A Combined Loss-Based Multiscale Fully Convolutional Network for High-Resolution Remote Sensing Image Change Detection

Xinghua Li, *Senior Member, IEEE*, Meizhen He, Huifang Li, *Member, IEEE*,
and Huanfeng Shen, *Senior Member, IEEE*

***Abstract*— In the task of change detection (CD), high-resolution remote sensing images (HRSIs) can provide rich ground object information. However, the interference from noise and complex background information can also bring some challenges to CD. In recent years, deep learning methods represented by convolutional neural networks (CNNs) have achieved good CD results. However, the existing methods have difficulty in detecting the detailed change information of the ground objects effectively. The imbalance of positive and negative samples can also seriously affect the CD results. In this letter, to solve the above problems, we propose a method based on a multiscale fully convolutional neural network (MFCN), which uses multiscale convolution kernels to extract the detailed features of the ground object features. A loss function combining weighted binary cross-entropy (WBCE) loss and dice coefficient loss is also proposed, so that the model can be trained from unbalanced samples. The proposed method was compared with six state-of-the-art CD methods on the DigitalGlobe dataset. The experiments showed that the proposed method can achieve a higher *F1*-score, and the detection effect of the detailed changes was better than that of the other methods.**

***Index Terms*— Change detection (CD), combined loss function, high-resolution remote sensing images (HRSIs), multiscale fully convolutional neural network (MFCN).**

I. Introduction

CHANGE detection (CD) is the process of identifying differences in the state of an object or phenomenon by observing it at different times [1]. It is widely applied in land cover, urban expansion, natural resource monitoring, disaster monitoring, and military fields. With the rapid development of remote sensing technology, high-resolution remote sensing images (HRSIs) are now being widely used in CD. HRSIs can provide more detailed ground object information, but due to the limitations of the imaging conditions, interference caused by noise and background information also appears, which undoubtedly increases the difficulty of the ground object information processing.

In this letter, existing CD methods are roughly classified into traditional methods and learning-based methods. The traditional methods include algebra-based methods [2], [3], transformation-based methods [4]–[6], and advanced models [7]–[9]. The algebra-based methods use basic mathematical calculations like image differencing, image ratioing, and change vector analysis (CVA) to obtain a change map, and obtain the CD result through threshold segmentation. The transformation-based methods are principal component analysis (PCA) and tasseled cap transformation (TCT), which reduce the redundant information by performing a certain transformation on the original multitemporal remote sensing images, and then implement CD by analyzing the key information. The main idea behind the advanced CD models is the conversion of the image reflectance to physical parameters. In general, the traditional CD methods are usually easy to implement and understand, but their threshold selection relies on researcher's experience, and they cannot provide complete change matrices. Under these circumstances, the traditional CD methods have difficulty in extracting robust features from the complex ground object information of HRSIs.

In recent years, because of their excellent feature learning and expression capabilities, many scholars have attempted to introduce deep learning methods to the CD task, especially the methods represented by convolutional neural networks (CNNs). For example, a pixel-level street view change detection network (CDNet) was proposed by Alcantarilla *et al.* [10]. Based on a common U-net architecture, this method can detect change information, but the accuracy is not satisfactory. In [11], the three methods of FC-EF, FC-Siam-conc, and FC-Siam-diff were proposed. FC-EF is an early fusion network based on the U-net framework, which uses skip connections to combine high- and low-level features for joint learning. In order to learn the similar features between the original bitemporal images, FC-Siam-conc introduces a Siamese neural network into FC-EF. For emphasizing the difference features from FC-Siam-conc, after calculating the difference of the same layer features of the Siamese network, FC-Siam-diff uses skip connections to guide the network to learn the difference characteristics between the bitemporal images. These methods are not very effective when

Manuscript received June 14, 2021; accepted July 10, 2021. Date of publication July 30, 2021; date of current version January 6, 2022. The work was supported in part by the National Key Research and Development Program of China under Grant 2019YFB2102904 and in part by the National Natural Science Foundation of China (NSFC) under Grant 41701394. *(Corresponding author: Huanfeng Shen.)*

Xinghua Li is with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China (e-mail: lixinghua5540@whu.edu.cn).

Meizhen He, Huifang Li, and Huanfeng Shen are with the School of Resource and Environment Sciences, Wuhan University, Wuhan 430079, China (e-mail: hmz_whu@163.com; huifangli@whu.edu.cn; shenhf@whu.edu.cn).

Digital Object Identifier 10.1109/LGRS.2021.3098774

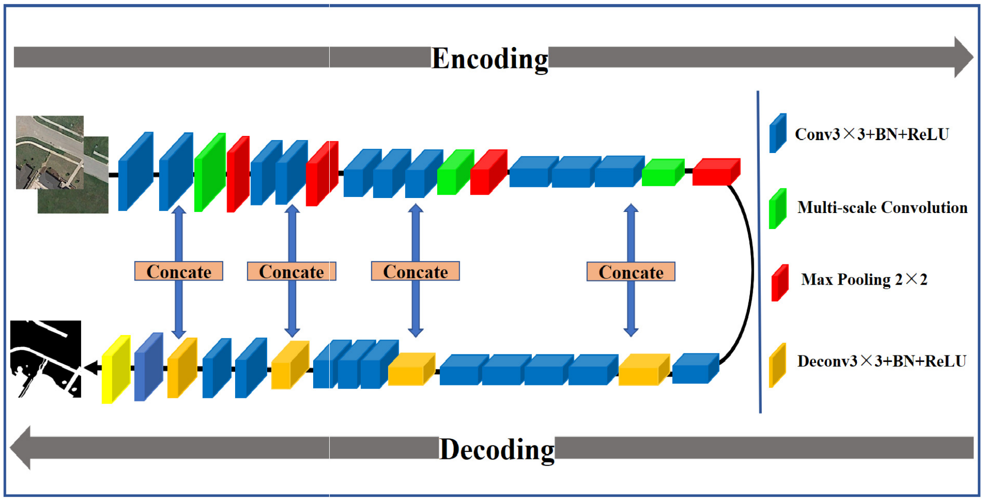

Fig. 1. Illustration of the proposed CD method.

used in high resolution, especially when detecting detailed features.

In summary, most of the existing CD methods are not ideal for HRSI CD. In order to solve this problem, an end-to-end CD method for HRSIs based on a multiscale fully convolutional neural network (MFCN) framework is proposed. The main contributions of this letter are as follows. Firstly, the multiscale convolutional architecture is used to improve the model's detection ability in small changed features such as roads. Secondly, by combining weighted binary cross-entropy (WBCE) loss and dice coefficient loss, the network's training efficiency on unbalanced samples is improved.

The rest of this letter is organized as follows. Section II introduces the basic principles of the proposed method, including the design of the network framework and the loss function. Section III describes the experiments conducted in this study. Finally, we draw our conclusions in Section IV.

II. Method

Fig. 1 shows the overall architecture of the proposed method, which is based on an end-to-end framework of a fully convolutional network. A multiscale convolution module with parallel branch structure is utilized to learn features of different scales. In the network training process, the combined loss function is used to alleviate the negative impact of the class imbalance on the CD results.

A. Fully CNN Framework

CNNs are usually connected with several fully connected layers after the convolutional layers, and the feature map generated by the convolutional layers is mapped into a fixed-length feature vector. The classic CNN structure is suitable for image-level classification and regression tasks, where a numerical probability description of the entire input image is required. Unlike CNNs, fully convolutional networks (FCNs) are obtained by replacing the last fully connected layer in the CNN with convolutional layers. A deconvolution layer is then used to up-sample the feature map obtained by the last convolutional layer to restore it to the same size as the input image, so that a prediction can be generated for each pixel while preserving the space in the original input image information. Pixel-by-pixel classification is then performed on the up-sampled feature map. Finally, a map that has been labeled pixel by pixel is output.

In this letter, we propose a CD framework based on an FCN. The model can be divided into two main parts, one of which is the encoding structure, and the other is the decoding structure. At the beginning, bitemporal images are stacked in the channel dimension and input into the network for the following training process. In the encoding part, the convolutional layers and the max pooling layers are used to learn different levels of the input feature. After the feature map passes through the max pooling layer, its length and width will be reduced to half of the original and its feature dimensionality will be doubled. Then, in the decoding part, the convolutional layers and the deconvolution layers are used to perform more abstract feature learning and gradually restore the size of the feature map to the size of the input image. After passing through the deconvolution layer, the feature map's length and width will become twice of the original and its feature dimensionality will be reduced by half. Skip connections are utilized between the encoding and decoding structures to combine the deep and shallow features for joint feature learning. At the end of the network structure, a softmax function is used to output the probability of each pixel being changed or unchanged. Finally, the class with the highest probability is selected to be the predicted class of the pixel.

B. Multiscale Convolution Module

Ordinary neural networks usually use convolutional layers with a filter size of 3 × 3 for feature learning. In order to learn multiscale features of the HRSI, and inspired by Szegedy *et al.* [12], we propose a parallel multibranch structure, which is called the multiscale convolution module. As shown in Fig. 2, this module is composed of four parallel branches, which are convolution kernels with the size of 1 × 1, 3 × 3, 5 × 5, and the max pooling layer with the size of 3 × 3. Among the different branches, the 1 × 1 convolution branch mainly learns pixel-level features. The 3 × 3 convolution

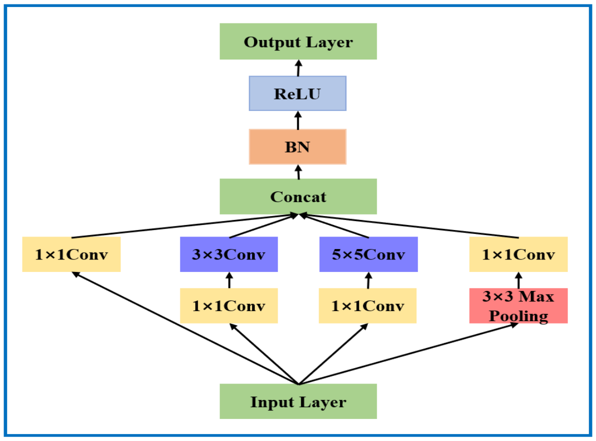

Fig. 2. Structure of the multiscale convolution module.

branch performs feature learning in the neighborhood. The 5 × 5 convolution branch performs feature learning in a larger receptive field. Finally, the last 3 × 3 max pooling layer focuses on the learning of salient features. Crucially, the 1 × 1 convolution used at the beginning of each branch is a bottleneck design. It mainly controls the number of parameters by compressing data in the channel dimensionality. Finally, all four branches are merged to obtain a general feature.

C. Loss Design

The class imbalance of the positive and negative samples is a common problem in CD. In fact, the number of negative samples (unchanged pixels) nearly always exceeds the number of positive samples (changed pixels). Therefore, in the process of neural network training, the network will learn more information about the negative samples that are not really of concern while neglecting the learning of positive samples. To solve this problem, a combined loss function composed of WBCE loss and dice coefficient loss is proposed to guide the network training process.

1) WBCE Los: WBCE loss is based on binary cross-entropy (BCE) loss, which is a measure of the difference between two probability distributions of a given random variable or event set. It is widely used in image classification and semantic segmentation tasks. CD is often regarded as a pixel-level binary classification problem, so binary cross-entropy loss can be directly used in CD tasks. The calculation formula for binary cross-entropy loss is as follows:

$$L_{\text{BCE}} = -\sum_{i=0}^{N} y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right) \quad (1)$$

where $y^{(i)}$ is the ith pixel ground-truth label, which has a value of 0 or 1, and $\hat{y}^{(i)}$ is the probability of the ith pixel being predicted as changed or unchanged. In the process of classifying the ith pixel into changed or unchanged based on bitemporal images, the number of changed pixels will be much smaller than that of unchanged pixels. The model tends to directly classify pixels into the unchanged type, which seriously affects the CD result. To solve this problem, WBCE loss is obtained by weighting the binary cross-entropy loss. The calculation formula is as follows:

$$L_{\text{WBCE}} = -\sum_{i=0}^{N} \beta y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right) \quad (2)$$

where β is the proportion of negative samples in the total sample. The purpose is that, when the number of samples is unbalanced, the model can better take the learning of positive samples into account and obtain a better CD effect.

2) Dice Coefficient Loss: Dice coefficient loss is a function used to measure the similarity of different variable sets, so it is commonly utilized to calculate the similarity between pixels of two sample sets. The calculation formula is as follows:

$$L_{\text{Dice}} = 1 - 2y^{(i)}\hat{y}^{(i)}/(y^{(i)} + \hat{y}^{(i)}). \quad (3)$$

Dice coefficient loss is often used when the samples are extremely unbalanced. However, if it is used in a general situation, it can make the training unstable.

3) Combined Loss:

$$L_c = L_{\text{WBCE}} + \lambda L_{\text{Dice}} \quad (4)$$

where λ is a parameter that balances L_{WBCE} and L_{Dice}.

III. EXPERIMENTS

This section is divided into two main parts. In Section III-A, the HRSI CD dataset [13] and the optimization method are introduced. In Section III-B, the quantitative evaluation metrics are presented, followed by comparisons between the proposed method and the six state-of-the-art CD methods proposed in [10], [11], [14], and [15]. We also describe the experiments conducted on the choice of loss function and the ablation experiments conducted with regard to the data augmentation.

A. Implementation Details

1) Dataset Description: DigitalGlobe is a real remote sensing dataset built by Lebedev *et al.* [13] in 2018, for which the raw data were obtained from Google Earth. The dataset has strong seasonal variation, and its spatial resolution varies from 3 to 100 cm. The DigitalGlobe dataset contains a large number of different types and scales of changed features, such as cars, buildings, roads, and different land-cover types, which are quite challenging for the CD task. The dataset consists of 16 000 sets of images with a size of 256 × 256, including 10 000 sets of training data, 3000 sets of verification data, and 3000 sets of test data.

2) Optimization: The proposed method was implemented with TensorFlow as the backend, which was powered by a workstation with an Intel(R) Core (TM) i7-9700K CPU at 3.6 GHz, 32 GB of RAM, and a single NVIDIA GeForce RTX 2080 Ti GPU. During the training process, the Adam optimizer with a learning rate of 0.004 was applied. Based on the GPU memory, the batch size was set to 40 for 100 000 iterations, and the learning rate was reduced by 0.5 after every 2500 iterations. As the proposed architecture is an FCN-based model, it is easy to train the model in an end-to-end manner for an arbitrary size of input image.

TABLE I
QUANTITATIVE EVALUATION OF THE DIFFERENT APPROACHES

Method	P	R	F1	OA	mIoU
PCA-kmeans [14]	0.113	0.532	0.187	0.454	0.260
SFA [15]	0.168	0.211	0.187	0.644	0.366
CDNet [10]	0.774	0.577	0.661	0.930	0.709
FC-EF [11]	0.860	0.688	0.764	0.950	0.782
FC-Siam-conc [11]	0.920	0.768	0.837	0.965	0.841
FC-Siam-diff [11]	0.929	0.774	0.844	0.966	0.847
Proposed method	**0.935**	**0.888**	**0.911**	**0.980**	**0.907**

B. Results and Evaluation

1) Evaluation Metrics: In order to evaluate the performance of the proposed method, the method was evaluated by comparing the prediction result with the ground-truth labels, and five evaluation metrics were calculated. The evaluation metrics are the precision (P), recall (R), *F1*-score (*F1*), overall accuracy (OA), and mIoU. In the CD task, a high precision rate represents a low false detection rate, while a high recall rate means a low missed detection rate. The $F1$, OA, and mIoU are metrics that can reflect the comprehensive performance of the method, where the higher the score, the better the performance. The formulas for these five metrics are as follows:

$$P = \mathrm{TP}/(\mathrm{TP}+\mathrm{FP}) \tag{5}$$

$$R = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN}) \tag{6}$$

$$F1 = 2\mathrm{PR}/(P+R) \tag{7}$$

$$\mathrm{OA} = (\mathrm{TN}+\mathrm{TP})/(\mathrm{TN}+\mathrm{FN}+\mathrm{FP}+\mathrm{TP}) \tag{8}$$

$$\mathrm{mIoU} = (\mathrm{TP}/(\mathrm{FN}+\mathrm{FP}+\mathrm{TP})+\mathrm{TN}/(\mathrm{FP}+\mathrm{FN}+\mathrm{TN}))/2 \tag{9}$$

where TP is the number of pixels that was correctly classified as changed, TN is the number of pixels that was correctly classified as unchanged, FP is the number of pixels that was classified as changed but was not actually changed, and FN is the number of pixels that was mistakenly classified as unchanged.

2) Comparison of the Proposed Method and the Six Other State-of-the-Art CD Methods: The proposed method was compared with the six other state-of-the-art CD networks proposed in [10], [11], [14], and [15] as listed in Table I. From Table I, it can be seen that the proposed method surpasses the six comparison methods in all five metrics and exceeds FC-Siam-diff by 0.5%, 11.3%, 6.6%, 1.4%, and 6% in P, R, *F1*, OA, and mIoU, respectively. By taking advantage of the combined loss function, as shown in Figs. 3 and 4, the proposed method can detect changed samples effectively when the positive and negative samples are extremely imbalanced. For instance, the changed roads and linear objects in areas 1 and 2 can be detected, while the comparison methods cannot detect these changed features effectively. Furthermore, Figs. 5 and 6 show that when the sample distribution is relatively balanced, both the proposed method and the comparison methods can better implement CD. However, due to the proper adoption of the multiscale convolution module, in the detection of the outlines and details of the ground object features in areas 3 and 4, the proposed method shows a much better performance.

3) Experiment With Different Loss Functions: In this section, we evaluate the effect on the model of using BCE loss, focal loss, WBCE loss, and the proposed combined loss. It can be found from Table II that the proposed model achieves the best effect when the combined loss is used, and the worst effect is seen when only the binary cross-entropy loss is used.

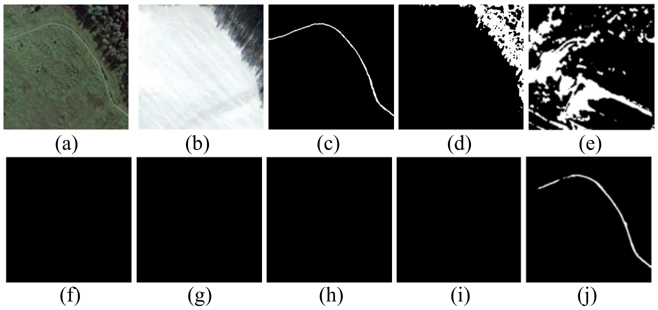

Fig. 3. Visual comparison of the CD results obtained using the different approaches for area 1. (a) Image T1. (b) Image T2. (c) Ground-truth map. (d) PCA-*k*-means. (e) SFA. (f) CDNet. (g) FC-EF. (h) FC-Siam-conc. (i) FC-Siam-diff. (j) Proposed method. The changed parts are marked in white, while the unchanged parts are in black.

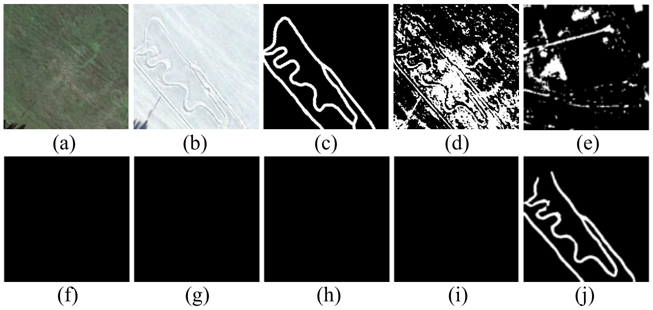

Fig. 4. Visual comparison of the CD results obtained using the different approaches for area 2. (a) Image T1. (b) Image T2. (c) Ground-truth map. (d) PCA-*k*-means. (e) SFA. (f) CDNet. (g) FC-EF. (h) FC-Siam-conc. (i) FC-Siam-diff. (j) Proposed method. The changed parts are marked in white, while the unchanged parts are in black.

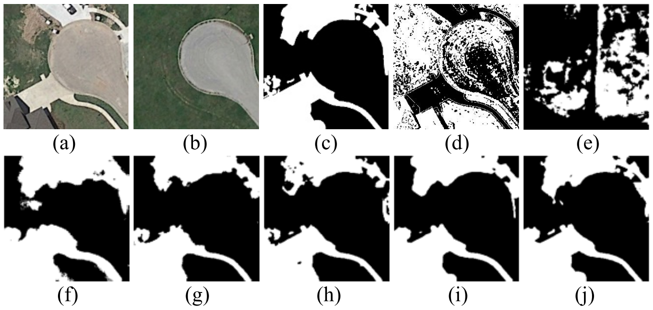

Fig. 5. Visual comparison of the CD results obtained using the different approaches for area 3. (a) Image T1. (b) Image T2. (c) Ground-truth map. (d) PCA-*k*-means. (e) SFA. (f) CDNet. (g) FC-EF. (h) FC-Siam-conc. (i) FC-Siam-diff. (j) Proposed method. The changed parts are marked in white, while the unchanged parts are in black.

TABLE II
QUANTITATIVE EVALUATION OF THE DIFFERENT LOSS FUNCTIONS

Loss Functions	P	R	F1	OA	mIoU
BCE	0.874	0.835	0.854	0.966	0.854
Focal	0.878	0.873	0.876	0.971	0.873
WBCE	0.912	0.867	0.889	0.974	0.886
Combined	**0.935**	**0.888**	**0.911**	**0.980**	**0.907**

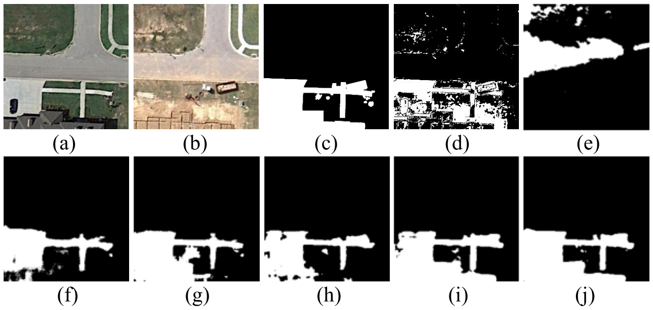

Fig. 6. Visual comparison of the CD results obtained using the different approaches for area 4. (a) Image T1. (b) Image T2. (c) Ground-truth map. (d) PCA-*k*-means. (e) SFA. (f) CDNet. (g) FC-EF. (h) FC-Siam-conc. (i) FC-Siam-diff. (j) Proposed method. The changed parts are marked in white, while the unchanged parts are in black.

TABLE III

QUANTITATIVE EVALUATION OF THE DIFFERENT λ

Λ	P	R	F1	OA	mIoU
0	0.912	0.867	0.889	0.974	0.886
0.25	0.917	0.873	0.894	0.976	0.891
0.50	**0.935**	**0.888**	**0.911**	**0.980**	**0.907**
0.75	0.761	0.864	0.809	0.952	0.813
1	0.915	0.884	0.899	0.977	0.895

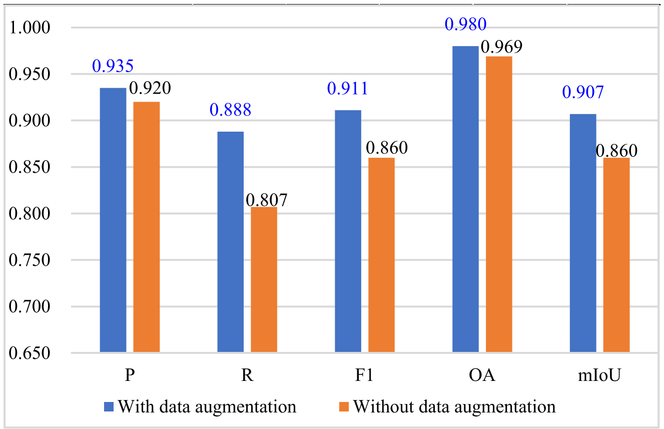

Fig. 7. Effect of data augmentation on the accuracy of our method.

4) Effect of λ on the Accuracy of Our Method: The parameter λ in the combined loss function is essential to balance the WBCE loss and the dice coefficient loss. In order to verify the sensitivity of λ, we varied its value from 0 to 1 and obtained results shown in Table III. When λ was set to 0.75, the effect of the method was the worst. While λ was set to 0.5, the accuracy of all metrics achieved the highest, indicating that when λ is 0.5, the experiment can achieve the best effect.

5) Effect of the Data Augmentation Strategies: Because the proposed method requires a large amount of data for parameter learning, and to prevent the model from overfitting, the 10 000 sets of training data were augmented in this experiment. Specifically, the images were randomly flipped up and down, and left and right. In addition, the hue, brightness, and saturation of the images were randomly augmented. Random rotations of 90°, 180°, and 270° were also applied. It can be seen from Fig. 7 that, after the data augmentation, the results of the proposed model are improved by 1.5%, 8.1%, 5.1%, 1.1%, and 4.7% in the five metrics of *P*, *R*, *F1*, OA, and mIoU, respectively. Therefore, it can be concluded that the data augmentation strategies used in this experiment are meaningful for the proposed method.

IV. CONCLUSION

In this letter, we have presented a CD method based on an FCN framework. By adding multiscale convolution modules to learn features of different scales, the effect of HRSI CD can be improved. The combination of WBCE loss and dice coefficient loss can relieve the negative impact of sample imbalance on the detection results. Experiments on the DigitalGlobe dataset fully verified that the proposed method is superior to the six other state-of-the-art CD methods. The proposed method is a supervised learning method, which relies on lots of ground-truth samples to train the model. In our future work, we will attempt to perform CD with fewer samples and also without ground-truth samples.

REFERENCES

[1] A. Singh, "Review article digital change detection techniques using remotely-sensed data," *Int. J. Remote Sens.*, vol. 10, no. 6, pp. 989–1003, 1989.

[2] L. Ke, Y. Lin, Z. Zeng, L. Zhang, and L. Meng, "Adaptive change detection with significance test," *IEEE Access*, vol. 6, pp. 27442–27450, 2018.

[3] V. Ferraris, N. Dobigeon, Q. Wei, and M. Chabert, "Detecting changes between optical images of different spatial and spectral resolutions: A fusion-based approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1566–1578, Mar. 2018.

[4] V. Sadeghi, F. F. Ahmadi, and H. Ebadi, "Design and implementation of an expert system for updating thematic maps using satellite imagery (case study: Changes of Lake Urmia)," *Arabian J. Geosci.*, vol. 9, no. 4, pp. 1–17, Apr. 2016.

[5] A. K. Thakkar, V. R. Desai, A. Patel, and M. B. Potdar, "An effective hybrid classification approach using tasseled cap transformation (TCT) for improving classification of land use/land cover (LU/LC) in semi-arid region: A case study of Morva–Hadaf watershed, Gujarat, India," *Arabian J. Geosci.*, vol. 9, no. 3, pp. 1–13, Mar. 2016.

[6] G. Xu, H. Li, Y. Zang, L. Xie, and C. Bai, "Change detection based on IR-MAD model for GF-5 remote sensing imagery," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 768, no. 7, 2020, Art. no. 072073.

[7] L. Yan, W. Xia, Z. Zhao, and Y. Wang, "A novel approach to unsupervised change detection based on hybrid spectral difference," *Remote Sens.*, vol. 10, no. 6, p. 841, May 2018.

[8] Y. Zeng, M. E. Schaepman, B. Wu, J. G. P. W. Clevers, and A. K. Bregt, "Scaling-based forest structural change detection using an inverted geometric-optical model in the Three Gorges region of China," *Remote Sens. Environ.*, vol. 112, no. 12, pp. 4261–4271, 2008.

[9] J. Meola, M. T. Eismann, R. L. Moses, and J. N. Ash, "Detecting changes in hyperspectral imagery using a model-based approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 7, pp. 2647–2661, Jul. 2011.

[10] P. F. Alcantarilla, S. Stent, G. Ros, R. Arroyo, and R. Gherardi, "Street-view change detection with deconvolutional networks," *Auton. Robots*, vol. 42, no. 7, pp. 1301–1322, 2018.

[11] R. C. Daudt, B. L. Saux, and A. Boulch, "Fully convolutional Siamese networks for change detection," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 4063–4067.

[12] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9.

[13] M. Lebedev, Y. V. Vizilter, O. Vygolov, V. Knyaz, and A. Y. Rubis, "Change detection in remote sensing images using conditional adversarial networks," *Int. Archives Photogram., Remote Sens. Spatial Inf. Sci.*, vol. 42, no. 2, pp. 565–571, 2018.

[14] T. Celik, "Unsupervised change detection in satellite images using principal component analysis and k-means clustering," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 4, pp. 772–776, Oct. 2009.

[15] C. Wu, B. Du, and L. Zhang, "Slow feature analysis for change detection in multispectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2858–2874, May 2014.