Siamese NestedUNet Networks for Change Detection of High Resolution Satellite Image

Kaiyu Li
Shandong University of Science and Technology, China
likyoo@sdust.edu.cn

Zhe Li
Shandong University of Science and Technology, China
lizhe@sdust.edu.cn

Sheng Fang
Shandong University of Science and Technology, China
fangs99@126.com

ABSTRACT

Change detection is an important task in remote sensing (RS) image analysis. With the development of deep learning and the increase of RS data, there are more and more change detection methods based on supervised learning. In this paper, we improve the semantic segmentation network UNet++ and propose a fully convolutional siamese network (Siam-NestedUNet) for change detection. We combine three types of siamese structures with UNet++ respectively to explore the impact of siamese structures on the change detection task under the condition of a backbone network with strong feature extraction capabilities. In addition, for the characteristics of multiple outputs in Siam-NestedUNet, we design a set of experiments to explore the importance level of the output at different semantic levels. According to the experimental results, our method improves greatly on a number of indicators, including precision, recall, F1-Score and overall accuracy, and has better performance than other SOTA change detection methods. Our implementation will be released at https://github.com/likyoo/Siam-NestedUNet.

CCS CONCEPTS

• **Computing methodologies** → Machine learning; • **Artificial intelligence**; • **Computer vision**; • **Computer vision tasks**; • **Applied computing**; • **Physical sciences and engineering**; • **Earth and atmospheric sciences**; • **Machine learning approaches**; • **Neural networks**; • **Human-centered computing**; • **Visualization**; • **Visualization application domains**; • **Geographic visualization**; • **Machine learning**; • **Learning paradigms**; • **Supervised learning**;

KEYWORDS

Change Detection, Deep Learning, Fully Convolutional Siamese Network, Remote Sensing Image Processing

ACM Reference Format:
Kaiyu Li, Zhe Li, and Sheng Fang. 2020. Siamese NestedUNet Networks for Change Detection of High Resolution Satellite Image. In *2020 International Conference on Control, Robotics and Intelligent System (CCRIS 2020), October 27–29, 2020, Xiamen, China.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3437802.3437810

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CCRIS 2020, October 27–29, 2020, Xiamen, China
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8805-4/20/10…$15.00
https://doi.org/10.1145/3437802.3437810

1 INTRODUCTION

The objective of change detection is to detect pixels with "semantic change" between two images, which are acquired at different times in the same area. There are many factors that may cause "change", such as the color change of the object, the relative motion of the object, etc. The key point of change detection task is that the change map should not contain non-semantic changes, or those we do not need to pay attention to, such as changes caused by camera motion, sensor noise, or light changes.

In recent years, fully convolutional neural networks [1, 2] are widely used in dense prediction tasks such as semantic segmentation. It replaces all fully connected layers with convolutional layers in convolutional neural networks (CNNs). Their output is not classification scores, but spatial maps. As a pixel-to-pixel detection task, change detection has made breakthrough progress under the use of FCNs.

Recently, U-Net [3] structures have been frequently used in image-based change detection methods [4, 5]. U-Net is first used in semantic segmentation of medical images, and achieves very representative performance. In medical images, the semantics are relatively simple, and the structures are quite fixed, so U-Net can still achieve good results with few parameters. However, remote sensing images are complex and have multiple collection sources, in other words, their semantic information is abundant and diverse. Therefore, we need to consider whether the feature extraction capability of U-Net is enough for the change detection. task. We believe that the improvement brought by the use of U-Net++ [6, 7] in [8] is sufficient to answer this question to a certain extent.

Definitely, change detection task and semantic segmentation are similar to some extent, they are both pixel-to-pixel prediction tasks and get a category map finally. However, this does not mean it is a reasonable method to concatenate the bi-temporal images and input them directly into FCNs. In other words, it is illogical to completely consider a change detection problem as a semantic segmentation problem. The siamese network [9] has better interpretability in change detection task [4, 10-13]. It shares the same weights in two branches, which means that it uses the same method to extract features of two images. The image pairs we input are generally obtained with the same device and in the same area, so there is a certain similarity between the two images. Therefore, it is reasonable to extract features using the same method. In addition, the weight sharing makes the network have fewer parameters and converge faster.

In this paper, we propose an end-to-end fully convolutional siamese network for change detection of VHR satellite images, which is called Siam-NestedUNet. We try to combine the siamese

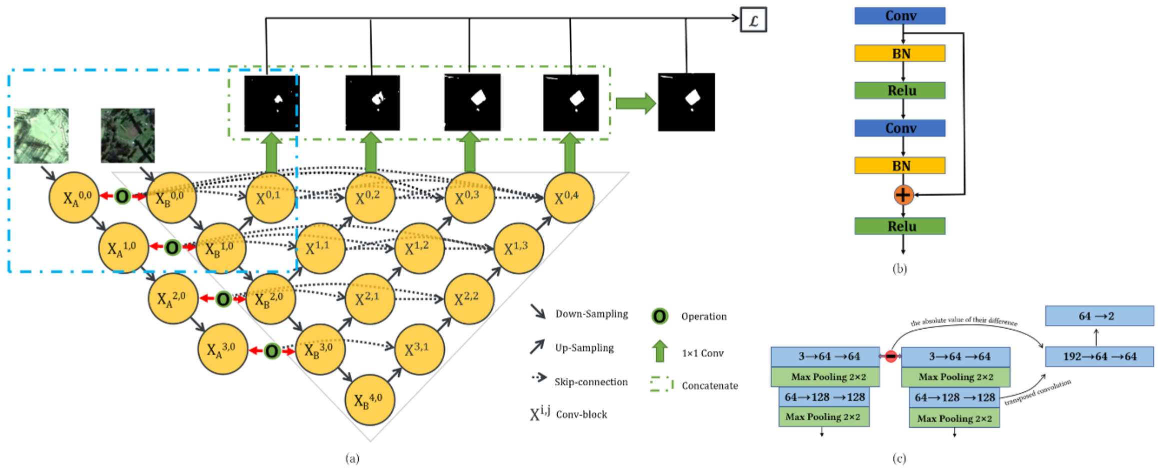

Figure 1: Illustration of the proposed architecture, named Siam-NestedUNet. (a) is the flowchart of Siam-NestedUNet, downward arrows and upward arrows indicate down-sampling and up-sampling respectively, the dotted arrows indicate skip connections; the node $X^{i,j}$ indicates a convolution block, the detailed structure is shown in (b); the green circle indicates the operation between the two branches; the green arrow indicates a 1×1 convolutional layer; the green dotted frame indicates concatenation. (c) is part of the details in the flowchart, indicated in (a) by a blue dotted frame, the red circle indicates operations of the Siam-diff structure.

structures in different ways on the basis of sufficient feature extraction capability of UNet++, and explore the impact of different siamese structures on its performance. Different from the work of [8], which directly concatenates two images and inputs them into UNet++, we introduce siamese structures and make improvements in other aspects, such as convolution block, network complexity and so on. In addition, for the multi-sided output of Siam-NestedUNet, we try to explore the importance level of the output on different sides to pay more attention to the sides containing crucial information.

This paper is organized as follows. In Section 2, we discuss some change detection methods and related works based on deep learning, which are instructive to our work. Section 3 describes the change detection method proposed in this paper. Section 4 contains a series of quantitative comparisons and analyses through experiments. Finally, the conclusion of this paper is drawn in Section 5.

2 RELATED WORK

Change detection has been widely used in many fields, such as vegetation change, urban expansion, and illegal building detection. Traditional change detection methods like image difference method ignore the context information of pixels, which may cause a lot of salt and pepper noise.

In many fields such as object detection and semantic segmentation, compared with traditional methods, deep learning methods have great advantages in feature representation. With the increase of remote sensing data, many supervised learning methods are gradually applied to the change detection of RS image.

In [8], these methods are divided into three categories: (1) feature-based methods [12-15], (2) patch-based methods [10, 11], and (3) image-based methods [4, 5, 8]. To some extent, the image-based method regards the change detection task as a semantic segmentation task, and directly converts the input to change map through the FCNs. This type of method achieves end-to-end change detection and avoids the accumulation of errors. In addition, it has a very large advantage in detection speed, which is beneficial for processing large amounts of data.

Three different fully convolutional architectures are proposed based on the U-Net architecture in [4]: FC-EF, FC-Siam-conc and FC-Siam-diff. Among them, the excellent performance of FC-Siam-diff structure is very enlightening for the follow-up work. To some extent, this architecture explicitly guides the network to compare two images. In [5], U-Net and LSTM are combined to extract spatial and temporal information simultaneously. However, this method requires additional data, which is often difficult to achieve. In [8], the Early-Fusion strategy [10] is used to fuse the bi-temporal images, and U-Net++ is used as the backbone network. The powerful feature extraction and fusion capabilities of UNet++ greatly improve the performance of this method.

In contrast to the above methods, we add different siamese structures to the network with strong feature extraction capabilities to enhance performance in change detection task.

3 METHODOLOGY

3.1 Network Architecture

The Siam-NestedUNet structure proposed in this paper uses UNet++ as the backbone network, and a siamese branch is added to the encoder to emphasize the particularity of change detection task compared to other segmentation tasks. It is not the first time that UNet++ is used as a backbone network in change detection. In [8], VHR images are input into UNet++ through an Early-Fusion strategy (EF-UNet++). Benefiting from the excellent performance

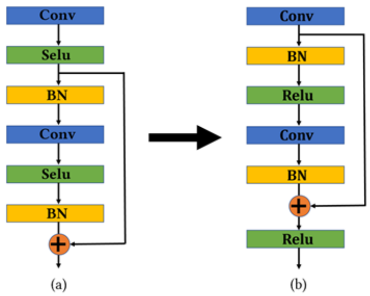

Figure 2: The difference between convolutional blocks.

of UNet++ in feature extraction and feature fusion, it achieves good results. However, we observe that the Early-Fusion strategy is not helpful for the network to clear its task and find the difference between two images.

In [4], two siamese structures are proposed for change detection: FC-Siam-conc and FC-Siam-diff, corresponding to concatenation and subtraction between two siamese branches in the encoder. We combine these two structures to produce a more complex structure: Siam-conc-diff. As the name implies, it is a violent combination of the previous two structures, which aims to use the difference to guide the network while combining low-resolution information and high-resolution information. Next, these three siamese structures (Siam-conc, Siam-diff, and Siam-conc-diff) are combined with Siam-NestedUNet structure to explore the impact of different siamese structures on the network with sufficient feature extraction capability.

The flowchart of the proposed method is shown in Figure 1(a). In Figure 1(a), each node $X^{i,j}$ denotes a convolution block as shown in Figure 1(b). Compared to the convolution block in [8], we make some improvements to make it more reasonable and efficient according to [16]. As shown in Figure 2, (a) is the convolution block in EF-UNet++, and (b) is that we use (the experimental results are in Section 4.3.1).

Let $x^{i,j}$ denotes the output of node $X^{i,j}$, $x^{i,j}$ is formulated as follows:

$$x^{i,j} = \begin{cases} \mathcal{P}(\mathcal{H}(x^{i-1,j})) & j = 0 \\ \mathcal{H}([x_A^{i,0} \odot x_B^{i,0}, \quad \mathcal{U}(x^{i+1,j-1})]) & j = 1 \\ \mathcal{H}([x_A^{i,0} \odot x_B^{i,0}, [x^{i,k}]_{k=1}^{j-1}, \quad \mathcal{U}(x^{i+1,j-1})]) & j > 1 \end{cases} \quad (1)$$

The function H(·) denotes the operation of the convolution block, the use of improved residual structure makes our network easier to converge. The function P(·) denotes a 2x2 max pooling operation using for down-sampling, which is indicated by downward arrows in Figure 1(a). The function U(·) denotes up-sampling using transpose convolution, which can restore the scale while further extracting relevant information. ⊙ contains three types of operations (concatenation, subtraction and the combination of them) between two siamese branches. [ ] denotes the concatenation on the channel dimension and aims to fuse information of different semantic levels.

The backbone network of Siam-NestedUNet has multiple outputs, for the output $\{x^{0,j}, j \in \{1,2,3,4\}\}$ of $\{X^{0,j}, j \in \{1,2,3,4\}\}$, a 1×1 convolutional layer is added to generate a 2×H×W feature map $\hat{Y}$ ("2" indicates change and no change) . $\hat{Y}$ can be formulated as follows:

$$\hat{Y}^j = h(x^{0,j}), j \in \{1,2,3,4\} \quad (2)$$

$$\hat{Y}^5 = h(\left[Y^1, Y^2, Y^3, Y^4\right]) \quad (3)$$

where h(·) denotes a 1×1 convolution operation, and $\hat{Y}^j$ denotes the output of j-th semantic level. From the perspective of ensemble learning, $\hat{Y}^5$ is a further decision of the previous four classifiers of different semantic levels, and the improvement of its performance is a normal thing.

3.2 Details of Loss Function

Similar to [6-8], we use a deep supervision strategy in our method. For each output of five semantic levels, the hybrid loss function (weighted cross-entropy loss and dice loss) is defined as:

$$\mathcal{L}_{side}^j = \mathcal{L}_{wce}^j + \mathcal{L}_{dice}^j \quad (4)$$

The change map $\hat{Y}$ of each level is regarded as a set of points, which can be represented as $\hat{Y} = \{\hat{y}_k, k = 1,2,\ldots,H \times W\}$. $\hat{y}_k$denotes a point in $\hat{Y}$, and it contains two values here. H and W denote the height and width of $\hat{Y}$, which is the same size as the original images. The weighted cross-entropy (WCE) loss can be formulated as:

$$\mathcal{L}_{wce} = \frac{1}{H \times W} \sum_{k=1}^{H \times W} \text{weig}ht[class] \cdot (\log(\frac{\exp(\hat{y}[k][class])}{\sum_{l=0}^{1} \exp(\hat{y}[k][l])})) \quad (5)$$

where the value of "class" is 0 or 1, corresponding to the unchanged and changed pixels, respectively. At the same time, the prediction result $\hat{Y}$ participates in calculating the dice loss after a softmax layer:

$$\mathcal{L}_{dice} = 1 - \frac{2 \cdot Y \cdot soft\max(\hat{Y})}{Y + soft\max(\hat{Y})} \quad (6)$$

where Y indicates the ground truth. In the data of change detection, the number of unchanged pixels always accounts for the vast majority, so there is a problem of sample imbalance. The parameter $weight[class]$ and the dice loss are used to balance the contribution of positive and negative sample to the hybrid loss. Finally, the overall loss function is defined as:

$$\mathcal{L} = \sum_{j=1}^{5} w_j \mathcal{L}_{side}^j \quad (7)$$

where w_j corresponds to the weights of the five semantic levels in Siam-NestedUNet. In the past use of U-Net++ structure, w_j is generally set to 1.0. However, intuitively, we believe that the output of the deeper semantic level should be paid more attention, and set a larger weight to make the network converge faster. This is because the deeper nodes always contain the information of the shallower nodes through skip pathways (it will be discussed in Section 4.3.4).

4 EXPERIMENTS

4.1 Dataset and Evaluation Metrics

To evaluate our method, we design a series of comparative experiments on CDD (Change Detection Dataset) [17] dataset. CDD dataset contains 11 pairs of RS images taken in different seasons

Table 1: Influence of the convolution block in EF-UNet++

Method	Precision	Recall	F1-Score	OA
EF-UNet++ [8]	0.890	0.846	0.866	0.973
EF-UNet++ *	0.911	0.883	0.896	0.978

obtained by Google Earth (Digital Globe), and the spatial resolution of these images is 3cm/px to 100cm/px. In [17], 16000 pairs of images with a size of 256 × 256 pixels are generated from the original image through cropping and rotation operations. 10,000 pairs are divided into the training set, 3000 pairs are divided into the verification set, and 3000 pairs are divided into the testing set.

For quantitative metrics evaluation, we use four indicators: Precision (P), Recall (R), F1-Score (F1) and Overall Accuracy (OA). They are expressed as follows:

$$P = \frac{TP}{TP + FP} \tag{8}$$

$$R = \frac{TP}{TP + FN} \tag{9}$$

$$F1 = \frac{2 \cdot P \cdot R}{P + R} \tag{10}$$

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

where TP, TN, FP, and FN respectively indicate the number of true positives, true negatives, false positives, and false negatives.

4.2 Implementation details

We implement Siam-NestedUNet using the Pytorch framework. In the training procedure, AdamW [18] is applied as optimizer. Learning rate is set to 5e-4 and decays by 0.5 every 8 epochs. The weights of each convolutional layer are initialized by the KaiMing normalization [19]. Two tensors with 256×256×3 pixels are input to siamese branches, and we get the output with a size of 256×256×2 pixels.

4.3 Comparison and Analysis

The comparison methods selected in this paper are FC-EF [4], FC-Siam-conc [4], FC-Siam-diff [4] and EF-UNet++ [8]. Compared with EF-UNet++, in addition to adding the siamese structure, we also make some other improvements to improve the performance of Siam-NestedUNet.

4.3.1 Convolution Block. The convolution block in [8] uses batch normalization layer after activation function as shown in Figure 2(a), which may lead to the input value cannot be distributed in the sensitive area of the nonlinear function. We can see the improvement brought by changing the convolution block in Table 1. The second row is the result of only changing the structure of the convolution block when other settings are same.

4.3.2 Siamese Structures. According to Table 2, the results of FC-EF, FC-Siam-conc and FC-Siam-diff illustrate the limitation of the feature extraction capability of U-Net in change detection. Meanwhile, the siamese networks improve the performance comprehensively compared with Early-Fusion structure. Siam-conc and Siam-diff indicate the use of concatenation and subtraction in the encoding stage of Siam-NestedUNet, respectively. Siam-conc-diff means the combination of the previous two structures. As listed in Table 2, Siam-diff achieves the best results in precision, and Siam-conc achieves the best performance in recall, F1-Score, and overall accuracy, corresponding to increases of 2.5%, 1.9% and 0.3% compared to EF-UNet++*, respectively.

When extracting features from the image pair, siamese networks activate the same location in the feature maps of two images by sharing weights. This means that the feature maps of two branches can be kept corresponding in the location, which is conducive to the comparison of the location information of the two images. The location information of objects is particularly important to change detection, which is why siamese networks can bring improvement.

4.3.3 Wider Networks. As we all know, the wider network can capture more fine-grained features and is easier to train. Therefore, for the complexity and abundance of RS images, we increase the number of convolution filters from {32, 64, 128, 256, 512} to {64, 128, 256, 512, 1024} in encoder to enhance the feature extraction capability of Siam-NestedUNet and extract more fine-grained features. The variation in the number of channels can be referred to in Figure 1(c).

In Table 3, the increase in the amount of model parameters brought an improvement beyond our expectations, and almost no overfitting occurs according to the training curve.
Compared with Siam-conc and Siam-conc-diff, the Siam-NestedUNet model with Siam-diff structure obtains the best F1-Score, and all its indicators are not less than 0.95. Compared to Siam-conc, the subtraction operation between the siamese branches explicitly guides Siam-NestedUNet to compare the differences between two images, as explained in [4]. For Siam-conc-diff, too much encoder information may lead to a slow and entangled learning process, which makes Siam-conc-diff perform worse than Siam-diff.

4.3.4 Attention to Deeper Features. As mentioned in the previous section, we try to explore the importance level of output at different semantic levels. Therefore, we prune the original network into Siam-NestedUNet R^1-R^4 as shown in Figure 3 to observe the impact of removing the output of one side.

As shown in Table 4, we observe that as the output of the deeper semantic level is removed, the influence on the final result has an increasing tendency. In other words, the deeper output is more important than the shallower. Therefore, we try to adjust the contributions of different semantic levels to the final result by adjusting the parameter w in formula (7). In this paper, we just adjust w roughly, changing {1, 1, 1, 1, 1} to {0.5, 0.5, 0.75, 0.75, 1.0}, aiming to make the network pay more attention to features of deeper levels. This improvement is listed in Siam-diff-w of Table 3, with F1-Score

Table 2: Performance comparison on CDD dataset

Method / Channel	Precision	Recall	F1-Score	OA
FC-EF [4]	0.609	0.583	0.592	0.911
FC-Siam-conc [4]	0.660	0.607	0.628	0.919
FC-Siam-diff [4]	0.709	0.603	0.637	0.933
EF-UNet++ * / 32	0.911	0.883	0.896	0.978
Siam-conc / 32	0.924	0.908	0.915	0.981
Siam-diff / 32	0.929	0.885	0.906	0.980
Siam-conc-diff / 32	0.921	0.895	0.907	0.980

Table 3: performance of wider networks

Method / Channel	Precision	Recall	F1-Score	OA
Siam-conc / 64	0.951	0.947	0.949	0.987
Siam-diff / 64	0.958	0.950	0.953	0.987
Siam-conc-diff / 64	0.948	0.952	0.950	0.987
Siam-diff-w / 64	0.959	0.955	0.956	0.988

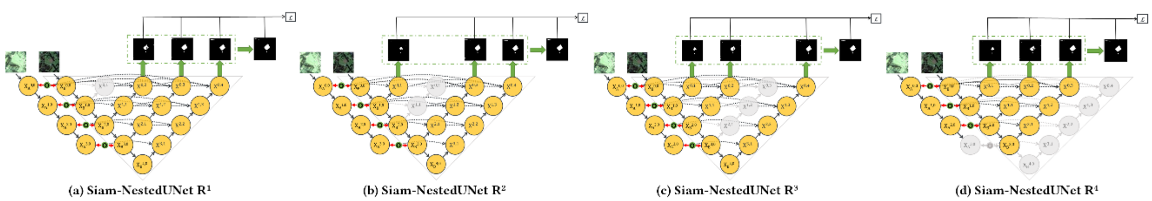

Figure 3: Remove some nodes to explore the importance level of different semantic levels. R^i means to remove the output of the i-th semantic level. The removed nodes are colored in gray.

Table 4: Influence of different semantic levels [a]

Method / Channel	Precision	Recall	F1-Score	OA
Siam-diff R^1 / 64	0.945	0.938	**0.941**	0.985
Siam-diff R^2 / 64	0.944	0.932	**0.938**	0.985
Siam-diff R^3 / 64	0.938	0.914	**0.925**	0.982
Siam-diff R^4 / 64	0.935	0.918	**0.926**	0.983

[a] The parameter w in formula (7) is set to {1,1,1,1,1}.

increase of 0.3% compared to Siam-diff, which is the best result achieved so far.

4.4 Pruning and Visualization

According to the characteristics of Siam-NestedUNet that can be pruned during testing, we make the analysis similar to [6]. Figure

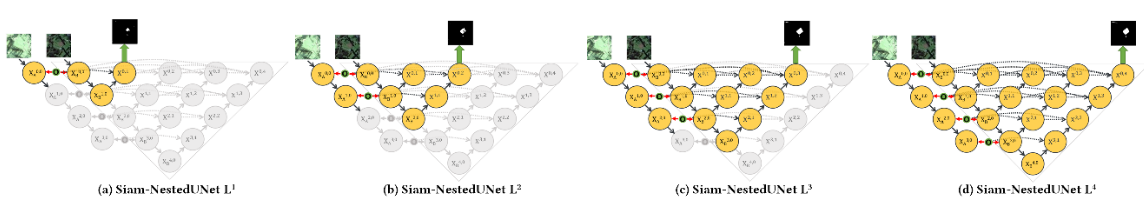

Figure 4: Structure pruning of Siam-NestedUNet. Using deep supervision strategy to train Siam-NestedUNet, the detection results can be obtained on multiple nodes $X^{0,j}$, which allows structure pruning during testing.

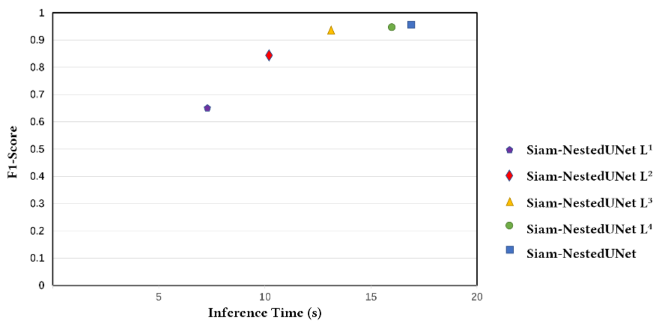

Figure 5: Comparison of F1-Score and inference time after structure pruning.

4 shows how to select branches to produce pruned architectures of varying complexity before ensemble.

Figure 5 shows the relationship between inference time and F1-Score corresponding to different modes. It can be observed that Siam-NestedUNet has the highest F1-Score, but it takes the longest inference time. When the batch size is set to 8 on a single NVIDIA Tesla V100, inferring one image with a size of 4725×2700 pixels requires about 17 seconds.

It is noteworthy that compared to Siam-NestedUNet, the inference time of Siam-NestedUNet L^3 is reduced by 4s, while the F1-Score is only reduced by 1.8 points. In order to compare the output of different pruning modes more directly, we visualize them as shown in Figure 6. We can observe that the results of Siam-NestedUNet L^1 and Siam-NestedUNet L^2 are frustrating. Only areas with very significant changes are predicted, and the edges of the target are very blurry. Siam-NestedUNet L^3 better balances the detection effect and the time required. The results of Siam-NestedUNet L^4 and Siam-NestedUNet are very similar, only differing in some minor details.

However, the proposed approach in this paper also has some problems, as shown in Figure 7. The green frames indicate the changes of small objects that are not clearly detected, and the red frames indicate error detection on the image boundary. To a certain extent, this is due to the limitation of the resolution, which is not enough to show the difference of small objects. For the detection of objects at the boundary of the image, it may be possible to optimize by inputting the context information of the target area, which is also one of our next work.

5 CONCLUSION

In this paper, we propose a fully convolutional siamese network, Siam-NestedUNet, which aims to combine siamese structures with

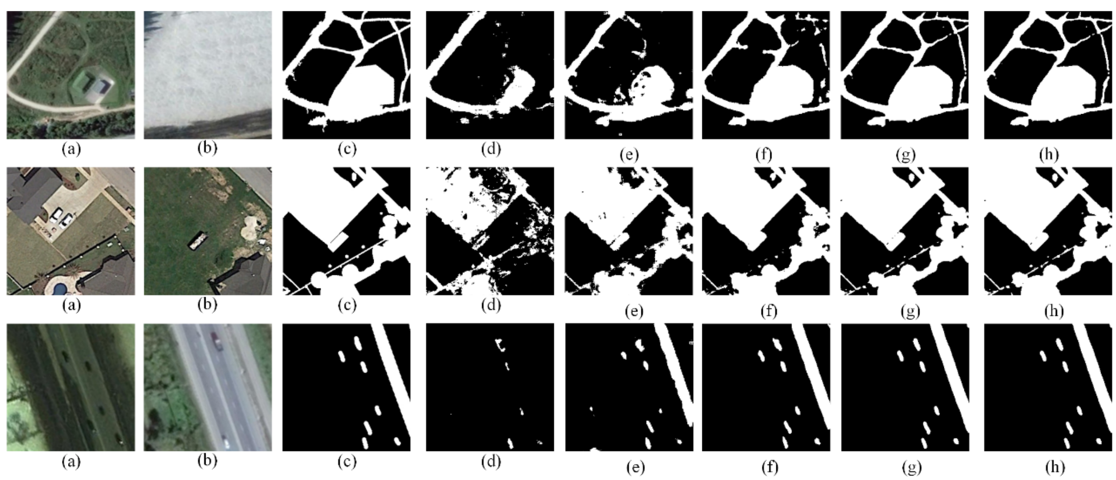

Figure 6: Visualized comparison of the results of various pruning modes. (a) and (b) are the original bi-temporal image, (c) is ground truth, (d), (e), (f) and (g) correspond to the prediction results of Siam-NestedUNet L^1 to Siam-NestedUNet L^4 respectively, and (h) is the final result of Siam-NestedUNet.

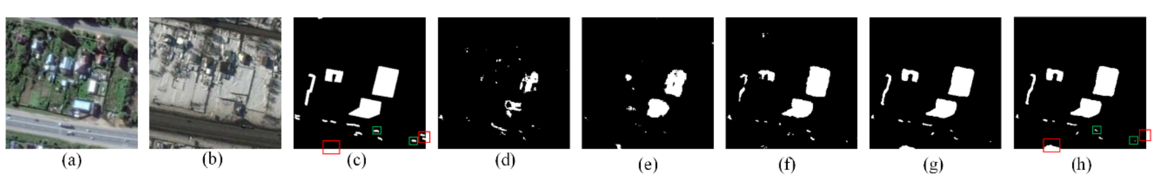

Figure 7: Problems and defects of the proposed approach. (a) and (b) are the original bi-temporal image, (c) is ground truth, (d), (e), (f) and (g) correspond to the prediction results of Siam-NestedUNet L^1 to Siam-NestedUNet L^4 respectively, and (h) is the final result of Siam-NestedUNet.

UNet++ for change detection. In the experiments, we use a variety of methods to improve the performance of the proposed method. Compared with other SOTA methods, our method obtains the best performance on quantitative metrics evaluation in CDD dataset. In the future, we will focus on solving the problem about small objects in change detection and enhance the practicability of our method.

ACKNOWLEDGMENTS

This work is partially supported by the National Natural Science Foundation of China (No. 61502278), the National Key R&D Program of China (No. 2018YFC0831002), the Key R&D Program of Shandong Province (No. 2018GGX101045)

REFERENCES

[1] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.
[2] Garcia-Garcia A, Orts-Escolano S, Oprea S, *et al.* A review on deep learning techniques applied to semantic segmentation[J]. arXiv preprint arXiv:1704.06857, 2017.
[3] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation[C]//International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015: 234-241.
[4] Daudt R C, Le Saux B, Boulch A. Fully convolutional siamese networks for change detection[C]//2018 25th IEEE International Conference on Image Processing (ICIP). IEEE, 2018: 4063-4067.
[5] Papadomanolaki M, Verma S, Vakalopoulou M, *et al.* Detecting urban changes with recurrent neural networks from multitemporal Sentinel-2 data[C]//IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium. IEEE, 2019: 214-217.
[6] Zhou Z, Siddiquee M M R, Tajbakhsh N, *et al.* Unet++: A nested u-net architecture for medical image segmentation[M]//Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. Springer, Cham, 2018: 3-11.
[7] Zhou Z, Siddiquee M M R, Tajbakhsh N, *et al.* Unet++: Redesigning skip connections to exploit multiscale features in image segmentation[J]. IEEE transactions on medical imaging, 2019, 39(6): 1856-1867.
[8] Peng D, Zhang Y, Guan H. End-to-end change detection for high resolution satellite images using improved unet++[J]. Remote Sensing, 2019, 11(11): 1382.
[9] Bromley J, Guyon I, LeCun Y, *et al.* Signature verification using a" siamese" time delay neural network[C]//Advances in neural information processing systems. 1994: 737-744.
[10] Daudt R C, Le Saux B, Boulch A, *et al.* Urban change detection for multispectral earth observation using convolutional neural networks[C]//IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, 2018: 2115-2118.
[11] Rahman F, Vasu B, Van Cor J, *et al.* Siamese network with multi-level features for patch-based change detection in satellite imagery[C]//2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP). IEEE, 2018: 958-962.
[12] Zhan Y, Fu K, Yan M, *et al.* Change detection based on deep siamese convolutional network for optical aerial images[J]. IEEE Geoscience and Remote Sensing Letters, 2017, 14(10): 1845-1849.
[13] Guo E, Fu X, Zhu J, *et al.* Learning to measure change: Fully convolutional siamese metric networks for scene change detection[J]. arXiv preprint arXiv:1810.09111, 2018.
[14] El Amin A M, Liu Q, Wang Y. Convolutional neural network features based change detection in satellite images[C]//First International Workshop on Pattern Recognition. International Society for Optics and Photonics, 2016, 10011: 100110W.
[15] Chen J, Yuan Z, Peng J, *et al.* DASNet: Dual attentive fully convolutional siamese networks for change detection of high resolution satellite images[J]. arXiv preprint arXiv:2003.03608, 2020.
[16] He K, Zhang X, Ren S, *et al.* Identity mappings in deep residual networks[C]//European conference on computer vision. Springer, Cham, 2016: 630-645.
[17] Lebedev M A, Vizilter Y V, Vygolov O V, *et al.* CHANGE DETECTION IN REMOTE SENSING IMAGES USING CONDITIONAL ADVERSARIAL NETWORKS[J]. International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, 2018, 42(2).
[18] Loshchilov I, Hutter F. Decoupled weight decay regularization[J]. arXiv preprint arXiv:1711.05101, 2017.
[19] He K, Zhang X, Ren S, *et al.* Delving deep into rectifiers: Surpassing human-level performance on imagenet classification[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1026-1034.