main concerns even with relatively simple models. The Aerial Imagery Change Detection (AICD) dataset contains synthetic aerial images with artificial changes generated with a rendering engine (Bourdis et al., 2011). These datasets do not contain semantic information about the land cover of the images, and contain either low resolution (OSCD, Air Change) or simulated (AICD) images.

For this reason, we have created the first large scale dataset for semantic change detection, which we present in this section. The High Resolution Semantic Change Detection (HRSCD) dataset will be released to the scientific community to be used as a benchmark for semantic change detection algorithms and to open the doors to the usage of state-of-the-art deep learning algorithms in this context. The dataset contains not only information about where changes have taken place, but also semantic information about the imaged terrain in all images of the dataset. Examples of image pairs, land cover maps (LCM) and change maps taken from the dataset are depicted in Fig. 1.

3.1. Images

The dataset contains a total of 291 RGB image pairs of 10000x10000 pixels. These are mosaics of aerial images taken by the French National Institute of Geographical and Forest Information (IGN). The image pairs contain an earlier image acquired in 2005 or 2006, and a second image acquired in 2012. They come from a database named *BD ORTHO* which contains orthorectified aerial images of several regions of France from different years at a resolution of 50 cm per pixel. The 291 selected image pairs are all the images in this database that satisfy the

conditions for the labels, which will be described below. The images cover a range of urban and countryside areas around the French cities of Rennes and Caen.

The dataset contains more than 3000 times more annotated pixel pairs than either OSCD or Air Change datasets. Also, unlike these datasets, the labels contain information about the types of change that have occurred. Finally, labels about the land cover of the images in the dataset are also available. This is much more data than was previously available in the context of change detection and it opens the doors for many new ideas to be tested. The amount of labelled pixels and surface area for land cover classification is also about 8 times larger in the proposed HRSCD dataset than in the DeepGlobe Land Cover Classification dataset (Demir et al., 2018), both of the datasets containing images of the same spatial resolution (50 cm/px).

The *BD ORTHO* images provided by IGN are available for free for research purposes, but not all images can be redistributed by the users. That is the case for the images taken in 2005 and 2006. Nevertheless, we will make available all the data for which we have the rights of redistribution and the rasters that we have generated for semantic change detection and land cover mapping. The dataset will also contain instructions for downloading the remaining images that are necessary for using the dataset directly from IGN's website.

3.2. Labels

The labels in the dataset come from the European Environment Agency's (EEA) Copernicus Land Monitoring Service - Urban Atlas project. It provides "reliable, intercomparable, high-resolution land use maps" for functional urban areas in Europe with more than 50000 inhabitants. These maps were generated for the years of 2006 and 2012, and a third map is available containing the changes that took place in that period. Only the images in the regions mapped in the Urban Atlas project and with a maximum temporal distance of one year were kept in the dataset.

The available land cover maps are divided in several semantic classes, which are in turn organised in different hierarchical levels. By grouping the labels at different hierarchical levels it is possible to generate maps that are more coarsely or finely divided. For example, grouping the labels with the coarsest hierarchical level yields five classes (plus the "no information" class) shown in Table 1. This hierarchical level will henceforth be referred to as L1.

Table 1: Urban Atlas land cover mapping classes at hierarchical level L1

Code	**Class**
0	No information
1	Artificial surfaces
2	Agricultural areas
3	Forests
4	Wetlands
5	Water

These maps are openly available in vector form online. We have used these vector maps and the georeferenced *BD ORTHO* images to generate rasters of the vector maps that are aligned with the rasters of the images. These rasters allow us to have ground truth information about each pixel in the dataset.

It is important to note that there are slight differences in the semantic classes present in Urban Atlas 2006 and in Urban Atlas 2012. These differences do not affect the L1 hierarchical grouping and therefore had no consequence in the work presented later in this paper. It may nevertheless affect future works done with the data. We leave it up to the users how to best interpret and deal with these differences. More information will be provided in the dataset files.

3.3. Dataset analysis

Despite its unprecedented size and qualities, we acknowledge in this section the dataset's limitations and challenges. Nevertheless, we will show later in this paper that despite these limitations, the dataset allows for the boundaries of the state-of-the-art in semantic change detection through machine learning to be pushed.

One issue is the accuracy of the labels contained in the Urban Atlas vector maps with respect to the *BD ORTHO* images. We do not have access to the images used to build the Urban Atlas vector maps, nor to the exact dates of their acquisitions, nor to the dates of acquisition of the images in *BD ORTHO*. Hence, there are some discrepancies between the information in the vector maps and in the images. Furthermore, EEA only guarantees a minimum label accuracy of 80-85% depending on the consid-

(a) Image 1. (b) Image 2. (c) Inaccurate border.

(d) Image 1. (e) Image 2. (f) False negative.

(g) Image 1. (h) Image 2. (i) False positive.

Figure 2: Examples of: ((a)-(c)) overly large change markings, ((d)-(f)) failure to mark changes, ((g)-(i)) false positive.

ered class. Most of the available data is accurate, but it is important to consider that the labels in the dataset are not flawless. Examples of false negatives and false positives can be see in Fig. 2 (d)-(f) and Fig. 2 (g)-(i), respectively. It is also worth noting that the labels have been created using previously known vector maps, mostly by labelling correctly each of the known regions. This means a single label was given to each region, and this led to inaccurate borders in some cases. This can be clearly seen in Fig. 2 (a)-(c).

One of the main challenges involved in using this dataset for supervised learning is the extreme label imbalance. As can be seen in Table 2, 99.232% of all pixels are labelled as no change, and the largest class is from agricultural areas to artificial surfaces (i.e. class 2 to class 1), which accounts for 0.653% of all pixels. These two classes together account for 99.885% of all pixels, which means all other change types combined account for only 0.115% of all pixels. Furthermore, many of the possible types of change have no examples at all in any of the images of the dataset. It is of paramount importance when

Table 2: Change class imbalance at hierarchical level L1. Row number represents class in 2006, column number represents class in 2012. Classes were defined in Table 1.

	1	2	3	4	5
1	0%	0.011%	0%	0.001%	0.001%
2	0.653%	0%	0.001%	0%	0.077%
3	0.014%	0.002%	0%	0%	0%
4	0%	0%	0%	0%	0%
5	0.001%	0.004%	0%	0.004%	0%

No change	99.232%

using this dataset to take into account this imbalance. This also means that using the overall accuracy as a performance metric with this dataset is not a good choice, as it virtually only reflects how many pixels of the no change class have been classified correctly. Other metrics, such as Cohen's kappa coefficient or the Sørensen-Dice coefficient, must be used instead. This class imbalance is characteristic of real world large scale data, where changes are much less frequent than unchanged surfaces. Therefore, this dataset provides a realistic evaluation tool for change detection methods, unlike carefully selected image pairs with large changed regions.

The problem of supervised learning using noisy labels has already been studied and evidence suggests that supervised learning with noisy labels is possible as long as a dataset of a large enough size is used (Rolnick et al., 2017). Other works attempt to explicitly deal with the noisy labels present in the dataset and prioritise the correct labels during training (Maggiolo et al., 2018).

Finally, we acknowledge how challenging it is to use hierarchical levels finer than L1 due to: 1) a massive increase in the number of possible changes, and 2) the difference between similar classes becomes more abstract and context based. For example, the difference between the "Discontinuous Medium Density Urban Fabric" and the "Discontinuous Low Density Urban Fabric" classes defined in Urban Atlas depends not only in correctly identifying the surface at a given pixel (e.g. building or grass), but also by understanding the surroundings of the pixel and calculating the ratio between these two classes at a given neighbourhood that is not clearly defined.

4. Methodology

4.1. Binary change detection

We have already showed in a previous work the efficacy of using three different architectures of fully convolutional neural networks for change detection (Daudt et al., 2018a). Chen et al. (2018b) simultaneously proposed a fully convolutional architecture for change detection that is very similar to one of the three initially proposed architectures. In both of these works, FCNN architectures performed better than previous methods for change detection.

Building on this previous work, we have modified the FC-EF architecture proposed in Daudt et al. (2018a) to use residual blocks, as proposed by He et al. (2016). The resulting network is later referred to as FC-EF-Res, and is depicted in Fig. 3. These residual blocks were used in an encoder-decoder architecture with skip connections to improve the spatial accuracy of the results (Ronneberger et al., 2015). These residual blocks were chosen to facilitate the training of the network, which is especially important for its deeper variations that will be discussed later.

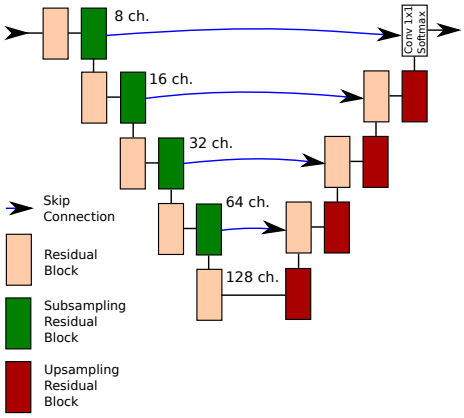


Figure 3: FC-EF-Res architecture, used for tests with smaller datasets to avoid overfitting. Using residual blocks improves network performance and facilitates training.

When testing on the OSCD dataset (Section 5.1), the size of the network has been kept approximately the same as in Daudt et al. (2018a) to avoid overfitting. When using the proposed HRSCD dataset (Section 5.2), the larger amount of annotated pixels allows us to use deeper and more complex models. In that case, the number of encoding levels and residual blocks per level has been increased, but the idea behind the network is the same as of FC-EF-Res.

4.2. Semantic change detection

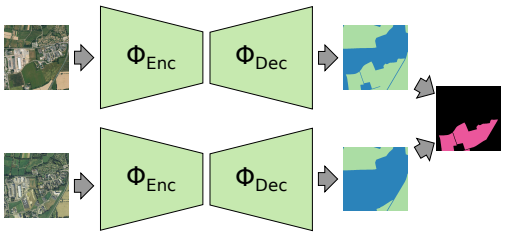
As was mentioned earlier, the efficiency of the proposed architecture for binary change detection and the availability of the HRSCD dataset enable us to tackle the problem of semantic change detection. This problem consists of two separate but not independent parts. The first task is analogue to binary change detection, i.e. we attempt to determine whether a change has occurred at each pixel in a co-registered multi-temporal image pair. The second task is to differentiate between types of changes. In our case, this consists of predicting the class of the pixel in each of the two given images. The problem of semantic change detection lies in the intersection between change detection and land cover mapping.

Below we will describe four different intuitive strategies to perform semantic change detection using deep neural networks. Starting from the plain comparison of land cover maps, we then develop more involved strategies. These strategies vary in complexity and performance, as will be discussed in Section 5.

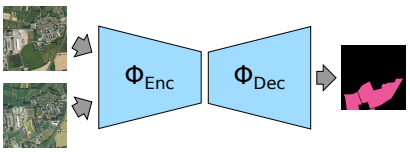
4.2.1. Strategy 1: Direct comparison of LCMs

The problem of automatic land cover mapping is a well studied problem. In particular, methods involving CNNs have recently been proposed, yielding good performances (Audebert et al., 2016). When the land cover information is available, as it is the case in the HRSCD dataset, the most intuitive method that can be proposed for semantic change detection would be to train a land cover mapping network and to compare the results for pixels in the image pair (see Fig. 4(a)).

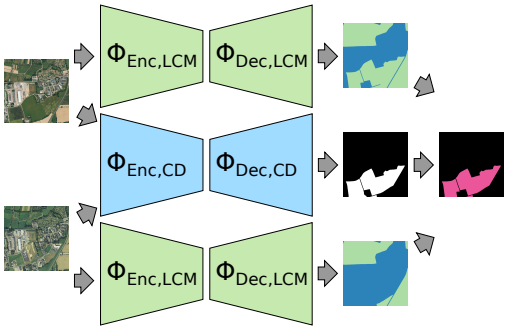
The advantage of this method is its simplicity. In many cases we could assume changes occurred where the predicted class label differs between the two images, and the type of change is given by the predicted labels at each of the two acquisition moments. The weakness of this method is that it heavily depends on the accuracy of the predicted land cover maps. While modern FCNNs are able to map areas to a good degree of accuracy, there are still many wrongly predicted labels, especially around the



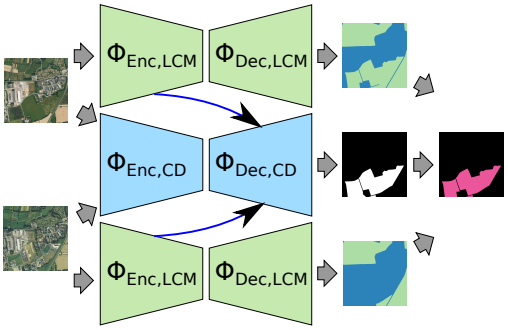
(a) Strategy 1: semantic CD from land cover maps.



(b) Strategy 2: direct semantic CD.



(c) Strategy 3: separate CD and LCM.



(d) Strategy 4: integrated CD and LCM.

Figure 4: Schematics for all four proposed strategies for semantic change detection. Φ represents the network branch's learnable parameters, "Enc" means encoder, "Dec" means decoder, "LCM" means land cover mapping, and "CD" means change detection.

boundaries between regions of different classes. Furthermore, when comparing the results for two acquisitions the prediction errors would accumulate. This means the accuracy of this change detection algorithm would be lower than the land cover mapping network, and would likely predict changes in the borders between classes simply due to the inaccuracy of the network.

4.2.2. Strategy 2: Direct semantic CD

A second intuitive approach is to treat each possible type of change as a different and independent label, considering semantic change detection as a simple semantic segmentation along the lines of what has been done to binary change detection in the past (Daudt et al., 2018a).

The weakness of this method is that the number of change classes grows proportionately to the square of the number of land cover classes that is considered. This, combined with the class imbalance problem that was discussed earlier, proves to be a major challenge when training the network.

4.2.3. Strategy 3: Separate LCM and CD

Since it has been proven before that FCNNs are able to perform both binary change detection and land cover mapping, a third possible approach is to train two separate networks that together perform semantic change detection (see Fig. 4(c)). The first network performs binary change detection on the image pair, while the second network performs land cover mapping of each of the input images. The two networks are trained separately since they are independent.

In this strategy, the two input images produce three outputs: two land cover maps and a change map. At each pixel, the presence of change is predicted by the change map, and the type of change is defined by the classes predicted by the land cover maps at that location. This way the number of predicted classes is reduced relative to the previous strategy (i.e. the number of classes is no longer proportional to the square of land cover classes) without loss of flexibility. This helps with the class imbalance problem. It also avoids the problem of predicting changes at every pixel where the land cover maps differ, since the

change detection problem is treated separately from land cover mapping.

We argue that such network may be able to identify changes of types it has not seen during training, as long as it has seen the land cover classes during training. For example, the network could in theory correctly classify a change from agricultural area to wetland even if such changes are not in the training set, as long as it has enough examples of those classes to correctly classify them in the land cover mapping branches. The combination of two separate networks allows us to split the problem into two, and optimise each part to maximise performance.

4.2.4. Strategy 4: Integrated LCM and CD

The last of the proposed approaches is an evolution of the previous strategy of using two FCNNs for the tasks of binary change detection and land cover mapping. We propose to integrate the two FCNNs into a single multi-task network (see Fig. 4(d) and Fig. 5) so that land cover information can be used for change detection. The combined network takes as input the two co-registered images and outputs three maps: the binary change map and the two land cover maps.

In the proposed architecture, information from the land cover mapping branches of the network is passed to the change detection branch of the network in the form of difference skip connections, which was shown to be the most effective form of skip connections for Siamese FC-NNs (Daudt et al., 2018a). The weights of the two land cover mapping branches are shared since they perform an identical task, allowing us to significantly reduce the number of learned parameters.

This multipurpose network gives rise to a new issue during the training phase. Given that the network outputs three different image predictions, it is necessary to balance the loss functions from these results. Since two of the outputs have exactly the same nature (the land cover maps), it follows from the symmetry of these branches that they can be combined into a single loss function by simple addition. The question remains on how to balance the binary change detection loss function and the land cover mapping loss function to maximise performance.

We have proposed and tested two different strategies for training the network. The first and more naive approach to this problem is to minimise a loss function that is a weighted combination of the two loss functions. This loss function would have the form

$$\begin{aligned}\mathcal{L}_\lambda(&\Phi_{\text{Enc,CD}},\Phi_{\text{Dec,CD}},\Phi_{\text{Enc,LCM}},\Phi_{\text{Dec,LCM}})\\ &=\mathcal{L}(\Phi_{\text{Enc,CD}},\Phi_{\text{Dec,CD}})+\lambda\mathcal{L}(\Phi_{\text{Enc,LCM}},\Phi_{\text{Dec,LCM}})\end{aligned} \tag{1}$$

where Φ represents the various network branch parameters, and $\mathcal{L}$ is a pixel-wise loss function. In this work, the pixel-wise cross entropy function was used as loss function as is traditional in semantic segmentation problems. The problem then becomes the search for the value of λ that leads to the best balance between the two loss terms. This can be found through a grid search, but the test of each value of λ is done by training the whole network until convergence, which is a slow and costly procedure. This will later be referred to as Strategy 4.1.

To reduce the aforementioned training burden, we propose a second approach to train the network that avoids the need of setting the hyperparameter λ. We train the network in two stages. First, we consider only the land cover mapping loss

$$\begin{aligned}\mathcal{L}_1(&\Phi_{\text{Enc,CD}},\Phi_{\text{Dec,CD}},\Phi_{\text{Enc,LCM}},\Phi_{\text{Dec,LCM}})\\ &=\mathcal{L}(\Phi_{\text{Enc,LCM}},\Phi_{\text{Dec,LCM}})\end{aligned} \tag{2}$$

and train only the land cover mapping branches of the network, i.e. we do not train $\Phi_{\text{Enc,CD}}$ or $\Phi_{\text{Dec,CD}}$ at this stage. Since the change detection branch has no influence on the land cover mapping branches, we can train these branches to achieve the maximum possible land cover mapping performance with the given architecture and data. Next, we use a second loss function based only on the change detection branch:

$$\begin{aligned}\mathcal{L}_2(&\Phi_{\text{Enc,CD}},\Phi_{\text{Dec,CD}},\Phi_{\text{Enc,LCM}},\Phi_{\text{Dec,LCM}})\\ &=\mathcal{L}(\Phi_{\text{Enc,CD}},\Phi_{\text{Dec,CD}})\end{aligned} \tag{3}$$

while keeping the weights for the land cover mapping $\Phi_{\text{Enc,LCM}}$ and $\Phi_{\text{Enc,LCM}}$ fixed. This way, the change detection branch learns to use the predicted land cover information to help to detect changes without affecting land cover mapping performance. This will later be referred to as Strategy 4.2.

5. Results

5.1. Multispectral change detection

We first evaluate the performance of the proposed FC-EF-Res network. As explained in Section 4.1, this net-

Table 3: Summary of proposed change detection strategies.

Str.	Description	Training
1	Diff. of LCMs	LCM supervision
2	Direct semantic CD	Multiclass CD supervision
3	Separate CD and LCM	Separate LCM and CD
4.1	Integrated CD and LCM	Triple loss function
4.2	Integrated CD and LCM	Sequential training

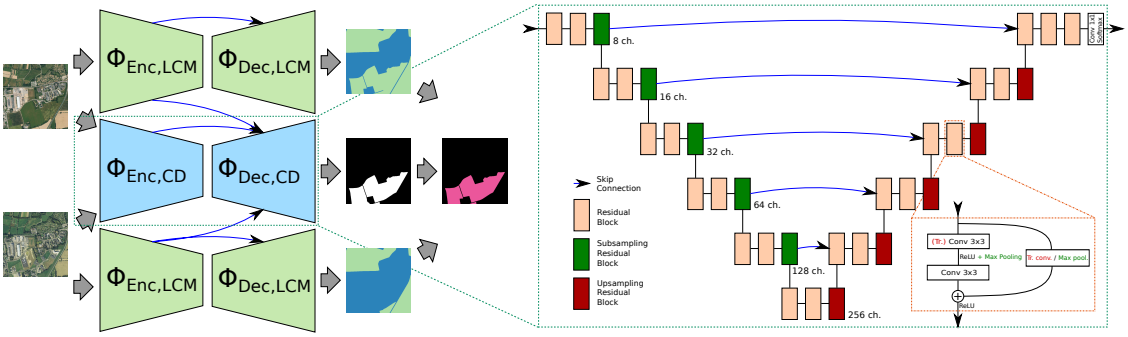


Figure 5: Detailed schematics for the integrated change detection and land cover mapping network (Strategy 4). The encoder-decoder architecture is the same that was used for all 4 strategies.

Table 4: Definitions of metrics used for evaluating results quantitatively. Legend: TP - true positive, TN - true negative, FP - false positive, FN - false negative, p_o - observed agreement between ground truth and predictions, p_e - expected agreement between ground truth and predictions given class distributions.

Tot. acc.	$(TP+TN)/(TP+TN+FP+FN)$
Precision	$TP/(TP+FP)$
Recall	$TP/(TP+FN)$
Dice	$2 \cdot TP/(2 \cdot TP+FP+FN)$
Kappa	$(p_o - p_e)/(1-p_e)$

work is an evolution of the convolutional architecture FC-EF proposed in Daudt et al. (2018a), to which residual blocks have been added in place of traditional convolutional layers.

The FC-EF-Res architecture was compared to the previously proposed FCNN architectures on the OSCD dataset for binary change detection, which contains lower-resolution Sentinel-2 image pairs with 13 multispectral bands. As expected, the residual extension of the FC-EF architecture outperformed all previously proposed architectures. The difference was noted on both the RGB and the multispectral cases. On the RGB case, the improvement was of such magnitude that the change detection performance on RGB images almost matched the performance on multispectral images. The results can be seen in Table 5. This corroborates the claims made by He et al. (2016) that using residual blocks improves the training performance of CNNs. For this reason, all networks that are tested with the HRSCD dataset use residual modules.

5.2. Very high resolution semantic change detection

To test the methods proposed in Section 4.2 we split the HRSCD images into two groups: 146 image pairs for training and 145 image pairs for testing. By splitting the train and test sets this way we can ensure that no pixel in the test set has been seen during training. Class weights were set inversely proportional to the number of training examples to counterbalance the dataset's class imbalance. The results for each of the proposed strategies can be seen in Table 6, and illustrative image results can be seen in Fig. 6.

Table 5: Change detection results of several methods on the OSCD dataset, for the RGB and multispectral (MS) cases. Results are in percent.

Data	Network	Prec.	Recall	Tot. acc.	Dice
RGB	FC-EF	44.72	53.92	94.23	48.89
	FC-Siam-conc	42.89	47.77	94.07	45.20
	FC-Siam-diff	49.81	47.94	94.86	48.86
	FC-EF-Res	**52.27**	**68.24**	**95.34**	**59.20**
MS	FC-EF	**64.42**	50.97	**96.05**	56.91
	FC-Siam-conc	42.39	65.15	93.68	51.36
	FC-Siam-diff	57.84	57.99	95.68	57.92
	FC-EF-Res	54.93	**66.48**	95.64	**60.15**

As is the case for most deep neural networks, the training times for the proposed methods are significantly larger than the testing times. Once the network has been trained, its fast inference speed allows it to process large amounts of data efficiently. The proposed methods took 3-5 hours of training time using a GeForce GTX 1080 Ti GPU with 11GB of memory. Inference times of the proposed methods were under 0.04 s for 512x512 image pairs using the same hardware.

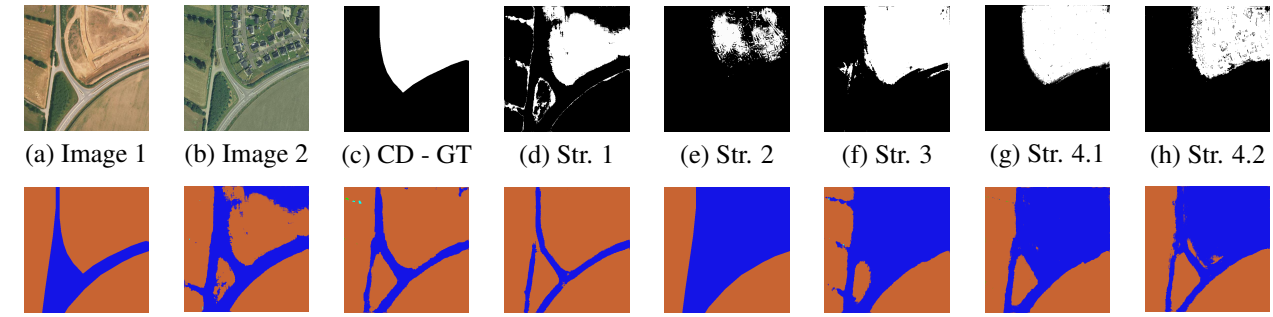
In Strategy 1, which naively attempts to predict change maps from land cover maps, we can see that the network succeeds in accurately classifying the imaged terrains, but this is not enough to predict accurate change maps. The change detection kappa coefficient for this strategy is very low, which means this method is marginally better than chance for change detection.

The results for Strategy 2 are a fair improvement over those of Strategy 1. The change detection Dice coefficient and the land cover mapping results for this method are not reported due to its nature, since Dice coefficients can only be calculated for binary classification problems, and this strategy bypasses the land cover mapping steps. Despite achieving a higher kappa coefficient, the network learned to always predict the same type of change where changes occurred. This means that despite using appropriately tuned class weights, the learning process did not succeed in overcoming the extreme class imbalance present in the dataset. In other words, the network learned to detect changes but no semantic information was present in the results.

For Strategy 3, the land cover mapping network that was used was the same as that of Strategy 1, which achieved good performance. A binary change detection network was trained to be used for masking the land cover maps. The performance of this network was better than that of Strategy 1 but worse than that of Strategy 2. The results show that this is due to an overestimation of the change class. This shows once again how challenging dealing with the extreme class imbalance is.

The results of Strategy 4 are the best ones overall. The simultaneous training strategy (Str. 4.1) achieves excellent performance in both land cover mapping and change detection, proving the viability of this strategy. The reported results were obtained with $\lambda = 0.05$, which is a value that prioritises the training of the change detection branch of the network. We then see that the same network trained with sequential training (Str. 4.2) obtained even better results in both change detection and land cover mapping without needing to search for an adequate parameter λ. This, according to our results, is the best semantic change detection method. By comparing the results for Strategies 3 and 4 we can see the improvements that result directly from integrating the change detection and land cover mapping branches of the networks. In other words, Strategy 4.2 allows us to maximise the change detection performance without reducing the land cover mapping accuracy.

The best performing land cover mapping method was the single purpose network that was trained and used for Strategies 1 and 3. The fact that it achieves a better kappa coefficient than Strategy 4.2 is merely due to the randomness of the initialisation and training of the network, as the land cover mapping branches of Strategy 4.2 are identical to those used in Strategies 1 and 3. This also explains why their results are so similar. By comparing these results to those of Strategy 4.1 it emphasises once again the fact that



(a) Image 1 (b) Image 2 (c) CD - GT (d) Str. 1 (e) Str. 2 (f) Str. 3 (g) Str. 4.1 (h) Str. 4.2

(i) LCM 1 - GT (j) Str. 1/3 (k) Str. 4.1 (l) Str. 4.2 (m) LCM 2 - GT (n) Str. 1/3 (o) Str. 4.1 (p) Str. 4.2

Figure 6: Illustrative images of the obtained results: (a)-(b) multitemporal image pair; (c) ground truth change detection map; (d)-(h) predicted change maps; (i)-(l) ground truth and predicted land cover maps for image 1; (m)-(p) ground truth and predicted land cover maps for image 2.

attempting to train the network shown in Fig. 5 all at once damages performance in both change detection and land cover mapping.

In Fig. 6 we can see the results of the proposed networks on a pair of images from the dataset. Note the amount of false detections by Strategy 1 due to the lack of accuracy of prediction of the land cover maps on region boundaries. The second row shows the predicted classes at each pixel for each image. The semantic information about the changes comes from comparing these two predictions. For example, comparing the images in Fig. 6 (k) and (o) we can say that the changes predicted in (g) were from the "Agricultural areas" class to the "Artificial surfaces" class.

In our tests we observed that the trained networks had the tendency to overestimate the size of the detected changes. It is likely that this happens simply due to the nature of the data that was used for training. The labels in the HRSCD dataset, which come from Urban Atlas, mark as a change the whole terrain where a change of class happened. This means that not only the pixels associated with a given change are marked as change, but the neighbouring pixels that are in the same parcel are also marked as change. This leads to the networks learning to overestimate the boundary of the detected changes in an attempt to also correctly classify the pixels surrounding the detected change. This once again reflects the challenges of the HRSCD dataset.

The performance of two state-of-the-art CD methods are also shown in Table 6. The first method, proposed by El Amin et al. (2016), is based on transfer learning and uses features from a pretrained VGG-19 model (Simonyan and Zisserman, 2015) to create pixel descriptors, whose Euclidean distance is used to build a difference image. The original method uses Otsu thresholding to perform CD, but we have found that such approach leads to overestimating changes. We therefore tuned a fixed threshold (T = 2300) using a few example images and used that value to test the algorithm on all test data, which significantly increased its performance by reducing false positives. Also included are the results by the method proposed by Celik (2009), which performs principal component analysis (PCA) and k-means clustering on the pixels to detect changes in an unsupervised manner. Both algorithms perform worse than the proposed method on the HRSCD dataset.

To evaluate the size of the dataset, we have also tested Strategy 4.2 using reduced amounts of data for training the network. The kappa coefficient, in percent, obtained by using the whole training dataset is 25.49. This value is reduced to 23.34 by using half the training data, and is further reduced to 22.18 by using a quarter of the data. This shows that, as expected, using more data for training the network leads to better results. Nonetheless, it also shows that the dataset is large enough to allow for even more complex and data hungry methods to be trained using the HRSCD dataset in the future.

Finally, it is important to note that the label imperfections in the HRSCD dataset occur not only in the training images, but also in the test images. This means that

Table 6: Change detection (CD) and land cover mapping (LCM) results of all four of the proposed strategies on the HRSCD dataset. Comparison with the methods proposed by El Amin et al. (2016) (Otsu [CNNF-O] and fixed [CNNF-F] thresholding) and by Celik (2009) ([PCA+KM]) are included. Results are in percent.

	CD			LCM	
	Kappa	Dice	Tot. acc.	Kappa	Tot. acc.
Str. 1	3.99	5.56	86.07	**71.92**	87.22
Str. 2	21.54	-	**98.30**	-	-
Str. 3	12.48	13.79	94.72	**71.92**	87.22
Str. 4.1	19.13	20.23	96.87	67.25	85.74
Str. 4.2	**25.49**	**26.33**	98.19	71.81	**89.01**
CNNF-O	0.74	2.43	64.54	-	-
CNNF-F	3.28	4.84	88.66	-	-
PCA+KM	0.67	2.31	83.95	-	-

Table 7: Change detection results on Eppalock lake test images. Results are in percent.

		ReCNN-LSTM	EF
Binary CD	Tot. acc.	98.67	**99.35**
	Kappa	97.28	**98.67**
	No change	98.83	**99.47**
	Change	98.46	**99.19**
Semantic CD	Tot. acc.	**98.70**	98.48
	Kappa	**97.52**	97.10
	No change	**98.49**	97.73
	City exp.	84.72	**100**
	Soil change	**100**	86.07
	Water change	99.25	**99.93**

the performance of the proposed methods may be even higher than the numbers suggest, since some of the disagreements between prediction and ground truth data are actually due to errors in the ground truth data.

5.3. Eppalock lake images

We compare our method in this section to the one proposed by Mou et al. (2019), which used recurrent convolutional neural networks for change detection. In that work, pixels were randomly split into train and test sets. We believe that this split leads to overfitting since neighbouring pixels contain redundant information. This is especially true when using CNNs, which take as inputs patches centred on the considered pixels, meaning the network sees the same information for training and testing. It is likely that overfitting takes place, since an accuracy of over 98% is achieved by using only 1000 labelled pixels to train a network with 67500 parameters (for their long short-term memory (LSTM) architecture, which performed the best). The data consists of a single image pair of 631x602 pixels only partially annotated, with a total of 8895 annotated pixels which is much less data than what is required for deep learning methods. The HRSCD dataset presented in Section 3 contains over 3 million times more labelled pixels than the Eppalock lake image pair. Despite the flaws of this testing scheme, we have followed it to achieve a fair comparison between the methods.

Using the CNN architecture labelled EF by Daudt et al. (2018b), we have achieved excellent numeric results which discouraged the usage of more complex methods which would lead to even more extreme overfitting. The results achieved by the EF network were better for binary change detection and equivalent for semantic change detection compared to ReCNN-LSTM. The results can be seen in Table 7.

6. Conclusion

The first major contribution presented in this paper is the first large scale very high resolution semantic change detection dataset that will be released to the scientific community. This dataset contains 291 pairs of aerial images, together with aligned rasters for change maps and land cover maps. This dataset allows for the first time for deep learning methods to be used in this context in a fully

supervised manner with minimal concern for overfitting. We have then proposed different methods for using deep FCNNs for semantic change detection. The best among the proposed methods is an integrated network that performs land cover mapping and change detection simultaneously, using information from the land cover mapping branches to help with change detection. We also proposed a sequential training scheme for this network that avoids the need of tuning a hyperparameter, which circumvents a costly grid search.

The automatic methods used to generate the HRSCD dataset resulted in noisy labels for both training and testing, and how to deal with this problem is still an open question. It would also be interesting to explore ways to explicitly deal with parallax problems which are present in VHR images which sometimes lead to false positives due to the different points of view and the geometry of the scene.

Acknowledgments

This work is part of ONERA's project DELTA. We thank X. Zhu and L. Mou (DLR) for the Eppalock Lake images.

References

Audebert, N., Le Saux, B., Lefèvre, S., 2016. Semantic segmentation of earth observation data using multimodal and multi-scale deep networks, in: Asian Conference on Computer Vision, pp. 180–196.

Bazi, Y., Bruzzone, L., Melgani, F., 2005. An unsupervised approach based on the generalized gaussian model to automatic change detection in multitemporal sar images. IEEE Transactions on Geoscience and Remote Sensing 43, 874–887.

Benedek, C., Szirányi, T., 2009. Change detection in optical aerial images by a multilayer conditional mixed markov model. IEEE Transactions on Geoscience and Remote Sensing 47, 3416–3430.

Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H., 2016. Fully-convolutional siamese networks for object tracking, in: European Conference on Computer Vision, Springer. pp. 850–865.

Bourdis, N., Denis, M., Sahbi, H., 2011. Constrained optical flow for aerial image change detection, in: International Geoscience and Remote Sensing Symposium, pp. 4176–4179.

Bovolo, F., Bruzzone, L., 2005. A wavelet-based change-detection technique for multitemporal sar images, in: International Workshop on the Analysis of Multi-Temporal Remote Sensing Images, IEEE. pp. 85–89.

Bovolo, F., Bruzzone, L., 2007. A theoretical framework for unsupervised change detection based on change vector analysis in the polar domain. IEEE Transactions on Geoscience and Remote Sensing 45, 218–236.

Bruzzone, L., Bovolo, F., 2013. A novel framework for the design of change-detection systems for very-high-resolution remote sensing images. Proceedings of the IEEE 101, 609–630.

Bruzzone, L., Prieto, D.F., 2000. Automatic analysis of the difference image for unsupervised change detection. IEEE Transactions on Geoscience and Remote Sensing 38, 1171–1182.

Celik, T., 2009. Unsupervised change detection in satellite images using principal component analysis and k-means clustering. IEEE Geoscience and Remote Sensing Letters 6, 772–776.

Chen, K., Weinmann, M., Sun, X., Yan, M., Hinz, S., Jutzi, B., Weinmann, M., 2018a. Semantic segmentation of aerial imagery via multi-scale shuffling convolutional neural networks with deep supervision. ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences 4.

Chen, Y., Ouyang, X., Agam, G., 2018b. MFCNET: End-to-end approach for change detection in images, in: IEEE International Conference on Image Processing, IEEE. pp. 4008–4012.

Chopra, S., Hadsell, R., LeCun, Y., 2005. Learning a similarity metric discriminatively, with application to face verification, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 539–546.

Coppin, P., Jonckheere, I., Nackaerts, K., Muys, B., Lambin, E., 2004. Digital change detection methods in

ecosystem monitoring: A review. International Journal of Remote Sensing 25, 1565–1596.

Dai, X., Khorram, S., 1999. Remotely sensed change detection based on artificial neural networks. Photogrammetric engineering and remote sensing 65, 1187–1194.

Daudt, R.C., Le Saux, B., Boulch, A., 2018a. Fully convolutional siamese networks for change detection, in: International Conference on Image Processing, 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: International Geoscience and Remote Sensing Symposium, pp. 2119–2122.

Demir, I., Koperski, K., Lindenbaum, D., Pang, G., Huang, J., Basu, S., Hughes, F., Tuia, D., Raskar, R., 2018. Deepglobe 2018: A challenge to parse the earth through satellite images. CoRR abs/1805.06561. URL: http://arxiv.org/abs/1805.06561.

El Amin, A.M., Liu, Q., Wang, Y., 2016. Convolutional neural network features based change detection in satellite images, in: First International Workshop on Pattern Recognition, International Society for Optics and Photonics. p. 100110W.

El Amin, A.M., Liu, Q., Wang, Y., 2017. Zoom out cnns features for optical remote sensing change detection, in: Int. Conference on Image, Vision and Computing, pp. 812–817.

Gopal, S., Woodcock, C., 1996. Remote sensing of forest change using artificial neural networks. IEEE Transactions on Geoscience and Remote Sensing 34, 398–404.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

Huang, C., Song, K., Kim, S., Townshend, J.R., Davis, P., Masek, J.G., Goward, S.N., 2008. Use of a dark object concept and support vector machines to automate forest cover change analysis. Remote Sensing of Environment 112, 970–985.

Hussain, M., Chen, D., Cheng, A., Wei, H., Stanley, D., 2013. Change detection from remotely sensed images: From pixel-based to object-based approaches. ISPRS Journal of Photogrammetry and Remote Sensing 80, 91–106.

Lambin, E.F., Strahlers, A.H., 1994. Change-vector analysis in multitemporal space: a tool to detect and categorize land-cover change processes using high temporal-resolution satellite data. Remote Sensing of Environment 48, 231–244.

Le Saux, B., Randrianarivo, H., 2013. Urban change detection in SAR images by interactive learning, in: International Geoscience and Remote Sensing Symposium, IEEE. pp. 3990–3993.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436.

Liu, G., Gousseau, Y., Tupin, F., 2019. A contrario comparison of local descriptors for change detection in very high spatial resolution satellite images of urban areas. IEEE Transactions on Geoscience and Remote Sensing .

Liu, J., Gong, M., Qin, K., Zhang, P., 2016. A deep convolutional coupling network for change detection based on heterogeneous optical and radar images. IEEE Transactions on Neural Networks and Learning Systems 29, 545–559.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440.

Maggiolo, L., Marcos, D., Moser, G., Tuia, D., 2018. Improving maps from cnns trained with sparse, scribbled ground truths using fully connected crfs, in: International Geoscience and Remote Sensing Symposium, IEEE. pp. 2103–2103.

Maggiori, E., Tarabalka, Y., Charpiat, G., Alliez, P., 2017. High-resolution image classification with convolutional networks, in: International Geoscience and Remote Sensing Symposium, IEEE. pp. 5157–5160.

Mnih, V., Hinton, G.E., 2010. Learning to detect roads in high-resolution aerial images, in: European Conference on Computer Vision, pp. 210–223.

Mou, L., Bruzzone, L., Zhu, X.X., 2019. Learning spectral-spatial-temporal features via a recurrent convolutional neural network for change detection in multispectral imagery. IEEE Transactions on Geoscience and Remote Sensing 57, 924–935.

Rolnick, D., Veit, A., Belongie, S.J., Shavit, N., 2017. Deep learning is robust to massive label noise. CoRR abs/1705.10694. URL: http://arxiv.org/abs/1705.10694.

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, pp. 234–241.

Rosin, P.L., Ioannidis, E., 2003. Evaluation of global image thresholding for change detection. Pattern Recognition Letters 24, 2345–2356.

Sesnie, S.E., Gessler, P.E., Finegan, B., Thessler, S., 2008. Integrating landsat tm and srtm-dem derived variables with decision trees for habitat classification and change detection in complex neotropical environments. Remote Sensing of Environment 112, 2145–2159.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations.

Singh, A., 1989. Review article digital change detection techniques using remotely-sensed data. International Journal of Remote Sensing 10, 989–1003.

Stent, S., Gherardi, R., Stenger, B., Cipolla, R., 2015. Detecting change for multi-view, long-term surface inspection, in: British Machine Vision Conference, pp. 127.1–127.12.

Vakalopoulou, M., Karantzalos, K., Komodakis, N., Paragios, N., 2015. Simultaneous registration and change detection in multitemporal, very high resolution remote sensing data, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 61–69.

Volpi, M., Tuia, D., 2017. Dense semantic labeling of subdecimeter resolution images with convolutional neural networks. IEEE Transactions on Geoscience and Remote Sensing 55, 881–893.

Volpi, M., Tuia, D., Bovolo, F., Kanevski, M., Bruzzone, L., 2013. Supervised change detection in vhr images using contextual information and support vector machines. International Journal of Applied Earth Observation and Geoinformation 20, 77–85.

Volpi, M., Tuia, D., Kanevski, M., Bovolo, F., Bruzzone, L., 2009. Supervised change detection in vhr images: a comparative analysis, in: 2009 IEEE International Workshop on Machine Learning for Signal Processing, IEEE. pp. 1–6.

Zagoruyko, S., Komodakis, N., 2015. Learning to compare image patches via convolutional neural networks, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 4353–4361.

Zhan, Y., Fu, K., Yan, M., Sun, X., Wang, H., Qiu, X., 2017. Change detection based on deep siamese convolutional network for optical aerial images. IEEE Geoscience and Remote Sensing Letters 14, 1845–1849.

Zhao, J., Gong, M., Liu, J., Jiao, L., 2014. Deep learning to classify difference image for image change detection, in: International Joint Conference on Neural Networks, IEEE. pp. 411–417.