Guided Anisotropic Diffusion and Iterative Learning for Weakly Supervised Change Detection

Rodrigo Caye Daudt[1,2] Bertrand Le Saux[1] Alexandre Boulch[1] Yann Gousseau[2]
[1] ONERA [2] Télécom ParisTech
[firstname.lastname]@[onera/telecom-paristech].fr

Abstract

Large scale datasets created from user labels or openly available data have become crucial to provide training data for large scale learning algorithms. While these datasets are easier to acquire, the data are frequently noisy and unreliable, which is motivating research on weakly supervised learning techniques. In this paper we propose an iterative learning method that extracts the useful information from a large scale change detection dataset generated from open vector data to train a fully convolutional network which surpasses the performance obtained by naive supervised learning. We also propose the guided anisotropic diffusion algorithm, which improves semantic segmentation results using the input images as guides to perform edge preserving filtering, and is used in conjunction with the iterative training method to improve results.

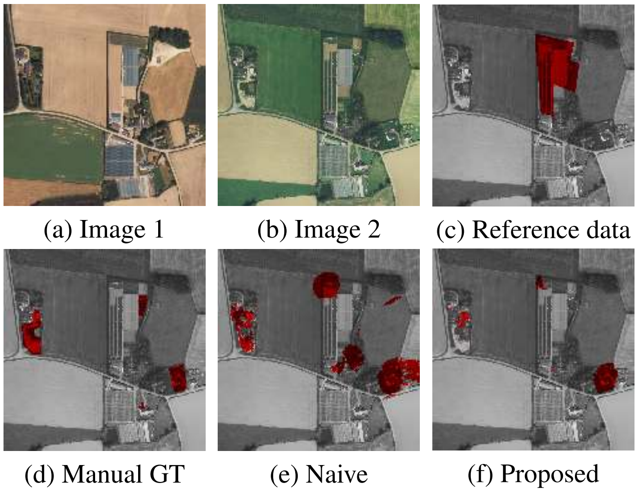

(a) Image 1 (b) Image 2 (c) Reference data
(d) Manual GT (e) Naive (f) Proposed

Figure 1. (a)-(b) image pair, (c) change labels from the HRSCD dataset, (d) ground truth created by manually annotating changes, (e) result obtained by naive supervised training, (f) result obtained by our proposed method.

1. Introduction

Change detection (CD) is one of the oldest problems studied in the field of remote sensing image analysis [19, 40]. It consists of comparing a pair or sequence of coregistered images and identifying the regions where meaningful changes have taken place between the first and last acquisitions. However, the definition of meaningful change varies depending on the application. Changes of interest are, for example, new buildings and roads, forest fires, and growth or shrinkage of water bodies for environmental monitoring. Although exceptions exist, such as object based methods, most change detection algorithms predict a change label for each pixel in the provided images by modelling the task mathematically as a segmentation or clustering problem.

Many variations of convolutional neural networks (CNNs) [29], notably fully convolutional networks (FCNs) [30], have recently achieved excellent performances in change detection tasks [7, 8, 15]. These methods require large amounts of training data to perform supervised training of the proposed networks [28]. Open labelled datasets for change detection are extremely scarce and are predominantly very small compared to labelled datasets in other computer vision areas. Benedek and Szirnyi [2] created the Air Change dataset which contain about 8 million labelled pixels, divided into three regions. Daudt *et al.* created the OSCD [4] dataset from Sentinel-2 multispectral images, with a total of about 9 million labelled pixels. While these datasets allow for simple models to be trained in a supervised manner, training more complex models with these data would lead to overfitting.

The recently proposed High Resolution Semantic Change Detection (HRSCD) dataset [8] is the first large scale change detection dataset. By combining an aerial image database with open change and land cover data, change maps and land cover maps were generated for almost 30 billion pixels, over 3000 times larger than previous change detection datasets. This dataset, however, contains unreliable labels due to having been generated automatically. The effect of naively using these data for supervised learning of change detection networks is shown in Fig. 1. Inaccuracies in the reference data stem primarily from two causes: imperfections in the vector data, and temporal misalignment between the annotations and the images. Naive supervision using such data leads to overestimation of the detected changes, as can be seen in Fig. 1(e). Nevertheless, there is much useful information in the available annotations that, if used adequately, can lead to better CD systems.

Due to the way the ground truth was generated, the labels in the dataset mark changes at a land parcel level with imprecise boundaries. While useful for global monitoring of changes in land cover, it cannot delineate precise object-level changes. In order to achieve a precise pixel-wise change detection, we propose a weakly supervised learning approach to change detection. We consider the parcel-wise reference data as approximations, similar to bounding-boxes, of an ideal unknown ground truth corresponding to changes at pixel level. For each parcel with detected changes, the reference data in HRSCD contained both good and bad labels. For this reason, the noise in the labels is not randomly distributed, but it is conditioned on the pixels' neighborhoods and highly structured.

We propose a weakly supervised approach to change detection that improves on previously proposed methods for semantic segmentation. We present a training scheme that harnesses the useful information in the HRSCD dataset for parcel-wise change detection, attempting to refine the reference data while training a fully convolutional network. By acknowledging the presence of incorrect labels in the training dataset (with respect to our fine grained objective), we are able to select good data and ignore bad ones, improving the final results as seen in Fig. 1(f). A preliminary version of this idea has been proposed in [3]. This paper's new contributions include detailed equations and algorithms, integration with image-guided processing methods, and quantitative evaluation of the proposed methods.

This paper describes two main contributions to this problem. The first one is an iterative training scheme that alternates between training a fully convolutional network for change detection and using this network to find bad examples in the training set. The second main contribution is the Guided Anisotropic Diffusion (GAD) algorithm, which is used in the iterative training scheme to better fit semantic segmentation predictions to the input images. The proposed GAD algorithm is not restricted to change detection and can be used as a post-processing technique to improve semantic segmentation algorithms.

2. Related Work

Change detection has a long history, being one of the early problems tackled in remote sensing image understanding [40]. It is done using coregistered image pairs or sequences, and consists of identifying areas in the images that have experienced significant modifications between the acquisitions. Many of the state-of-the-art ideas in pattern recognition have been used for change detection in the past, from pixel-level comparison of images, to superpixel segmentation, object-level image analysis, and image descriptors [19]. In this paper we treat change detection as a two class semantic segmentation problem, in which a label is predicted for each pixel in the input images. With the rise of machine learning algorithms for semantic segmentation, notably convolutional neural networks, many algorithms have attempted to learn to perform change detection. Most algorithms circumvented the problem of scarcity of training data through transfer learning by using pretrained networks to generate pixel descriptors [39, 10, 11]. Fully convolutional networks trained end-to-end to perform change detection have recently been proposed by several authors independently, usually using Siamese architectures [43, 7, 8, 5, 15].

Semantic segmentation algorithms attempt to understand an input image and predict to which class among a known set of classes each pixel in an input image belongs. Change detection is modelled in this paper and many others as a semantic segmentation problem which takes as input two or more images. Long *et al.* proposed the first fully convolutional network for semantic segmentation, which achieved excellent performance and inference speed [30]. Since then, several improvements have been proposed for CNNs and FCNs. Ioffe and Szegedy have proposed batch normalization layers, which normalize activations and help avoid the vanishing/exploding gradient problem while training deep networks [20]. Ronneberger *et al.* proposed the usage of skip connections that transfer details and boundary information from earlier to later layers in the network, which improves the accuracy around the edges between semantic regions [38]. He *et al.* proposed the idea of residual connections, which have improved the performance of CNNs and FCNs and made it easier to train deep networks [18].

Noisy labels for supervised learning is a topic that has already been widely explored [13, 14]. In many cases, label noise is completely random and independent from the data, and is modelled mathematically as such [33, 42, 37]. Rolnick *et al.* showed that supervised learning algorithms are robust to random label noise, and proposed strategies to further minimize the effect label noise has on training, such as increasing the training batch sizes [37]. In the case presented in this paper, the assumption that the label noise is random does not hold. Incorrect change detection labels are usually around edges between regions or grouped together, which leads the network to learn to overestimate detected changes as seen in Fig. 1(e). Ignoring part of the training dataset, known as data cleansing (or cleaning), has already been proposed in different contexts [32, 22, 16, 21].

Weakly supervised learning is the name given to the group of machine learning algorithms that aim to perform different or more complex tasks than normally allowed by the training data at hand. Weakly supervised algorithms have recently gained popularity because they provide an alternative when data acquisition is too expensive. The problem of learning to perform semantic segmentation using only bounding box data or image level labels is closely related to the task discussed in this paper, since most methods propose the creation of an approximate semantic segmentation ground truth for training and dealing with its imperfections accordingly. Dai *et al*. proposed the BoxSup algorithm [6] where region proposal algorithms are used to generate region candidates in each bounding box, then a semantic segmentation network is trained using these annotations, and finally it is used to select better region proposal candidates iteratively. Khoreva *et al*. proposed improvements to the BoxSup algorithm that includes using *ad hoc* heuristics and an ignore class during training [23]. They obtained best results using region proposal algorithms to create semantic segmentation training data directly from bounding boxes. Lu *et al*. modelled this problem as a simultaneous learning and denoising task through a convex optimization problem [31]. Ahn and Kwak proposed combining class activation maps, random walk and a learned network that predicts if pixels belong to the same region to perform semantic segmentation from image level labels [1].

Post-processing methods that use information from guide images to filter other images, such as semantic segmentation results, have also been proposed [36, 26, 12]. A notable example is the Dense CRF algorithm proposed by Krähenbühl and Koltun, in which an efficient solver is proposed for fully connected conditional random fields with Gaussian edge potentials [27]. The idea of using a guide image for processing another is also the base of the Guided Image Filtering algorithm proposed by He *et al*. [17], where a linear model that transforms a guide image into the best approximation of the filtered image is calculated, thus transferring details from the guide image to the filtered image. The use of joint filtering is popular in the field of computational photography, and has been used for several applications [36, 26, 12]. One of the building blocks of the filtering method we propose in this paper is the anisotropic diffusion, proposed by Perona and Malik [35], an edge preserving filtering algorithm in which the filtering of an image is modelled as a heat equation with a different diffusion coefficient at each edge between neighbouring pixels depending on the local geometry and contrast. However, to the best of our knowledge, this algorithm has not yet been used for guided filtering.

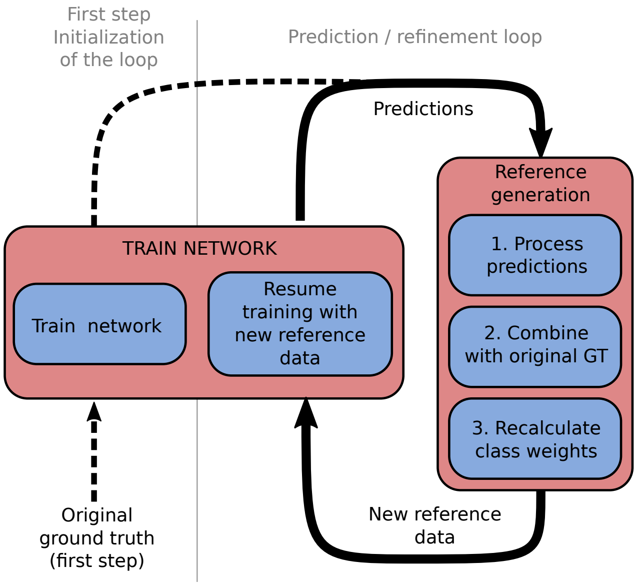


Figure 2. Iterative training method: alternating between training and data cleaning allows the network to simultaneously learn the desired task and to remove bad examples from the training dataset.

3. Method

The two main contributions of this paper are: 1) an iterative training scheme that aims to efficiently learn from inaccurate and unreliable ground truth semantic segmentation data and 2) the guided anisotropic diffusion algorithm, which uses information from the input images to filter and improve semantic segmentation results. These contributions are presented in sections 3.1 and 3.2 below, respectively. While these two ideas are presented in this paper in the context of change detection, the proposed methods' scope is broader and could be used for other semantic segmentation problems, together or separately.

3.1. Iterative Training Scheme

The label noise present in the HRSCD dataset for change detection is challenging due to its spatial structure and correlation between neighbors. In the taxonomy presented in [13, 14], this type of label noise would be classified as ”label noise not at random” (NNAR). NNAR is the most complex among the label noise models in the taxonomy. In the case of HRSCD, most errors can be attributed to one of the following reasons: the available information is insufficient to perform labelling, errors on the part of the annotators, subjectiveness of the labelling task, and temporal misalignment between the databases used to create the HRSCD dataset.

It is important to note that, as discussed by Frénay and Kabán in [13], label noise has an even more powerful damaging impact when a dataset is imbalanced since it alters the perceived, but not the real, class imbalance and therefore the methods used to mitigate class imbalance during training

Algorithm 1 Iterative training pseudocode.

```
Input: $I$: Image pairs, $GT_o$: Original unreliable ground truths, $N$: Number of hyperepochs, $\Phi_r$: Initial random network weights.
Output: $\Phi_N$: Trained network weights.
$w_0 \leftarrow$ calculate class weights inversely proportional to number of class examples
$\Phi_0 \leftarrow$ Train network with $I$ and $GT_0$ until convergence or fixed number of epochs
for ($i \leftarrow 1; i \leq N; i++$) do
    $P_i \leftarrow$ generate predictions for training dataset with current network
    $P_{i,pp} \leftarrow$ Post-processing of predictions
    $GT_i \leftarrow$ Combine $P_{i,pp}$ with $GT_0$ to generate cleaner ground truth data
    $\Phi_i \leftarrow$ Continue training network from $\Phi_{i-1}$ using $I$ and $GT_i$ until convergence
end for
```

		Orig. GT	
		0	**1**
Pred.	**0**	0	0
	1	0	1

(a) Intersection

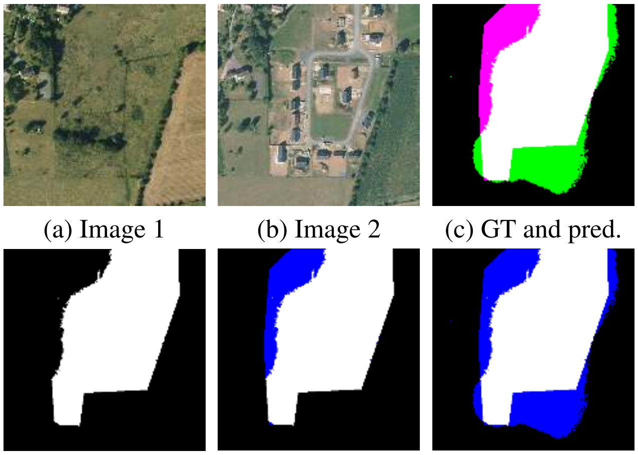
		Orig. GT	
		0	**1**
Pred.	**0**	0	2
	1	0	1

(b) FN← Ignore

		Orig. GT	
		0	**1**
Pred.	**0**	0	2
	1	2	1

(c) FN∪FP← Ignore

Figure 3. Proposed methods for merging original labels and network predictions. Classes: 0 is no change, 1 is change, 2 is ignore. (a) Intersection between original and detected changes. (b) Ignore false negatives from the perspective of original labels. (c) Ignore all pixels with label disagreements.



(a) Image 1 (b) Image 2 (c) GT and pred.

(d) Intersection (e) FN← Ignore (f) FN∪FP← Ign.

Figure 4. Example case of the three proposed merge strategies. In (c), black is true negative, white is true positive, magenta is false negative, and green is false positive. In (d)-(f) blue represents the ignore class.

are less effective. In the case of change detection with the HRSCD dataset, the no change class outnumbers the change class 130 to 1, which means the label noise could significantly alter the calculated class weights used for training.

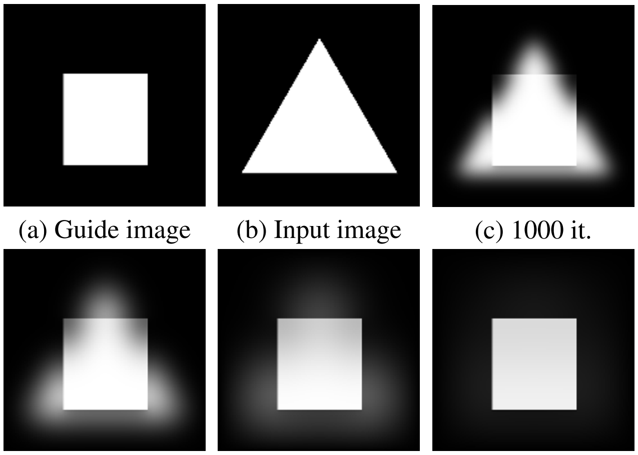
It has been noticed in [8] and in our own experiments that change detection networks trained directly on the HRSCD dataset had the capacity to detect changes in image pairs but tended to predict blobs around the detected change instances, as is depicted in Fig. 6(c), likely in an attempt to minimize the loss for the training images where the surrounding pixels of true changes are also marked as having experienced changes. In many cases, it was observed that the network predictions were correct where the ground truth labels were not. Based on this observation, we propose a method for training the network that alternates between actual minimization of a loss function and using the network predictions to clean the reference data before continuing the training. A schematic that illustrates the main ideas of this method is shown in Fig. 2. For the remainder of this paper, the iteration cycles of training the network and cleaning of training data will be referred to as *hyperepochs*.

Alternating between training a semantic segmentation network and using it to make changes to the training data has already been explored [6, 23]. Such iterative methods are named ”classification filtering” [14]. The main differences between the method proposed in this paper and previous ones are:

1. **No bounding box information is available**: we work directly with pixel level annotations, which were generated form vector data;
2. **Each annotated region may contain more than one instance**: the annotations often group several change instances together;
3. **Annotations are not flawless**: the HRSCD dataset contains both false positives and false negatives in change annotations.

It has also been shown by Khoreva *et al.* in [23] that simply using the outputs of the network as training data leads to degradation of the results, and that it is necessary to use priors and heuristics specific to the problem at hand to prevent a degradation in performance. In this paper we use two ways to avoid degradation of the results with iterative training. The first is using processing techniques that bring information from the input images into the predicted semantic segmentations, improving the results and providing a stronger correlation between inputs and predictions. The Guided Anisotropic Diffusion algorithm presented in Section 3.2 serves this purpose, but other algorithms such as Dense CRF [27] may also be used. The second way the degradation of results is avoided is by combining network predictions with the original reference data at each iteration, instead of simply using predictions as reference data.

We propose three ways of merging the original labels


(a) Guide image (b) Input image (c) 1000 it.

(d) 3000 it. (e) 10000 it. (f) 30000 it.

Figure 5. Results of guided anisotropic diffusion. Edges in the guide image (a) are preserved in the filtered image (b). (c)-(f) show results using different numbers of iterations.

with network predictions. When merging, each pixel will have a binary label from the original ground truth and a binary label from the network prediction. If these labels agree, there is no reason to believe the label for that pixel is wrong, and it is therefore kept unchanged. In case the labels disagree, the following options to decide the pixel's label are proposed:

1. **The intersection of predicted and reference change labels is kept as change**: this strategy assumes all changes are marked in both the reference data and in the prediction. It also puts pixels with uncertain labels in the no change class, where they are more easily diluted during training due to the class imbalance.

2. **Ignore false negatives**: using an ignore class for false negatives attempts to keep only good examples in the change class, improving the quality of the training data. It assumes all changes are marked in the original labels provided.

3. **Ignore all disagreements**: marking all label disagreements to be ignored during training attempts to keep only clean labels for training at the cost of reducing the number of training examples. This approach is the only one that is class agnostic.

In practice, the ignored pixels are marked as a different class that is given a class weight of 0 during the training. Tables for the three proposed methods can be found in Fig. 3.1, and an example can be found in Fig. 4.

3.2. Guided Anisotropic Diffusion

In their seminal paper, Perona and Malik proposed an anisotropic diffusion algorithm with the aim of performing scale space image analysis and edge preserving filtering [35]. Their diffusion scheme has the ability to blur the inside of regions with homogeneous colours while preserving or even enhancing edges. This is done by modelling the filtering as a diffusion equation with spatially variable coefficients, and as such is an extension of the linear heat equation, whose solution is mathematically equivalent to Gaussian filtering when diffusion coefficients are constant [25]. Diffusion coefficients are set to be higher where the local contrast of the image is lower.

More precisely, we consider the anisotropic diffusion equation

$$\frac{\partial I}{\partial t} = div(c(x,y,t)\nabla I) = c(x,y,t)\Delta I + \nabla c \cdot \nabla I \quad (1)$$

where I is the input image, $c(x,y,t)$ is the coefficient diffusion at position (x,y) and time t, div represents the divergence, ∇ represents the gradient, and Δ represents the Laplacian. In its original formulation, $c(x,y,t)$ is a function of the input image I. To perform edge preserving filtering, one approach is using the coefficient

$$c(x,y,t) = \frac{1}{1 + \left(\frac{||\nabla I(x,y,t)||}{K}\right)^2}, \quad (2)$$

which approaches 1 (strong diffusion) where the gradient is small, and approaches 0 (weak diffusion) for large gradient values. Other functions with these properties and bound in $[0,1]$ may also be used. The parameter K controls the sensitivity to contrast in the image.

In the guided anisotropic diffusion algorithm the aim is to perform edge preserving filtering on an input image, but instead of preserving the edges in the filtered image we preserve edges coming from a separate guide image (or images). Doing so allows us to transfer properties from the guide image I_g into the filtered image I_f. An illustrative example is shown in Fig. 5, where the image of a rectangle (a) is used as a guide to filter the image of a triangle (b). The edges from the guide image I_g are used to calculate $c(x,y,t)$, which in practice creates barriers in the diffusion of the filtered image I_f, effectively transferring details from I_g to I_f. These edges effectively separate the image in two regions, inside and outside the rectangle, and the gray values in each of these regions experience diffusion, but there is virtually no diffusion happening between them.

Our aim is to use this guided anisotropic diffusion (GAD) algorithm to improve semantic segmentation results based on the input images. Given that the change detection networks trained on the HRSCD dataset have the tendency to overestimate the area of the detected changes, GAD provides a way to improve these semantic segmentation results by making them more precisely fit the edges present in the input images. A few design choices were made to extend the

Algorithm 2 Guided Anisotropic Diffusion pseudocode.

```
**Input:**$I_1$, $I_2$, $I_in$, $N$, $K$, $\lambda$
**Output:**$I_f$
$I_f \leftarrow I_in$
**for** ($i \leftarrow 1; i \leq N; i++$) **do**
    **for** ($I_j = \{I_1, I_2\}$) **do**
        $\nabla I_j \leftarrow$ Calculate gradient of $I_j$
        $c_{I_j} \leftarrow$ Calculate using Eq. 3
        $I_j \leftarrow I_j + \lambda \cdot \nabla I_j \cdot c_{I_j}$
    **end for**
    $\nabla I_f \leftarrow$ Calculate gradient of $I_f$
    $c_f \leftarrow$ Calculate using Eq. 4
    $I_f \leftarrow I_f + \lambda \cdot \nabla I_f \cdot c_f$
**end for**
```

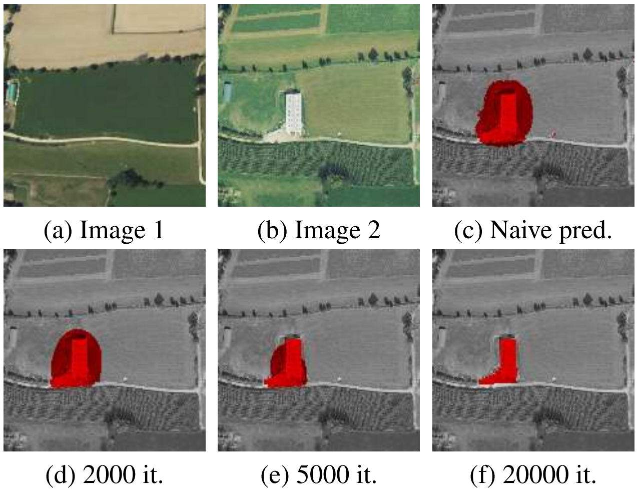

(a) Image 1
(b) Image 2
(c) Naive pred.
(d) 2000 it.
(e) 5000 it.
(f) 20000 it.

Figure 6. Guided anisotropic diffusion for filtering a real example of semantic segmentation. The diffusion allows edges from the guide images to be transferred to the target image, improving the results.

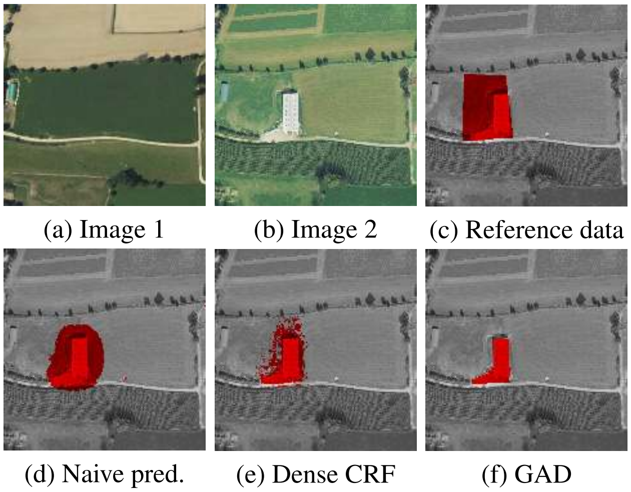

(a) Image 1
(b) Image 2
(c) Reference data
(d) Naive pred.
(e) Dense CRF
(f) GAD

Figure 7. Comparison between (c) original dataset ground truth, (e) prediction filtered by Dense CRF, and (f) prediction filtered with guided anisotropic diffusion for 20000 iterations.

anisotropic diffusion from gray level images to RGB image pairs. The extension to RGB image was done by taking the mean of the gradient norm at each location

$$c_I(x,y,t) = \frac{1}{1+\left(\sum_{C\in\{R,G,B\}} \frac{||\nabla I_C(x,y,t)||}{3\cdot K}\right)^2}, \tag{3}$$

so that edges in any of the color channels would prevent diffusion in the filtered image. To extend this further to be capable of taking multiple guide images simultaneously, which is necessary for the problem of change detection, the minimum diffusion coefficient at each position (x, y, t) was used, once again to ensure that any edge present in any guide image would be transferred to the filtered image:

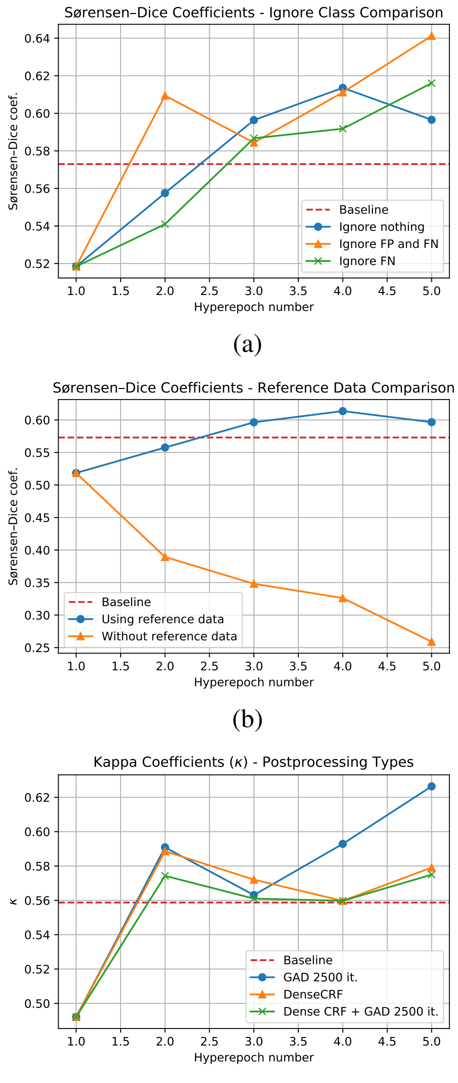
$$c_{I_1,I_2}(x,y,t) = min_{i\in\{1,2\}} c(I_i)(x,y,t). \tag{4}$$

Guided anisotropic diffusion aims to improve semantic segmentation predictions by filtering the class probabilities yielded by a fully convolutional network. It is less adequate to correct for large classification mistakes, as opposed to non-local methods such as Dense CRF, but it leads to smoother predictions with more accurate edges. It can also be easily extended for any number of guide images by increasing the number of images considered in Eq. 4. The pseudocode for the GAD algorithm can be found in Alg. 2. As mentioned in the original anisotropic diffusion paper, the algorithm is unstable for $\lambda > 0.25$ when using 4-neighborhoods for the calculations.

4. Experiments

To validate the methods proposed in Section 3 we adopted the hybrid change detection and land cover mapping fully convolutional network presented in [8], since it was already proven to work with the HRSCD dataset. We adopted *strategy 4.2* described in the paper, in which the land cover mapping branches of the network are trained before the change detection one to avoid setting a balancing hyperparameter. The land cover mapping branches of the network were fixed to have the same parameter weights for all tests presented in this paper, and evaluating those results is not done here as the scope of this paper is restricted to the problem of change detection.

We applied the GAD algorithm to the predictions from a network trained directly on the reference data from HRSCD to evaluate its performance. In Fig. 6 there is an example of the obtained results. As noted before, we can see in (c) that the change is detected but unchanged pixels around it are also classified as changes by the network. In (d)-(f) it can be clearly seen how the GAD algorithm improves the results by diffusing the labels across similar pixels while preserv-



(a)

(b)

(c)

Figure 8. Ablation studies. (a) Comparison between strategies for merging network predictions and reference data. (b) Comparison between iterative training with and without the usage of original reference data. (c) Comparison between GAD and Dense CRF.

ing edges from the input images in the semantic segmentation results. As expected, more iterations of the algorithm lead to a stronger erosion of incorrect labels. For these results and all others in this section, GAD was applied with $K = 5$ and $\lambda = 0.24$. In Fig. 7 we can see a comparison between GAD and the Dense CRF[1] algorithm [27]. While the non-local nature of fully connected CRFs is useful in some cases, we can see the results are less precise and significantly noisier than the ones obtained by using GAD.

To perform quantitative analysis of results, it would be meaningless to use the test data in the HRSCD dataset given that we are attempting to perform a task which is not the one for which ground truth data are available, *i.e.* we are attempting to perform pixel-level precise change detection and not parcel-level change detection. For this reason we have manually annotated the changes as precisely as possible for two 10000x10000 image pairs in the dataset, for a total of $2{\cdot}10^8$ test pixels, or 50 km^2. The image pairs were chosen before any tests were made to avoid biasing the results. Due to the class imbalance, total accuracy, *i.e.* the percentage of correctly classified pixels, provides us with a skewed view of the results biased towards the performance on the class more strongly represented. Therefore, the Sørensen-Dice coefficient (equivalent to the F1 score for binary problems) from the point of view of the change class was used [9, 41]. The Sørensen-Dice coefficient score is defined as

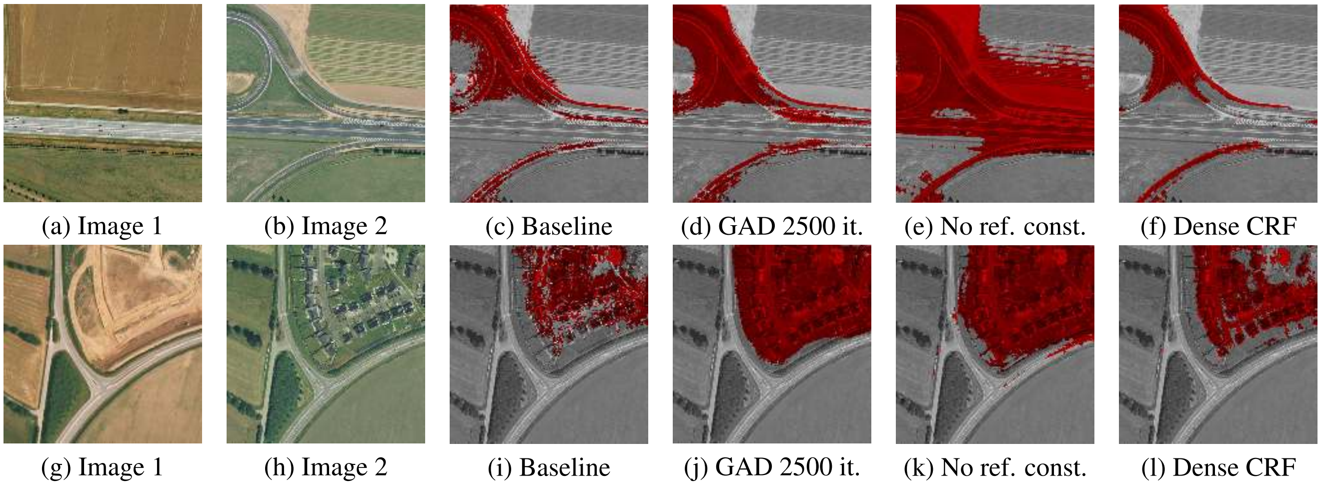
$$Dice = (2 \cdot TP)/(2 \cdot TP + FP + FN) \qquad (5)$$

where TP means true positive, FP means false positive, and FN means false negative. It serves as a balanced measurement of performance even for unbalanced data.

All tests presented here were done using PyTorch [34]. At each hyperepoch, the network was trained for 100 epochs with an ADAM algorithm for stochastic optimization [24], with learning rate of 10^{-3} for the first 75 epochs and 10^{-4} for the other 25 epochs. The tests show the performance of networks trained with the proposed method for 5 hyperepochs (iterations of training and cleaning the data), where the first one is done directly on the available data from the HRSCD dataset. For accurate comparison of methods and to minimize the randomness in the comparisons, the obtained network at the end of hyperepoch 1 is used as a starting point for all the methods. This ensures all networks have the same initialization at the point in the algorithm where they diverge. A baseline network was trained for the same amount of epochs and hyperepochs but with no changes done to the training data. This serves as a reference point as to the performance of the fully convolutional network with no weakly supervised training methods.

The first comparison, shown in Fig. 8(a), compares the three methods proposed in Section 3.1 to combine the network predictions with the original ground truth from the HRSCD dataset. We notice that all three strategies surpass the baseline network using the proposed iterative training method, which validates the ideas presented earlier. In Fig. 8(b) we see a comparison between a training using the full training scheme proposed in this paper (without the usage of an ignore class) and the same method but without using the original reference data, *i.e.* using only network predictions processed by GAD to continue training at each hyperepoch. Our results, which corroborate the ones in [23], show that referring back to the original data at each hyperepoch is essential to avoid a degradation in performance.

[1] https://github.com/lucasb-eyer/pydensecrf


(a) Image 1 (b) Image 2 (c) Baseline (d) GAD 2500 it. (e) No ref. const. (f) Dense CRF

(g) Image 1 (h) Image 2 (i) Baseline (j) GAD 2500 it. (k) No ref. const. (l) Dense CRF

Figure 9. Change maps obtained by using different methods on two image pairs. Detected changes are marked in red color.

In Fig. 8(c) we show a comparison between using the proposed GAD algorithm versus the Dense CRF [27] algorithm in the iterated training procedure, as well as using both together. We see that using the Dense CRF algorithm to process predictions leads to good performance in early hyperepochs, but is surpassed by GAD later on. This is likely explained by the non local nature of Dense CRF and its ability to deal with larger errors, but its inferior performance relative to GAD for finer prediction errors.

Figure 9 shows the predictions by networks trained by different methods on two example images. We see that the best results are obtained by using the full training scheme with GAD in (d)/(j), followed by Dense CRF, which also achieves good results shown in (f)/(l). The baseline results in (c)/(i), obtained by naively training the network in a supervised manner, and the ones without using the reference data as constraint in the iterative training scheme shown in (e)/(k) are significantly less accurate than those using GAD or Dense CRF.

5. Analysis

One possible criticism of the proposed iterative training method is that it would get rid of hard and important examples in the training dataset. It is true that the performance of this weakly supervised training scheme would likely never reach that of one supervised with perfectly clean data, but the results in Section 4 show that using the proposed method we can consistently train networks that perform better than those naively trained with noisy data directly.

The results also made clear that it is of paramount importance to refer back to the ground truth data every time the training ground truth is being modified. Not doing so leads to a fast degradation in performance, since the network simply attempts to learn to copy itself and stops learning useful operations from the data. The results also showed that separating dubiously labelled pixels leads to a small increase in performance, likely due to the fact that we end up providing a cleaner and more trustworthy dataset at training time.

The guided anisotropic diffusion algorithm was compared against the Dense CRF algorithm for using information from the input images to improve semantic segmentation results. While both algorithms were successful when used in the proposed iterative training scheme, GAD outperformed Dense CRF at later hyperepochs for quantitative metrics. Both algorithms yielded visually pleasing results, each performing better in different test cases.

6. Conclusion

In this paper we have proposed an iterative training method for training with noisy data that alternates between training a fully convolutional network and leveraging its predictions to clean the training dataset from mislabelled examples. We showed that the proposed method outperforms naive supervised training using the provided reference data for change detection. We proposed three methods for merging network predictions with reference data, the best of which aimed to ignore suspiciously labelled examples. Our results corroborated previous results which stated that referring back to reference data when performing classification filtering for data cleaning. We also proposed the guided anisotropic diffusion algorithm for improving semantic segmentation results by performing a cross image edge preserving filtering. The GAD algorithm was used in conjunction with the iterative training method to obtain the best results in our tests. The GAD algorithm was compared against the Dense CRF algorithm, and was found to be superior in performance when used with the proposed iterative training scheme.

References

[1] J. Ahn and S. Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4981–4990, 2018. 3

[2] C. Benedek and T. Szirányi. Change detection in optical aerial images by a multilayer conditional mixed markov model. *IEEE Transactions on Geoscience and Remote Sensing*, 47(10):3416–3430, 2009. 1

[3] R. Caye Daudt, A. Chan-Hon-Tong, B. Le Saux, and A. Boulch. Learning to understand earth observation images with weak and unreliable ground truth. In *International Geoscience and Remote Sensing Symposium (Accepted)*. IEEE, 2019. 2

[4] R. Caye Daudt, B. Le Saux, A. Boulch, and Y. Gousseau. Urban change detection for multispectral earth observation using convolutional neural networks. In *International Geoscience and Remote Sensing Symposium*, pages 2119–2122. IEEE, 2018. 1

[5] Y. Chen, X. Ouyang, and G. Agam. MFCNET: End-to-end approach for change detection in images. In *2018 25th IEEE International Conference on Image Processing*, pages 4008–4012. IEEE, 2018. 2

[6] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1635–1643, 2015. 3, 4

[7] R. C. Daudt, B. Le Saux, and A. Boulch. Fully convolutional siamese networks for change detection. In *2018 25th IEEE International Conference on Image Processing*, pages 4063–4067, October 2018. 1, 2

[8] R. C. Daudt, B. Le Saux, A. Boulch, and Y. Gousseau. High resolution semantic change detection. *CoRR*, abs/1810.08452, 2018. 1, 2, 4, 6

[9] L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945. 7

[10] A. M. El Amin, Q. Liu, and Y. Wang. Convolutional neural network features based change detection in satellite images. In *First International Workshop on Pattern Recognition*, pages 100110W–100110W. International Society for Optics and Photonics, 2016. 2

[11] A. M. El Amin, Q. Liu, and Y. Wang. Zoom out CNNs features for optical remote sensing change detection. In *Image, Vision and Computing (ICIVC), 2017 2nd International Conference on*, pages 812–817. IEEE, 2017. 2

[12] D. Ferstl, C. Reinbacher, R. Ranftl, M. Rüther, and H. Bischof. Image guided depth upsampling using anisotropic total generalized variation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 993–1000, 2013. 3

[13] B. Frénay, A. Kabán, et al. A comprehensive introduction to label noise. In *European Symposium on Artificial Neural Networks*, 2014. 2, 3

[14] B. Frénay and M. Verleysen. Classification in the presence of label noise: a survey. *IEEE transactions on neural networks and learning systems*, 25(5):845–869, 2014. 2, 3, 4

[15] E. Guo, X. Fu, J. Zhu, M. Deng, Y. Liu, Q. Zhu, and H. Li. Learning to measure change: Fully convolutional siamese metric networks for scene change detection. *CoRR*, abs/1810.09111, 2018. 1, 2

[16] I. Guyon, N. Matic, V. Vapnik, et al. Discovering informative patterns and data cleaning. In *Association for the Advancement of Artificial Intelligence*, 1996. 3

[17] K. He, J. Sun, and X. Tang. Guided image filtering. *IEEE transactions on pattern analysis and machine intelligence*, 35(6):1397–1409, 2013. 3

[18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[19] M. Hussain, D. Chen, A. Cheng, H. Wei, and D. Stanley. Change detection from remotely sensed images: From pixel-based to object-based approaches. *ISPRS Journal of Photogrammetry and Remote Sensing*, 80:91–106, 2013. 1, 2

[20] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 2

[21] P. Jeatrakul, K. W. Wong, and C. C. Fung. Data cleaning for classification using misclassification analysis. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 14(3):297–302, 2010. 3

[22] G. H. John. Robust decision trees: Removing outliers from databases. In *KDD*, pages 174–179, 1995. 3

[23] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 876–885, 2017. 3, 4, 7

[24] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[25] J. J. Koenderink. The structure of images. *Biological cybernetics*, 50(5):363–370, 1984. 5

[26] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele. Joint bilateral upsampling. In *ACM Transactions on Graphics*, volume 26, page 96. ACM, 2007. 3

[27] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011. 3, 4, 7, 8

[28] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436, 2015. 1

[29] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 1

[30] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1, 2

[31] Z. Lu, Z. Fu, T. Xiang, P. Han, L. Wang, and X. Gao. Learning from weak and noisy labels for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(3):486–500, 2017. 3

[32] N. Matic, I. Guyon, L. Bottou, J. Denker, and V. Vapnik. Computer aided cleaning of large databases for character recognition. In *International Conference on Pattern Recognition*, pages 330–333. IEEE, 1992. 3

[33] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari. Learning with noisy labels. In *Advances in neural information processing systems*, pages 1196–1204, 2013. 2

[34] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017. 7

[35] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on pattern analysis and machine intelligence*, 12(7):629–639, 1990. 3, 5

[36] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama. Digital photography with flash and no-flash image pairs. *ACM Transactions on Graphics*, 23(3):664–672, 2004. 3

[37] D. Rolnick, A. Veit, S. J. Belongie, and N. Shavit. Deep learning is robust to massive label noise. *CoRR*, abs/1705.10694, 2017. 2

[38] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2

[39] K. Sakurada and T. Okatani. Change detection from a street image pair using cnn features and superpixel segmentation. In *British Machine Vision Conference*, pages 61–1, 2015. 2

[40] A. Singh. Review article digital change detection techniques using remotely-sensed data. *International journal of remote sensing*, 10(6):989–1003, 1989. 1, 2

[41] T. Sørensen. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. *Biol. Skr.*, 5:1–34, 1948. 7

[42] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang. Learning from massive noisy labeled data for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2691–2699, 2015. 2

[43] Y. Zhan, K. Fu, M. Yan, X. Sun, H. Wang, and X. Qiu. Change detection based on deep siamese convolutional network for optical aerial images. *IEEE Geoscience and Remote Sensing Letters*, 14(10):1845–1849, 2017. 2