EVALUATION OF SEMI-SUPERVISED LEARNING FOR CNN-BASED CHANGE DETECTION

E. Bousias Alexakis *, C. Armenakis

Geomatics Engineering, GeoICT Lab, Department of Earth and Space Science and Engineering, Lassonde School of Engineering, York University, Toronto, Canada –(bousiasa,armenc)@yorku.ca

Commission III, WG III/7

KEY WORDS: Change Detection, CNN, Encoder-Decoder, Semi-Supervised Learning, UNet, Mean Teacher, Self-Ensembling

ABSTRACT:

Over the past few years, many research works have utilized Convolutional Neural Networks (CNN) in the development of fully automated change detection pipelines from high resolution satellite imagery. Even though CNN architectures can achieve state-of-the-art results in a wide variety of vision tasks, including change detection applications, they require extensive amounts of labelled training examples in order to be able to generalize to new data through supervised learning. In this work we experiment with the implementation of a semi-supervised training approach in an attempt to improve the image semantic segmentation performance of models trained using a small number of labelled image pairs by leveraging information from additional unlabelled image samples. The approach is based on the Mean Teacher method, a semi-supervised approach, successfully applied for image classification and for sematic segmentation of medical images. Mean Teacher uses an exponential moving average of the model weights from previous epochs to check the consistency of the model's predictions under various perturbations. Our goal is to examine whether its application in a change detection setting can result in analogous performance improvements. The preliminary results of the proposed method appear to be compatible to the results of the traditional fully supervised training. Research is continuing towards fine-tuning of the method and reaching solid conclusions with respect to the potential benefits of the semi-supervised learning approaches in image change detection applications.

1. INTRODUCTION

For the past few decades, the development of automatic change detection applications has been an active research area in remote sensing. A reliable change detection pipeline can be a very useful tool in many Earth Observation related applications including environmental monitoring, urban planning, map updating and disaster management.

Since the notable success of AlexNet (Krizhevsky et al., 2012) in the Large Scale Visual Recognition Challenge of 2012, Convolutional Neural Networks (CNN) have become a very popular Artificial Intelligence (AI) approach for computer vision-related tasks such as image classification, object detection and semantic segmentation. CNN models are specialized to work with data that have grid-like topology, are easier to train and can generalize better than traditional fully connected neural networks. Thanks to their stacks of convolutional and pooling layers CNN can learn useful context information from images by taking advantage of the hierarchical structure of an image's features.

Recently, CNN approaches based on encoder-decoder architectures have been successfully applied to the change detection (CD) task (Peng et al., 2019; Zhang et al., 2019a; Bousias Alexakis & Armenakis, 2020). These models perform image semantic segmentation in an end-to-end manner producing state-of-the-art results. The networks take as input a pair of co-registered image instances collected at different time periods and produce a prediction mask classifying each pixel location as changed or unchanged.

Even though the existing CNN-based architectures have been successfully applied in multiple research works, there are still open issues regarding their training and application that need to be resolved. CNN models, as all learn by example approaches, are only as good as the data that they are trained on. Therefore, in order to get high quality results there is a need for training data of similar quality. In addition, modern CNN architectures need a very high volume of data in order to generalize effectively and not overfit on the training data. The change detection training data are usually obtained from time-consuming and labour-intensive processes such as human interpretation of remotely sensed datasets with the help of semi-automated CD pipelines. Thus, there is a need for methods that can help decrease the amount of labelled data needed to successfully train a CNN-based encoder-decoder architecture and simultaneously use effectively the very large amount of available unlabelled data.

Recognizing this need, in this work we aim to improve the segmentation accuracy of encoder-decoder models in the absence of a sufficient number of labelled training samples by applying a semi-supervised training approach based on the concepts of consistency regularization and self-ensembling. In the case of specific remote sensing applications from satellite imagery, like land-cover and land-use classification or change detection applications, there is a very large amount of unlabelled training data available from sources such as Google Earth or Sentinel 2 imagery. The most challenging step for the successful training of a CNN-based CD application is to create reliable annotations for a sufficiently large training dataset so that the algorithm may learn to generalize well to new images. The proposed semi-supervised approach utilizes all the additional unlabelled information by encouraging the predictions of images subjected to various transformations to remain consistent expecting that it will lead to CD models that generalize well even when trained with a limited number of labelled examples.

* Corresponding author

2. RELATED WORK

There are numerous recent research works that address the change detection problem by training a CNN based algorithm that performs end-to-end semantic segmentation of co-registered image pairs. Most approaches use architectures inspired by Fully Convolutional Networks (Long et al., 2015), especially variations or extensions of the UNet architecture (Ronneberger et al., 2015) such as the works of (Daudt et al., 2019, 2018; Peng et al., 2019; Papadomanolaki et al., 2020; C. Zhang et al., 2019a; Bousias Alexakis & Armenakis, 2020). Even though these approaches achieve state-of-the-art results, they have always been trained and tested on small datasets due to the lack of more labelled data. One way to avoid overfitting to small datasets is to apply transfer learning techniques (Yosinski et al., 2014) as did Cao et al. (2019) when experimenting with multiple common CNN architectures for land use classification and land use change analysis.

In order to address the lack of training data, many other unsupervised or semi-supervised approaches based on Neural Networks (NN) or CNN have been recently proposed. Some of them make use of autoencoders to automatically extract features from the image pairs and then apply complex algorithms like the Chan-Vese algorithm (Zhang et al., 2019b) or a stacked mapping network and a clustering algorithm like fuzzy c-means (FCM) (Su et al., 2017). In the latter the unsupervised method is mainly based on models which learn feature representations from images. A stacked denoising autoencoder is applied to two images for feature extraction. Then mapping functions are generated by a stacked mapping network to form relationships between the features of each class. The change detection is performed by comparing the features and at the end applying a clustering algorithm. More unsupervised approaches for CD are cited by Khelifi & Mignotte (2020), who provide a comprehensive review and meta-analysis of deep learning change detection methods for remote sensing images, but in most cases the proposed methods do not make end-to-end predictions and only incorporate the Deep Neural Network as a feature extractor in the CD pipeline.

In our work we make use of the concepts of consistency constraint and self-ensembling in Deep Neural Networks (Laine & Aila, 2016; Tarvainen & Valpola, 2017). It should be mentioned that we have not been able to find a similar approach for change detection in the literature, so we will devote the rest of the literature review on works that have introduced or applied these principles for image classification and medical image semantic segmentation.

Laine & Aila (2016) introduced self-ensembling, which predicts the unknown sample labels by averaging the outputs of multiple instances of the same network on different training epochs and by also applying multiple regularizations and augmentations to the initial inputs of the models. Two different implementations of self-ensembling are proposed: a) Π-model, whose aim is to produce consistent predictions for both labelled and unlabelled data among models that undergo stochastic (thus different) dropout given the same input subjected to Gaussian noise and other augmentations; and b) temporal ensembling, which extends the Π-model by incorporating the model predictions over multiple previous training epochs. The approach was applied to an image classification task and achieved state-of-the-art results reducing by far the classification error rate of the corresponding supervised approach and also proved to be robust in the presence of incorrect labels.

Tarvainen & Valpola (2017) built on the work of Laine & Aila (2016) and proposed Mean Teacher, a method that computes the Exponential Moving Average (EMA) of model weights instead of averaging over the model's predictions. This way the EMA (also known as teacher model or Mean Teacher) is updated after each iteration and not after each epoch, which significantly increases the pace at which the training information is incorporated into the training process. The results indicate a significant training accuracy improvement and enable the models to learn using a smaller number of labelled samples compared to the approach proposed by Laine & Aila (2016).

Li et al. (2021) propose a semi-supervised semantic segmentation approach for medical imagery that incorporates both a supervised and an unsupervised component in the loss function used for training the network. For the unlabelled samples of the dataset, the algorithm learns to make consistent predictions by utilizing a regularization term that tries to minimize the difference between predictions of the same input that has been subjected to different perturbations (gaussian noise, dropout, geometric augmentations). The model also makes use of a mean teacher and student scheme (Tarvainen & Valpola, 2017) when computing the consistency regularization term, where the weights of the teacher are an exponential moving average of the student's weights on different training epochs. The proposed approach was validated in three different medical image segmentation tasks and achieved state-of-the-art results.

A very recent application of the mean teacher training scheme in a remote sensing setting was reported by (Hobley et al., 2021), where the training scheme is used to train a Fully Convolutional Network for seagrass monitoring from Remotely Piloted Aircraft (RPA) Very High Resolution (VHR) imagery. The method was compared to a fully supervised training setup as well as to an Object-based Image Analysis (OBIA) approach, resulting in improved results compared to the fully supervised setting but still not as good as the results achieved using OBIA.

3. METHODOLOGY

The method we apply to train our model follows the approach proposed by (Li et al., 2021). As mentioned earlier, the approach aims to leverage the abundance of unlabelled multi-temporal satellite data to address the lack of available labelled training data and improve the semantic segmentation performance of encoder decoder architectures for change detection applications.

Before describing the approach step by step we should once again point out that it is based on two simple ideas:

- The first one is the consistency assumption, which suggests that the model's outputs should be consistent even if the input images have been subjected to a certain number of transformations. This is also called transformation equivariance and it can be encouraged by incorporating a consistency regularization term (i.e. a term that encourages the predictions of the same input even when subjected to various perturbations to remain consistent) into the loss function.

- The second is the mean teacher training framework, which is a form of self-ensembling. Instead of only using our training model to compute the consistency regularization term we compare the results of our training model (student) to the results of a mean model (teacher), whose weights are computed based on an exponential moving average of the weights of the student model throughout the training epochs.

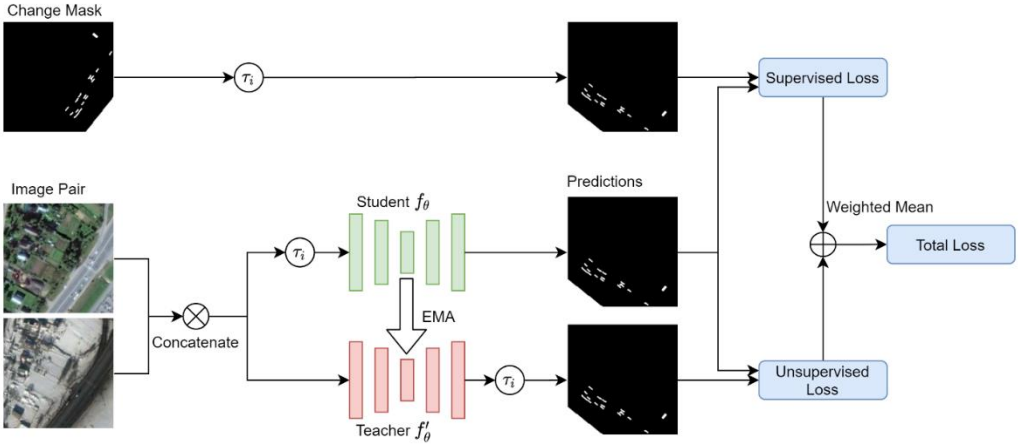

Figure 1: Overview of the proposed training approach (based on Li et al. (2021)).

Even though in the actual experiments we used a minibatch size of 8 image pairs, for the sake of simplicity, we are going to describe the training process considering a single image pair (Figure 1). For each sample image pair, the two RGB images are concatenated into a new six channel image, x_i. A set of augmentation transformations is then applied to the image pair that is used as input to the student model f_θ that produces as output a change prediction map p_i. This prediction map is then compared to the ground truth change mask, y_i, in order to compute the supervised component of the loss function, L_1. For the supervised loss component, we are using a Binary Cross Entropy (BCE) loss function (Equation 1).

The original image pair is then fed to the teacher network, f'_θ, to produce a second prediction $f'_\theta(x_i)$ that is then subjected to the same set of transformations, τ_i, so that the transformed prediction, p'_i, can be comparable to the one produced by the student model. Using predictions p_i and p'_i we can now compute the consistency regularization term of the loss function, R_1, that measures how consistent the model's predictions are when the images are subjected to random augmentations. For this unsupervised component of the loss function we have used a Mean Squared Error (MSE) loss function (Equation 2).

The total loss function is a weighted average of the supervised and unsupervised components (Equation 3), where $\lambda(t)$ is an exponential weighting function that increases the weight of the consistency regularization term as the network's training progresses and the its predictions become more accurate (Equation 4). The terms t_{max} and κ are hyperparameters of the function: t_{max} is used to set the number of epochs (threshold) after which the ramp-up function takes its maximum value, and κ is a scaling factor indicating the general weighting factor of the regularization term on the aggregated loss function.

The student's weights are updated through backpropagation while the teacher's weights are updated by an EMA (Equation 5).

$$L_1 = -(y_i \log(p_i) + (1 - y_i) \log(1 - p_i)) \quad (1)$$

$$R_1 = \|p_i - p'_i\|_2 \quad (2)$$

$$Loss = L_1 + \lambda(t)(R_1 + R_2) \quad (3)$$

$$\lambda(t) = \begin{cases} \kappa \exp\left(-5\left(1 - \frac{t}{t_{max}}\right)^2\right), & if\ t < t_{max} \\ \kappa, & if\ t > t_{max} \end{cases} \quad (4)$$

$$f'_{\theta,t} = a\, f'_{\theta,t-1} + (a-1) f'_{\theta,t} \quad (5)$$

As it was mentioned earlier, the training process was described for a single image pair, while for the actual training we used a batch size of 8 image pairs. In the process so far, we have only considered the labelled examples. For the unlabelled part of the dataset we can compute solely the consistency regularization term, but since in practical applications the number of unlabelled samples greatly exceeds the number of labelled ones, we expect that the additional unlabelled information will improve the networks performance on the semantic segmentation task and will lead to more robust predictions. Thus, the complete formula of the loss function will include one additional term, R_2, of the same form as R_1, referring to the consistency regularization term for the unsupervised image pairs of each minibatch. The complete training process is summarized in Algorithm 1.

Data: X_l : *labeled image pairs*, Y : *labels*,
X_u : *unlabeledimagepairs*
Input: $Pairs(x_{i,1}, x_{i,2}) \in X_l \rightarrow y_i \in Y, \quad (x'_{i,1}, x'_{i,2}) \in X_u$
concatenate $x_{i,1}, x_{i,2}$ into $x_i \forall (x_{i,1}, x_{i,2})$;
concatenate $x'_{i,1}, x'_{i,2}$ into $x_i \forall (x'_{i,1}, x'_{i,2})$;
initialize student model, f_θ ;
initialize teacher model, $f'_\theta = f_\theta$;
create random minibatches M
with m labeled and m' unlabeled samples ;
for *t in number of epochs* **do**
  for *each minibatch M* **do**
    create random augmentations τ_i ;
    $p_{i \in m} \leftarrow f_\theta(\tau_i(x_m))$;
    $p'_{i \in m} \leftarrow \tau_i(f'_\theta(x_m))$;
    $L_1 = \frac{1}{|m|} \sum_{i \in m} -(y_i \log(p_i) + (1 - y_i) \log(1 - p_i))$;
    $R_1 = \frac{1}{|m|} \sum_{i \in m} \|p_i - p'_i\|_2$;
    $R_2 = \frac{1}{|m'|} \sum_{i \in m'} \|p_i - p'_i\|_2$;
    $Loss = L_1 + \lambda(t)(R_1 + R_2)$;
    update f_θ weights using Adam optimizer ;
    update f'_θ weights using EMA ;
  end
  create new random minibatches ;
end

Algorithm 1: Semi-supervised training process based on the approach of Li et al. (2021).

For our student and teacher models we are using the UNet architecture (Ronneberger et al., 2015). UNet (Fig. 2) consists of a contracting and a symmetrical expanding path and takes advantage of both the contextually rich semantic information of the coarser lower layers and the spatially accurate activations of the fine-grained higher layers by introducing multiple skip connections between contrastive and expanding blocks that share

the same resolution. We followed the original UNet architecture (Ronneberger et al., 2015) and also added a batch normalization layer after each convolutional layer as it has been shown to help the models learn faster (Ioffe and Szegedy, 2015).

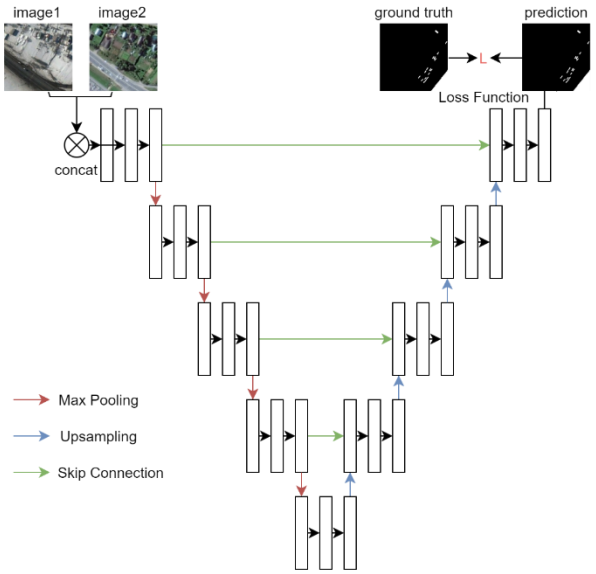

Figure 2: UNet architecture.

4. EXPERIMENTS

4.1 Dataset

Our experiments were conducted on a CD dataset proposed by Lebedev et al. (2018) that consists of 16000 RGB image pairs of size 256 × 256 pixel, taken from Google Earth (DigitalGlobe), and their binary change masks. The 16000 samples are split into a 10000 samples train set and into a 3000 samples validation and test sets. The pixel ground resolution ranges from 30cm to 1m. The masks are based solely on changes that relate to the appearance or disappearance of objects between the two instances of the pair and ignore any seasonal variations.

4.2 Training Details

We have randomly split the training set into smaller sets $S_{500} \subset S_{1000} \subset S_{2500} \subset S_{5000} \subset S_{7500}$ each containing 500, 1000, 2500, 5000, 7500 labelled image pairs. For each smaller set, we have used the rest of the image pairs as unlabelled training samples. So, for example S_{500}will be trained using 500 labelled and 9500 unlabelled samples, S_{1000} will be trained using 1000 labelled and 9000 unlabelled samples and so on and so forth. Finally, we have trained the network using all 10000 labelled training samples in a fully supervised way to use as a benchmark for the results retrieved using the smaller labelled training sets

The training was conducted on a NVIDIA Quadro RTX 5000 GPU using PyTorch (Paszke et al., 2019). For the image data augmentations, we have used the Albumentations (Buslaev et al., 2020) library. For the transformations τ_i we have used random 90-degree angle rotations, random horizontal and vertical flipping and random crop and rescale transforms. Besides from the geometric transformations we have also used a couple of radiometric augmentations: an RGB shift augmentation, where the RGB values of an image are shifted by a randomly chosen value for each channel in the interval (-20, +20), as well as a random brightness and contrast augmentation.

In order to reduce the training time, we have first trained a UNet model from scratch on the 2500 sample set without any data augmentation for 150 epochs (we noticed that at that point the network started to overfit to the training set). For the training sets containing 2500 samples or more we used the pretrained network's weights as starting weights and trained for another 46875 iterations[1]. We used a learning rate of 0.0003 for the first 2/3 of the training and of 0.0001 for the last 1/3. The training sets containing less than 2500 labelled examples (S_{500} and S_{1000}) were trained from scratch following the same principles.

4.3 Results

Figure 3 and Table 1 present the Intersection over Union (IoU) results we retrieved on the training and validation dataset for different labelled sample sizes. The only case where the semi-supervised training achieves better performance on the validation set is for the 2500 labelled sample size. In all other cases the supervised training with augmentations performs better than the proposed semi-supervised approach.

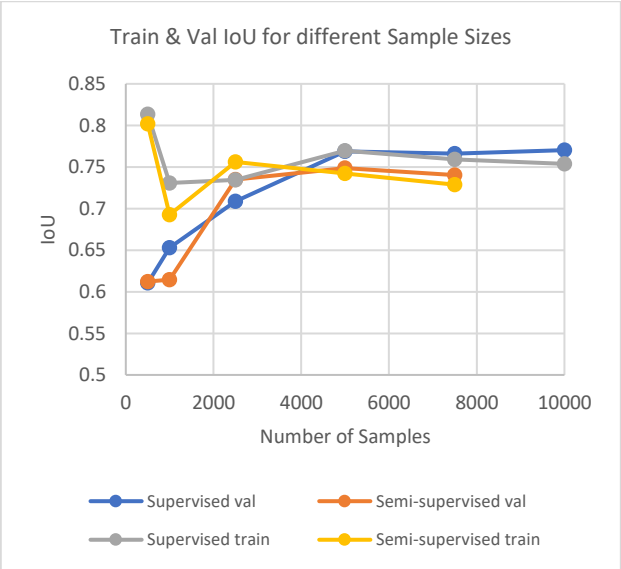

Figure 3. Training and validation IoU metrics for semi-supervised and supervised models for varying number of labelled training samples.

The results are contrary to our initial expectations. We expected that for small number of samples the semi-supervised approach would perform better than the supervised one thanks to the extra information provided by the unlabelled data and that as the number of training samples increased, the benefits of the semi-supervised training scheme would wear out with the two methods producing similar results for higher sample sizes. Instead, there is no distinct pattern connecting the relative performance of the two methods with the number of samples. On the smallest labelled sample size both approaches perform similarly (around 61%) and greatly overfit to the data. When the labelled sample size is set to 1000, the semi-supervised approach has a validation

[1] The number of iterations is not rounded because in our implementation we used the number of epochs and not the number of iterations to iterate over the datasets. Thus 150 epochs on the 2500 sample set with a minibatch size of 8 translates to 46875 iterations and to 75 epochs on the 5000-sample set with the same minibatch size and so on for the rest of the training sets.

Number of labelled samples	Dropout	Training method	Initial Weights	Train loss	Train IoU	Val loss	Val IoU
500	✗	Sup	Random	0.0483	0.8136	0.1617	0.6105
500	✗	Semi-Sup	Random	0.0649	0.8021	0.1554	0.6123
500	✓	Semi-Sup	Random	0.0552	0.8316	0.1664	0.6101
1000	✗	Sup	Random	0.0759	0.7308	0.1081	0.6529
1000	✗	Semi-Sup	Random	0.0952	0.6926	0.1178	0.6146
1000	✓	Semi-Sup	Random	0.0851	0.7415	0.1185	0.6403
2500	✗	Sup	Pretrained	0.0751	0.7347	0.0823	0.7087
2500	✗	Semi-Sup	Pretrained	0.0780	0.7561	0.0727	0.7351
5000	✗	Sup	Pretrained	0.0649	0.7695	0.0621	0.7686
5000	✗	Semi-Sup	Pretrained	0.0836	0.7421	0.0686	0.7487
5000	✓	Semi-Sup	Pretrained	0.0796	0.7426	0.0709	0.7446
7500	✗	Sup	Pretrained	0.0673	0.7592	0.0613	0.7663
7500	✗	Semi-Sup	Pretrained	0.0892	0.7290	0.0710	0.7402
10000	✗	Sup	Pretrained	0.0694	0.7539	0.0607	0.7703

Table 1. Training and validation IoU metrics for semi-supervised and supervised models for varying number of labelled training samples

IoU about 4% lower than the supervised approach and for larger training sizes the IoU of the fully supervised approach exceeds the semi-supervised IoU (by about 2% on $S_{5000}$and 2.5% on S_{7500}). The only case where the semi-supervised approach outperforms the fully supervised results is on the S_{2500} (by 2.6%).

In our initial experiments we did not use any dropout layers (Srivastava et al., 2014) in order to examine whether the geometric and radiometric augmentations would be sufficient perturbations for the semi-supervised training to succeed. Since Li et al. (2021) used dropout in their solution that outperformed the fully supervised training we also ran extra experiments applying an additional dropout regularization with a probability of 0.3 (or 30%) on the output of the last convolutional block and before applying the final convolutional layer of the model. Dropout was applied on three of the training schemes ($S_{500}, S_{1000}, S_{5000}$) and resulted in an improved IoU for S_{1000} (about 2.5%), but still lower than the respective fully supervised result, and in slightly worse IoUs (less than 0.5%) in the case of S_{500} and S_{5000}.

When considering each method individually, the results seem reasonable. For very small sample sizes there is a large gap between training and validation IoU suggesting overfitting to the small training set for both models (the gap is around 20% for both models), that is gradually closing as the number of samples increases, with the validation IoU being even higher than the training IoU for bigger labelled sample sizes (this is the case for S_{7500} for both supervised and semi-supervised methods and S_{10000} for the supervised one) indicating that more labelled training samples help the models generalize better, which is the expected behaviour. The fact that the validation IoU is higher than the training IoU may relate to a condition included in the training when saving the best model. The condition was based solely on the IoU performance on the validation set as a safety net against overfitting.

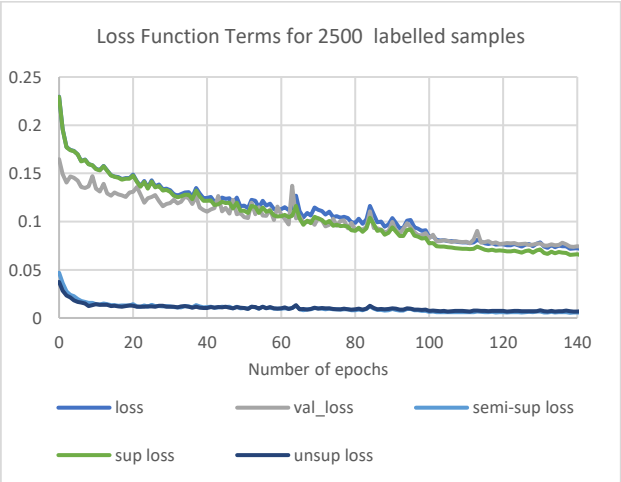

Figure 4. Loss function terms of the semi-supervised method for S_{2500}. *Loss* is the aggregate loss, *sup loss* is the supervised component of the loss function (L_1), *semi-sup loss* is the consistency regularization term for the labelled samples (R_1), and *unsup loss* is the consistency regularization term for the unlabelled samples (R_2) of the dataset.

The training and validation loss curves (average loss function values per epoch) presented in Figure 4 and 5 can help us examine the learning behaviour of the semi-supervised approach. We can see that the consistency regularization terms have a significantly lower range of values. All terms decrease over time and seem to converge by the end of training. Also, the change of the learning rate from 0.0003 to 0.0001 at epoch 100 (for S_{2500}) and 75 (for S_{2500}) is visible for both learning curves.

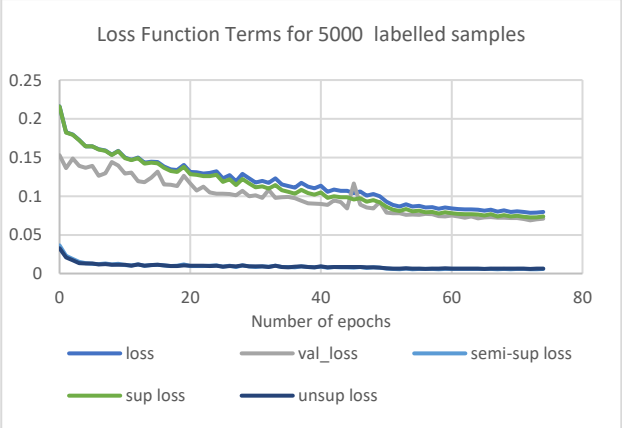

Figure 5. Loss function terms of the semi-supervised method for S_{5000}. Loss is the aggregate loss, sup loss is the supervised component of the loss function (L_1), semi-supervised loss is the consistency regularization term for the labelled samples (R_1), and unsup loss is the consistency regularization term for the unlabelled samples (R_2) of the dataset.

In Figure 6 we present predictions from different models for 8 image pairs selected randomly from the validation set. Overall, a qualitative analysis of the predictions suggests that the models trained on 2500 images (both supervised and unsupervised) seem to perform similarly to the fully supervised model trained on 10000 images, while the models trained on the 500 images do not seem to produce accurate predictions, especially when it comes to small or thin elongated objects and regions with complex shapes/boundaries.

5. CONCLUSION

In this work we implemented a Mean Teacher semi-supervised training setup following the work of Li et al. (2021) and applied it to a Change Detection setting to explore the potential benefits of the method compared to a fully supervised training process, especially when only a few labelled training examples are available. We expected that the consistency regularization constraint would allow the model to learn useful information from unlabelled data, improving the model's performance when limited labelled samples are available, which is often the case in CD applications.

The preliminary results indicate that the proposed approach does not outperform the fully supervised training setup for the particular change detection dataset. Contrary to our initial expectations, there is no clear relation between the size of the labelled training set and any performance benefits of applying the semi-supervised training scheme instead of a fully supervised solution. In general, the fully supervised approach slightly outperforms the semi-supervised approach for almost all labelled training set sizes, with the exception of S_{2500}.

Even though the preliminary results are not in favour of the proposed semi-supervised method, further experiments are required in order to extract more solid conclusions regarding the usefulness of the method for Change Detection applications. Future work will involve larger testing datasets and stronger and more varied perturbations to the input data which will hopefully lead to higher model performance and safer conclusions regarding the training of CNN models using a limited number of labelled samples and consistency regularization.

ACKNOWLEDGEMENTS

This work is financially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC Discovery and CREATE grants) and York University. The Change Detection Dataset was provided by Lebedev et al., 2018: https://drive.google.com/file/d/1GX656JqqOyBi_Ef0w65kDGVto-nHrNs9

REFERENCES

Bousias Alexakis, E., & Armenakis, C. (2020). Evaluation of UNet and UNet++ Architectures In High Resolution Image Change Detection Applications. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B3-2020, 1507–1514. https://doi.org/10.5194/isprs-archives-XLIII-B3-2020-1507-2020

Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M., & Kalinin, A. A. (2020). Albumentations: Fast and Flexible Image Augmentations. Information, 11(2). https://doi.org/10.3390/info11020125

Cao, C., Dragićević, S., & Li, S. (2019). Land-Use Change Detection with Convolutional Neural Network Methods. *Environments*, 6(2), 25. https://doi.org/10.3390/environments6020025

Daudt, C. R., Le Saux, B., & Boulch, A. (2018). Fully Convolutional Siamese Networks for Change Detection. *25th IEEE International Conference on Image Processing (ICIP)*, 4063–4067. https://doi.org/10.1109/ICIP.2018.8451652

Daudt, C. R., Le Saux, B., Boulch, A., & Gousseau, Y. (2019). Multitask learning for large-scale semantic change detection. *Computer Vision and Image Understanding*, 187, 102783. https://doi.org/10.1016/j.cviu.2019.07.003

Hobley, B., Arosio, R., French, G., Bremner, J., Dolphin, T., & Mackiewicz, M. (2021). Semi-supervised segmentation for coastal monitoring seagrass using RPA imagery. https://doi.org/10.20944/preprints202103.0780.v1

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *ArXiv:1502.03167* [Cs]. http://arxiv.org/abs/1502.03167

Khelifi, L., & Mignotte, M. (2020). Deep Learning for Change Detection in Remote Sensing Images: Comprehensive Review and Meta-Analysis. *ArXiv:2006.05612* [Cs]. http://arxiv.org/abs/2006.05612

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Proceedings of the 25th International Conference on Neural Information Processing Systems* - Volume 1, 1097–1105.

Laine, S., & Aila, T. (2016). *Temporal Ensembling for Semi-Supervised Learning.* https://openreview.net/forum?id=BJ6oOfqge

Lebedev, M. A., Vizilter, Y. V., Vygolov, O. V., Knyaz, V. A., & Rubis, A. Y. (2018). Change Detection in Remote Sensing Images Using Conditional Adversarial Networks. *ISPRS - International Archives of the Photogrammetry, Remote Sensing*

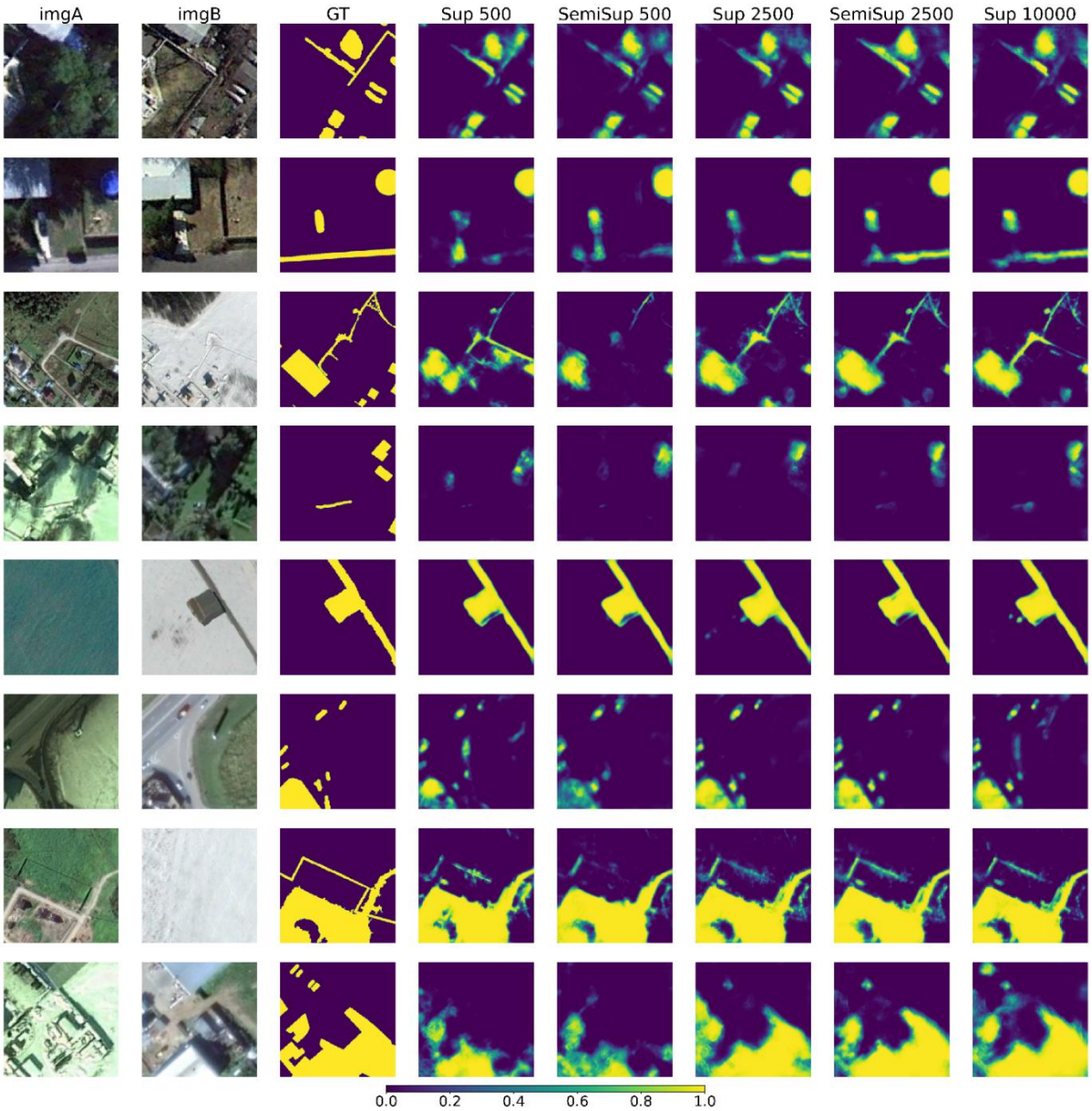

Figure 6. Prediction examples for different sample sizes for both supervised and semi-supervised training.

and Spatial Information Sciences, XLII–2, 565–571. https://doi.org/10.5194/isprs-archives-XLII-2-565-2018

Li, X., Yu, L., Chen, H., Fu, C.-W., Xing, L., & Heng, P.-A. (2021). Transformation-Consistent Self-Ensembling Model for Semisupervised Medical Image Segmentation. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2), 523–534. https://doi.org/10.1109/TNNLS.2020.2995319

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3431–3440. https://doi.org/10.1109/CVPR.2015.7298965

Papadomanolaki, M., Vakalopoulou, M., & Karantzalos, K. (2020). Urban Change Detection Based on Semantic Segmentation and Fully Convolutional Lstm Networks. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, V-2–2020, 541–547. https://doi.org/10.5194/isprs-annals-V-2-2020-541-2020

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., … Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* 32 (pp. 8024–8035). Curran Associates, Inc. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Peng, D., Zhang, Y., & Guan, H. (2019). End-to-End Change Detection for High Resolution Satellite Images Using Improved UNet++. *Remote Sensing*, 11(11), 1382. https://doi.org/10.3390/rs11111382

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.),

Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015 (pp. 234–241). Springer International Publishing. https://doi.org/10.1007/978-3-319-24574-4_28

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(56), 1929–1958.

Su, L., Gong, M., Zhang, P., Zhang, M., Liu, J., & Yang, H. (2017). Deep learning and mapping based ternary change detection for information unbalanced images. *Pattern Recognition*, 66, 213–228. https://doi.org/10.1016/j.patcog.2017.01.002

Tarvainen, A., & Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 1195–1204.

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? ArXiv:1411.1792 [Cs]. http://arxiv.org/abs/1411.1792

Zhang, C., Wei, S., Ji, S., & Lu, M. (2019a). Detecting Large-Scale Urban Land Cover Changes from Very High Resolution Remote Sensing Images Using CNN-Based Classification. *ISPRS International Journal of Geo-Information*, 8(4), 189. https://doi.org/10.3390/ijgi8040189

Zhang, X., Shi, W., Lv, Z., & Peng, F. (2019b). Land Cover Change Detection from High-Resolution Remote Sensing Imagery Using Multitemporal Deep Feature Collaborative Learning and a Semi-supervised Chan–Vese Model. *Remote Sensing*, 11(23), 2787. https://doi.org/10.3390/rs11232787