IGARSS 2021 - 2021 IEEE International Geoscience and Remote Sensing Symposium | 978-1-6654-0369-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/IGARSS47720.2021.9555003

GRAPH-LEVEL NEURAL NETWORK FOR SAR IMAGE CHANGE DETECTION

Rongfang Wang[1], *Liang Wang*[1], *Pinghai Dong*[2], *Licheng Jiao*[1], *Jia-Wei Chen*[1, *]

[1]Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, School of Artificial Intelligence, Xidian University, Xi'an 710071, China.
[2] Tsinghua Shenzhen International Graduate School, Shenzhen 518055, China.

ABSTRACT

Graph neural network (GNN) has been widely applied to computer vision as well as remote sensing image analysis. In this paper, we propose an end-to-end graph-level neural network (GLNN) for SAR image change detection. In the proposed method, a GNN is applied to exploit the local structure of an image patch at a graph-level and learn a more discriminative representation. Then, based on these graph representations, change detection is conducted by training an end-to-end neural network. Our method is verified on four cross-dataset of SAR image and compared with three state-of-art deep learning SAR image change detection methods. The experimental results show that the proposed GLNN outperforms other compared methods.

Index Terms— change detection, SAR image, graph neural network, graph-lever classification, cross-dataset.

1. INTRODUCTION

Over the past decades, synthetic aperture radar (SAR) images have been widely employed in change detection tasks since SAR is a microwave sensor working without the limitations of seasons and weather. Most traditional SAR image change detection methods are developed based on the three steps pipeline proposed in [1]. The detection accuracy of traditional methods greatly depends on the quality of a difference image (DI). It is challenging to extract accurate and clear features from the noisy DI. In the recent decade, due to the powerful ability to exploit essential and robust features, deep learning [2] has been successful performed in computer vision and remote sensing image interpretation. Some deep learning-based change detection methods [3,4] have been developed and obtained a better detection performance than traditional ones.

Recently, with the convincing performance and high interpretability, graph neural network (GNN) [5] has become a popular deep learning approach and achieved satisfactory results in many applications such as semantic segmentation [6], visual question answering [7], and point cloud classification and segmentation [8]. Furthermore, Saha et al. [9] proposed a semi-supervised graph convolutional network (ssGCN) for change detection, where a graph is proposed to exploit the structures of bitemporal Very High Resolution (VHR) images. The multiscale parcels are obtained via a superpixel segmentation algorithm and treated as nodes to form a region adjacency graph. The information from the labeled parcels is propagated to the unlabeled ones over training iterations, ssGCN has shown good performance for VHR image change detection.

However, SAR image quality is degraded as speckle noise and it is quite challenging to exploit its structures. The ssGCN cannot be directly applied to bitemporal SAR images due to the inaccurate superpixel segmentation, which will affect the node-based ssGCN. For another, the number of nodes in the bitemporal images must be consistent for the ssGCN, which is a rigorous condition.

To cope these limitations, in this paper, we propose an end-to-end graph-level neural network (GLNN) for bitemporal SAR image change detection. Specifically, we first stack the two temporal SAR images and the difference image to build three-channel blocks pixel by pixel and then construct a graph on each block. The graph representation means that we utilize GNN to exploit the local structure of each pixel neighborhood block at a graph-level and learn an entire graph representation for each block. Then, based on these representations, change detection is conducted by training an end-to-end neural network. Our methodological contributions are: 1) Constructing graph on the three-channel block, which can simultaneously exploit the shared information associated with the bitemporal images and DI, moreover utilize the spatial contextual information of the pixel neighborhood to reduce the influence of speckle noise. 2) Graph-level GNN change detection, which can use the feature propagation and aggregation of an entire graph nodes to learn more discriminative representations for each block. We compare our proposed method with three deep learning-based methods on four cross-dataset of the SAR image. The overall experimental results show that the proposed GLNN achieves outstanding performance.

* Corresponding author: Jia-Wei Chen, jawaechan@gmail.com.

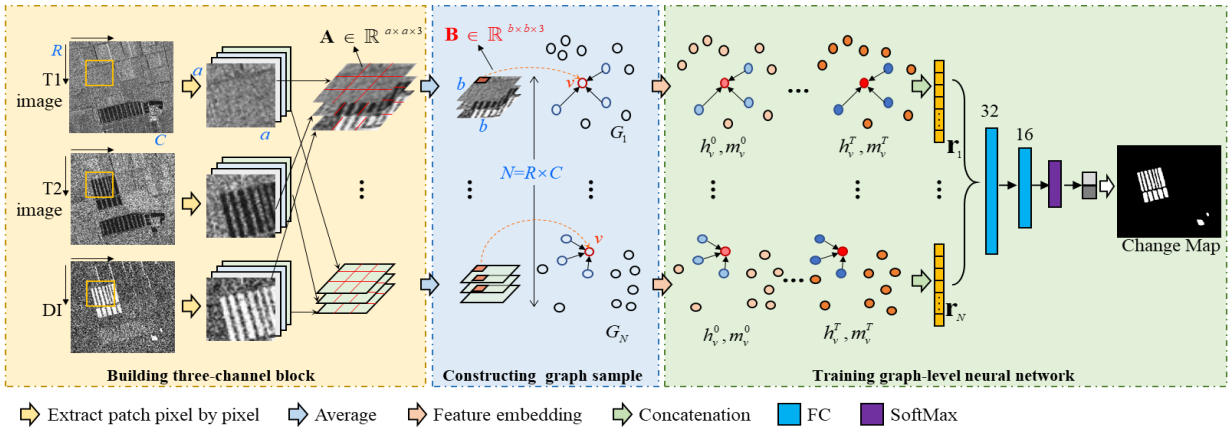


Fig.1. The framework of the proposed GLNN.

2. METHODOLOGY

The framework of our method GLNN is illustrated in Fig.1. GLNN consists of three phases: (1) building three-channel block; (2) constructing graph sample; and (3) training graph-level neural network. The details of each stage are described in the following.

2.1. Building Three-Channel Block

Given a set of bitemporal SAR images $\mathbf{I}_1$ and $\mathbf{I}_2$ with a size of $R\times C$, which are obtained over the same region and have been aligned. The DI is generated by the method of log-ratio:

$$\mathbf{I}_{DI} = \left|\log(\mathbf{I}_1 + 1) - \log(\mathbf{I}_2 + 1)\right| \tag{1}$$

Then both the bitemporal SAR images and the generated DI are zero-padded with the width of $(a\text{-}1)/2$ pixels along the boundary, where a is odd and $a \geq 3$. For $N=R\times C$ pixels, the overlapping patches around each pixel of size $a\times a$ are selected from the same position of padded $\mathbf{I}_1$, $\mathbf{I}_2$ and $\mathbf{I}_{DI}$ in turn. And then, three patches are stacked to get N three-channel blocks $\mathbf{A} \in \mathbb{R}^{a\times a\times 3}$.

2.2. Constructing Graph Sample

The graph sample $G=\{V,E\}$ on each three-channel block $\mathbf{A}$ is constructed as follows. For utilizing the spatial contextual information of the pixel neighborhood to reduce the influence of speckle noise, we partition each channel patch of $\mathbf{A}$ into $s\times s$ nonoverlapping sub-patches. And then, we average each sub-patch of each channel separately to get a new block $\mathbf{B} \in \mathbb{R}^{b\times b\times 3}$, where $b=a/s$. Fixing the channel dimension, we reshape the block $\mathbf{B}$ into a matrix with size $3\times b^2$, and each column vector $\bar{\mathbf{x}}_v = [\bar{x}_v^1, \bar{x}_v^2, \bar{x}_v^3]^T, v = 1,2,...,b^2$ is used as a node, the number of graph node is $|V| = b^2$. The edges of each node are built by k-nearest neighbors (KNN) in the space of b^2 nodes. The label of the constructed graph sample G is the value of the reference image corresponding to the central pixel of block $\mathbf{A}$.

2.3. Training Graph-Level Neural Network

Based on the graph sample set $\mathbf{G}=\{G_i=\{V,E\}, i=1,...,N\}$, we train GLNN by the following two steps:

Step 1: learning graph representation. Let $h_v^0 = \bar{\mathbf{x}}_v$ as the initial feature of the node v, and $h_v^t \in \mathbb{R}^{3\times 1}$ as the hidden representation which denotes the state of node v at the time step t. For each graph sample $G=\{V,E\}$, motivated by [7], we update h_v^t according to Eq. (2) and (3):

$$m_v^t = \frac{1}{k}\sum_{n\in\Omega_v} \mathcal{F}_1\left(h_n^t \mid n \in \Omega_v\right) \tag{2}$$

$$h_v^{t+1} = \mathcal{F}_2(\text{cat}[h_v^t, m_v^t]) \tag{3}$$

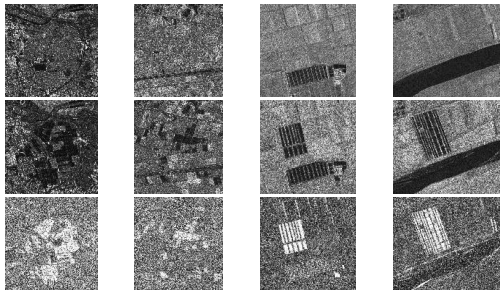
where cat[· , ·] indicates concatenating two vectors. In the time step t, we use Eq. (2) to compute the node v's message $m_v^t \in \mathbb{R}^{3\times 1}$, which is aggregated from the neighbors set Ω_v of node v, k is the number of neighbors. $\mathcal{F}_1(\cdot)$ denotes one fully connected layer with the non-linear activation function ReLU, which is used to compute message from neighbors. Then we use Eq. (3) to update the hidden representation h_v^{t+1} of the node v at the time step t+1. $\mathcal{F}_2(\cdot)$ denotes another one fully connected layer as the updating function. At one iteration we do the same operations for all the nodes in turn. After T iterations, all the states of b^2 nodes are concatenated to get the final representation vector of each graph sample $\mathbf{r} = \text{cat}[h_1^T, h_2^T, ..., h_{b^2}^T]^T \in \mathbb{R}^{(3\times b^2)\times 1}$.

Step 2: classifying graph sample. We train a three layers MLP based on the graph representation set $\mathbf{R}=\{\mathbf{r}_i, i=1,2,...,N\}$ to classify a graph sample into the change class or un-change class. The loss function is the softmax cross-entropy. The model is trained with the back-propagation algorithm.

3. EXPERIMENTS AND ANALYSIS

3.1. Datasets

The proposed method is verified on four datasets of bitemporal SAR images [4] shown in Fig.2. Two scenes (YR-A and YR-B) are from bitemporal Yellow River SAR images acquired by the Radarsat-2 satellite in 2008 and 2009, respectively. The sizes of these two image sets are 306×291 and 400×350, respectively. The other two scenes (Sendai-A and Sendai-B) are acquired from the Sendai earthquake in Japan. The sizes of these two image sets are 590×687 and 689×734, respectively.



(a)Sendai-A (b)Sendai-B (c)YR-A (d)YR-B

Fig. 2. Four datasets of bitemporal SAR images. The first two rows are bitemporal images and the last row is the DI.

3.2. Parameter Setup

For GLNN, we set the patch size a =15, and the sub-patch size s=3. The number of nearest neighbors k=3. The maximum iterations T=7. For the three comparison methods, all the parameters are set to the given value according to the corresponding work. Moreover, we verify the performance of all methods by leave-one-out-manner, i.e., the network trained on YR-B, Sendai-A and Sendai-B is applied to an unknown testing dataset YR-A. For YR-B Sendai-A and Sendai-B, the training and testing datasets are crossed in the same way. Note that the all training parameters on the four datasets remain the same.

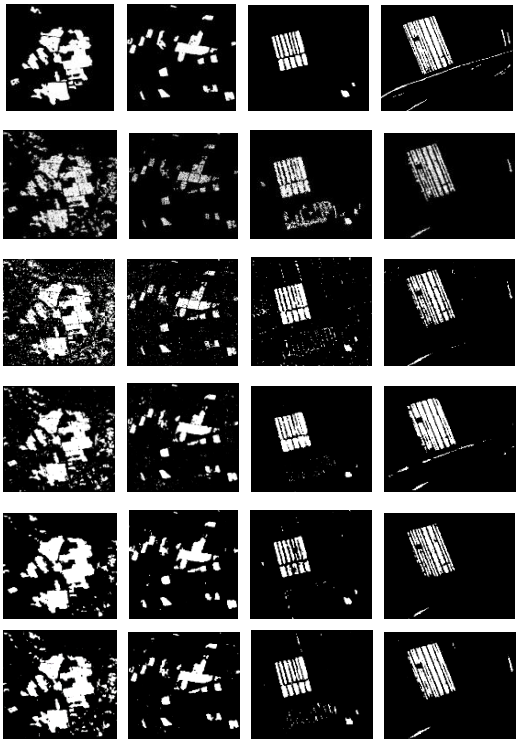
3.3. Detection Results Analysis

We compared our proposed GLNN with three deep learning-based methods, CNN [3] PSP-Net [4], and DNN [10], where CNN and DNN are the two-channel methods based on bitemporal SAR images, and PSP-Net and our GLNN are the three-channel methods based on bitemporal SAR images and DI image. To analyze the impact of graph representation on GLNN performance, we deleted this part, i.e., directly fed three-channel blocks into the three layers MLP. This modified model was named MLP. The visual comparison results were shown in Fig.3. The quantitative comparison results were shown in Fig. 4. The performance was evaluated by probabilistic missed alarm (pMA), probabilistic false alarm (pFA), and kappa coefficient, where pFA (pMA) are calculated by the ratios between FA (MA) and the number of unchanged pixels (NC).

From the above comparisons, it was shown that our GLNN got better Kappa than DNN, CNN, MLP , and PSP-Net on the Sendai-B YR-A and YR-B datasets. Moreover, on the Sendai-A dataset, GLNN performs better than MLP, DNN, and CNN, PSP-Net and the GLNN got the almost same Kappa on the Sendai-A.

3.4 Effect of Parameter *a*

First, we conduct GLNN with different values of a. The results with different values of a are shown in Fig.4(d). The parameters s is used to extract features for each node during the initialization of the graph construction. The parameters s set to small, resulting in the initialization features of each node are too similar, which is not conductive to subsequent iteration, and the s set to large, resulting in each node can't extract enough structured information. So, in this ablation study, we keep the parameters s fixed, s = 3.



(a)Sendai-A (b)Sendai-B (c)YR-A (d)YR-B

Fig. 3. The visual results of the comparison methods. The first row is the reference image. From the second row to bottom, they are the visual results based on the method DNN, CNN, MLP, PSP-Net, and GLNN, respectively.

4. CONCLUSION

In this paper, we proposed a novel GLNN model for bitemporal SAR image change detection, which constructs a graph on a three-channel pixel neighborhood block. Our method is verified on four cross-dataset of SAR image and compared with three state-of-art deep learning-based methods. The experimental results show that the proposed GLNN obtains outstanding performance，and the performance of the three-channel method added DI is generally better than the method only with bitemporal images.

5. ACKNOWLEDGMENT

This work was supported by the State Key Program of National Natural Science of China (No. 61836009), the Fujian Key Laboratory of Data Science and Statistics (Minnan Normal University) (No.2020L0707).

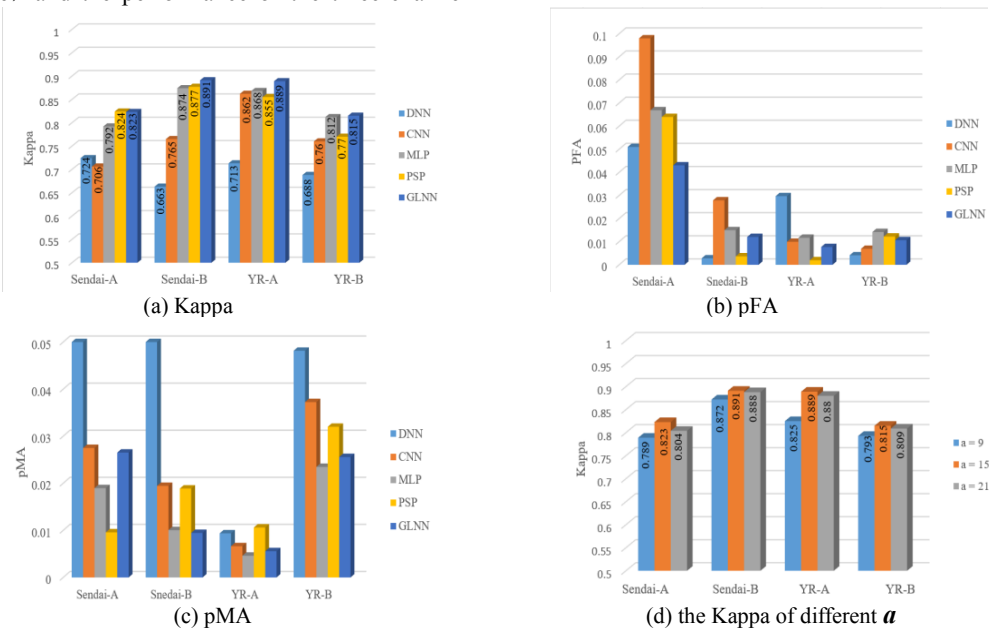


Fig. 4. (a)-(c) are the quantitative results of the comparison methods, and (d) is the quantitative results of the different a in GLNN .

6. REFERENCES

[1] Lorenzo Bruzzone, and F. Prieto Diego. "Automatic analysis of the difference image for unsupervised change detection." *IEEE Transactions on Geoscience and Remote sensing*, vol. 38, no. 3, pp. 1171-1182, 2000.

[2] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." *nature*, vol. 521, no. 7553, pp. 436-444, 2015.

[3] Yangyang Li, Cheng Peng, Yanqiao Chen, Licheng Jiao, Linhao Zhou, and Ronghua Shang. "A deep learning method for change detection in synthetic aperture radar images." *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 8, pp. 5751-5763, 2019.

[4] Jia-Wei Chen, Rongfang Wang, Fan Ding, Bo Liu, Licheng Jiao, and Jie Zhang. "A Convolutional Neural Network with Parallel Multi-Scale Spatial Pooling to Detect Temporal Changes in SAR Images." *Remote Sensing*, vol. 12, no. 10, pp. 1619, 2020.

[5] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, and Maosong Sun. "Graph neural networks: A review of methods and applications." *arXiv: 1812.08434* (2018)

[6] Xiaojuan Qi, Renjie Liao, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. "3D graph neural networks for RGBD semantic segmentation." *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5199-5208, 2017.

[7] Gao, Difei, Ke Li, Ruiping Wang, Shiguang Shan, and Xilin Chen. "Multi-Modal Graph Neural Network for Joint Reasoning on Vision and Scene Text." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12746-12756, 2020.

[8] Xiang Gao, Wei Hu, and Guo-Jun Qi. "GraphTER: Unsupervised Learning of Graph Transformation Equivariant Representations via Auto-Encoding Node-wise Transformations." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7163-7172, 2020.

[9] Sudipan Saha, Lichao Mou, Xiao Xiang Zhu, Francesca Bovolo, and Lorenzo Bruzzone. "Semisupervised Change Detection Using Graph Convolutional Network." *IEEE Geoscience and Remote Sensing Letters*, pp. 1-5, 2020.

[10] Maoguo Gong, Jiaojiao Zhao, Jia Liu, Qiguang Miao, and Licheng Jiao. "Change detection in synthetic aperture radar images based on deep neural networks." *IEEE transactions on neural networks and learning systems*, vol. 27, no. 1, pp. 125-138, 20