

# A Generative Discriminatory Classified Network for Change Detection in Multispectral Imagery

Maoguo Gong<sup>ID</sup>, Senior Member, IEEE, Yuelei Yang, Tao Zhan<sup>ID</sup>, Xudong Niu<sup>ID</sup>, and Shuwei Li

**Abstract**—Multispectral image change detection based on deep learning generally needs a large amount of training data. However, it is difficult and expensive to mark a large amount of labeled data. To deal with this problem, we propose a generative discriminatory classified network (GDCN) for multispectral image change detection, in which labeled data, unlabeled data, and new fake data generated by generative adversarial networks are used. The GDCN consists of a discriminatory classified network (DCN) and a generator. The DCN divides the input data into changed class, unchanged class, and extra class, i.e., fake class. The generator recovers the real data from input noises to provide additional training samples so as to boost the performance of the DCN. Finally, the bitemporal multispectral images are input to the DCN to get the final change map. Experimental results on the real multispectral imagery datasets demonstrate that the proposed GDCN trained by unlabeled data and a small amount of labeled data can achieve competitive performance compared with existing methods.

**Index Terms**—Change detection, deep learning, generative adversarial networks (GANs), multispectral imagery.

## I. INTRODUCTION

**I**MAGE change detection is such an important technique aimed at identifying the changes occurred on the ground by comparing and analyzing two (or more) images that are obtained at different observation times over the same area [1]. It has been widely applied to various fields, such as natural disaster evaluation [2] and agricultural assessment [3]. Multispectral imagery is usually obtained by a passive optical sensor, which captures the information of the ground objects in different spectral bands. Thus, the multispectral remote sensing images are often rich in spectral information, which has become one of the main data source for detecting the changes occurred on the ground [4].

Manuscript received May 23, 2018; revised September 18, 2018 and November 4, 2018; accepted December 4, 2018. Date of publication January 2, 2019; date of current version January 21, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61772393, in part by the National Key Research and Development Program of China under Grant 2017YFB0802200, and in part by the Key Research and Development Program of Shaanxi Province under Grant 2018ZDXM-GY-045. (Corresponding author: Maoguo Gong.)

M. Gong, Y. Yang, T. Zhan, and X. Niu are with the School of Electronic Engineering, Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an 710071, China (e-mail: gong@ieee.org; 1259437724@qq.com; OMEGAZhanT@gmail.com; xudong.niu@qq.com).

S. Li is with the School of Electronic Science and Engineering, Southeast University, Nanjing 210096, China (e-mail: 840142797@qq.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JSTARS.2018.2887108

As a typically used technique, change vector analysis (CVA) [5] measures the difference between these two images based on the Euclidean distance via pixel-by-pixel comparison, which takes all the available spectral channels into account for highlighting the changes [6]. In addition, principal component analysis (PCA) is also widely applied to make a difference image (DI) by selecting a few principal components for comparison [7], which has the ability to retain the key information as well as to suppress the noises contained in the images. However, this method is hinge on the statistical property of the image being processed. In addition, the iteratively reweighed multivariate alteration detection (IR-MAD) is also a traditional image analysis method [8]. In this method, different weights are assigned to different observed image data. The image data that show a smaller change are given a larger weight, and vice versa. Nevertheless, they do not take the links between bitemporal bands into account.

For the analysis of the DI, the thresholding and clustering methods are employed to segment the DI into two different classes, that is, changed and unchanged classes. These approaches can often acquire good performance in many tasks with high efficiency [9]–[12]. However, due to the increasingly improvement in the spatial resolution of the image, traditional change detection methods face many new challenges, which make them unable to meet the requirements in practice, because the individual pixel in the image is not spatial independently. Hence, it is necessary to develop new techniques for detecting the changes that occurred on the ground in high-resolution multispectral imagery.

In recent years, due to its powerful ability in representation learning, deep learning has become a hot research topic and gained great attention. Deep learning is such a technique that can learn the abstract and high-level feature representations from the input data by establishing a multilayer neural network. It is found to be effective in image processing [13], [14]. A variety of deep learning networks have achieved state-of-the-art performance in many tasks, such as deep belief networks [15], sparse autoencoders [16], and convolutional neural networks (CNNs) [17]. These neural network models usually need to be fed into plenty of labeled data for training, which makes them learn the key feature representations from the input data. Here, it should be noted that these features are extracted automatically without elaborately designing complex feature extractors by human engineers who are often rich in expert knowledge in some particular areas. When the network is trained well, it can be used to

address some intractable problems, such as image classification, image recognition, and change detection.

In the literature, the methods of change detection based on deep learning can be categorized in relation to the different levels. First is feature-level change detection. Owing to the fact that the deep neural network (DNN) has the strong ability in feature learning, it is easy for the DNN to extract the useful features from the original images. Then, some traditional methods can be used to segment the DI that is generated by comparing these features and obtain final change detection results [18], and so second is decision level change detection. This kind of method mainly includes three stages. First, some traditional change detection methods are used to process the bitemporal images in order to obtain the initial labels of image pixels, change or unchanged. After that, according to the preclassification result, some appropriate samples are selected for training the DNN so that it learns the concepts of changed and unchanged pixels. Finally, the original image data are fed into the network when it is trained well, and then, the final change detection result is obtained [19]–[21].

The decision-level change detection method can get better results compared to the traditional method with higher accuracy. However, it is difficult to obtain enough reliable labeled samples for training the network because it needs to spend much time and labor in marking the labels of remote sensing images. In this paper, we try to address this problem by adding unlabeled data for training the network. It has been verified that the use of unlabeled data can improve the performance of the multiclassifier [22]. We design a generative discriminatory classified network (GDCN) based on the generative adversarial networks (GANs) [23] for multispectral image change detection. Recently, GANs have attracted a great deal of attention from many researchers because of their advanced abilities for image generation models compared to traditional generation models [24]. GANs can generate realistic high-quality images, and there is no need to define any functional relationship between the input noises and real image data.

GANs have been widely used in image translation [25], [26] and classification tasks [27]. In particular, Gong *et al.* [28] aim to use GANs to obtain a better DI of bitemporal images, and then, it is processed by a clustering algorithm to get the change map (CM). The performance of this method can be affected by the clustering algorithm. Our method is largely different from it as: 1) a discriminatory classified network (DCN) is used to learn the concepts of changed and unchanged pixels; 2) we use labeled and unlabeled data obtained by preclassification, and generated data to compete for getting real labels by adversarial training; and 3) we propose a decision-level change detection method, and it is an end-to-end approach. In the proposed method, the GDCN is composed of a DCN and a generator. The DCN classifies the input data into changed class, unchanged class, or fake class (extra class). The generator can convert random noises into data that are similar to the real image data, providing additional training samples. This can promote the DCN to better learn the concepts of changed and unchanged pixels. When the network is trained well, the DCN can classify the raw image data into changed and unchanged class to obtain the final CM. While the GDCN requires training samples provided by preclassification,

preclassification does not require human intervention; overall, the proposed GDCN-based approach is a completely unsupervised approach. The performance analysis over real multispectral remote sensing image datasets with high resolution shows that the proposed method can accurately and effectively distinguish the changed pixels from the unchanged ones and is robust to noises.

The remainder of this paper is organized as follows. Section II introduces the relevant background of the proposed method. Section III explains the proposed multispectral image change detection method in detail. In Section IV, the effects of related parameters on the performance of the GDCN and the experimental results on four different real high-resolution multispectral remote sensing images datasets are analyzed. Section V discusses the influence of input noises, the DCN network structure, as well as training data on the change detection results. Finally, Section VI concludes this paper.

## II. BACKGROUND

In this section, the multispectral image change detection problem is described concisely, and the correlative theory about GANs is introduced.

### A. Problem Statement

The image change detection problem is intended to detect the changes between two coregistered images  $I_1$  and  $I_2$ , which share the same size and are obtained in the same area at the different times  $t_1$  and  $t_2$ , respectively. The widely used methods in image change detection are based on the DI analysis, which can be divided into three steps [29]: image preprocessing, generation of the DI, and analyzing the DI to obtain the CM. Among them, the most important part is the generation and analysis of the DI. In other words, image change detection is a process that classifies each pixel in the DI into two categories: changed and unchanged classes. This process can be formulated as

$$\text{CM} = F(I_1, I_2) = f(I_1 \Theta I_2) \quad (1)$$

where  $\Theta$  represents the operator that generates the DI, and  $f$  analyzes the DI to obtain the CM. Many traditional image change detection methods follow this formula, such as CVA-based and PCA-based approaches.

However, traditional image change detection methods have many problems in terms of effectiveness and accuracy. In particular, multispectral remote sensing images have improved image spatial resolution and many spectral channels that make it difficult to accurately detect changed areas of interest in the specific application. Therefore, the methods based on the DNN are proposed, and they have good performance in change detection tasks. But for the change detection method based on deep learning, it is hard to obtain many reliable training samples for training neural networks, especially for remote sensing images. Thus, in the proposed approach, we try to deal with this issue by adding unlabeled and generated data. Before giving a detailed description of the proposed GDCN-based approach, the theory of GANs is introduced.

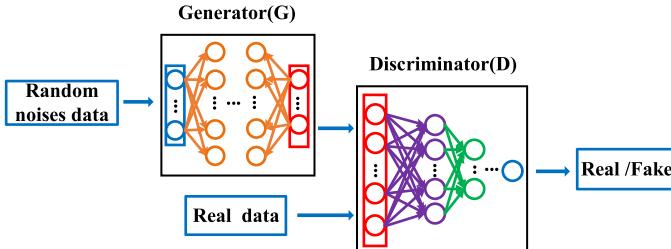


Fig. 1. Structure of the GANs.

### B. Generative Adversarial Networks

In 2014, Goodfellow *et al.* presented a new framework designed for training deep generative models, i.e., the GANs. The basic network structure of GANs is shown in Fig. 1. GANs are composed of a generator (G) and a discriminator (D). In the initial GANs, G and D are defined as multilayer perceptron (MLP). After that, many researchers have explored the new network structure of GANs and have achieved great success in many applications [30], [31].

G generates the  $G(z)$  according to the given noises distribution  $p_z(z)$ , and its goal is to recover the distribution of real data so that it can generate fake data that are close to the real data. D aims to accurately distinguish the real data and the fake data generated by G. In the training phase, train D in order to maximize the probability that both the real data and the data generated by G are set to the correct label, while train G so as to generate data that are very similar to the real data so that it cannot be distinguished by D. In the process, G and D can be seen as two competitors in the minimax game that follows this formulation:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log(D(x))] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))] \quad (2)$$

where  $x$  represents real data,  $z$  are the additional noises, and  $\mathbb{E}$  represents the empirical estimate of the probability expectation. By continuous adversarial learning, the ability of D to identify real data and fake data gradually increases; at the same time, the ability of G to generate data becomes stronger, and the generated data are increasingly closer to the real data. Until D cannot distinguish the real and generated data, the learning process continues [23]. Eventually, GANs can convert the distribution of noises into the distribution of real data. We can set a specific network structure for GANs according to the specific application, such as CNNs, rather than a fixed model [32]. Besides, a large number of functions can be used to the model; there is no need to infer, and only the backpropagation algorithm is applied to get the gradient for the learning [33].

### III. METHODOLOGY

In this section, we will elaborate the proposed change detection method for multispectral imagery. The overall framework is shown in Fig. 2. Given two registered multispectral remote sensing images that are acquired by different dates over the same area, we aim to highlight changes that have occurred between

them. We develop a GDCN to achieve this goal. First, we use CVA and Otsu [34] methods to deal with the images of different phases to obtain initial change detection result. According to initial change detection result, reliable labeled data and unlabeled data are selected to be fed into the DCN. The fake data generated by G are also input to the DCN. The DCN can learn the concepts of changed and unchanged pixels through adversarial training with these three types of data. When the network is trained well, it will be used to classify raw bitemporal image data into changed and unchanged classes. Thus, the final change detection result can be obtained.

### A. Training Sample Selection

First, for two registered multispectral remote sensing images  $X_1$  and  $X_2$ , which are acquired over the same area at different times  $t_1$  and  $t_2$ , CVA and Otsu methods are used to deal with them to obtain both labeled and unlabeled data. We use the CVA algorithm to obtain the DI. As a thresholding algorithm, Otsu is widely used in image change detection; it can automatically determine the threshold, with the advantages of easy implementation and good performance. Here, Otsu is used to classify the DI into two different classes, namely, changed and unchanged classes, thus obtaining the initial change detection result. Then, the neighborhood-based criterion is used to select labeled data [20]. According to the initial image pixel label, the pixel can be chosen as labeled data when the pixel  $p_{ij}$  satisfies the condition as follows:

$$\frac{Q(p_{ab} \in M_{ij} \wedge L_{ab} = L_{ij})}{n \times n} = 1 \quad (3)$$

where the pixel  $p_{ab}$  is in the neighborhood  $M_{ij}$  of  $p_{ij}$  and  $Q(p_{ab} \in M_{ij} \wedge L_{ab} = L_{ij})$  represents the number of pixels that the label is the same as  $L_{ij}$  in the neighborhood. On the contrary, the remaining data form unlabeled data.

G converts the input noises into data that match the real image data, i.e., fake data. Labeled data, unlabeled data, and fake data are used to train the GDCN. In addition, for the purpose of extracting robust local features and utilizing the spatial information of pixels, we use the neighborhood data of pixels as the input vector to input into the DCN [21].

### B. Network Establishment

In this section, we will detail the network structure of the GDCN. The GDCN consists of a G and a DCN. The G network structure is shown in Fig. 3; the dimension of the input noises can be set according to the concrete application, followed by a series of MLPs. We use the rectified linear unit (ReLU) as an activation function other than output. After the output of the ReLU, we use batch normalization (BN) to speed up network training [35]. The network structure is represented as “MLP + ReLu + BN” in Fig. 3. The Tanh [32], [36] as activation for output can speed up the convergence of the network and prevent the network from falling into local minimum. The network structure is represented as “MLP + Tanh” in Fig. 3. In the proposed GDCN, the network architecture of G is set to  $D - 50 - 80 - 100 - \omega \times \omega \times N$ , where  $D$  represents the

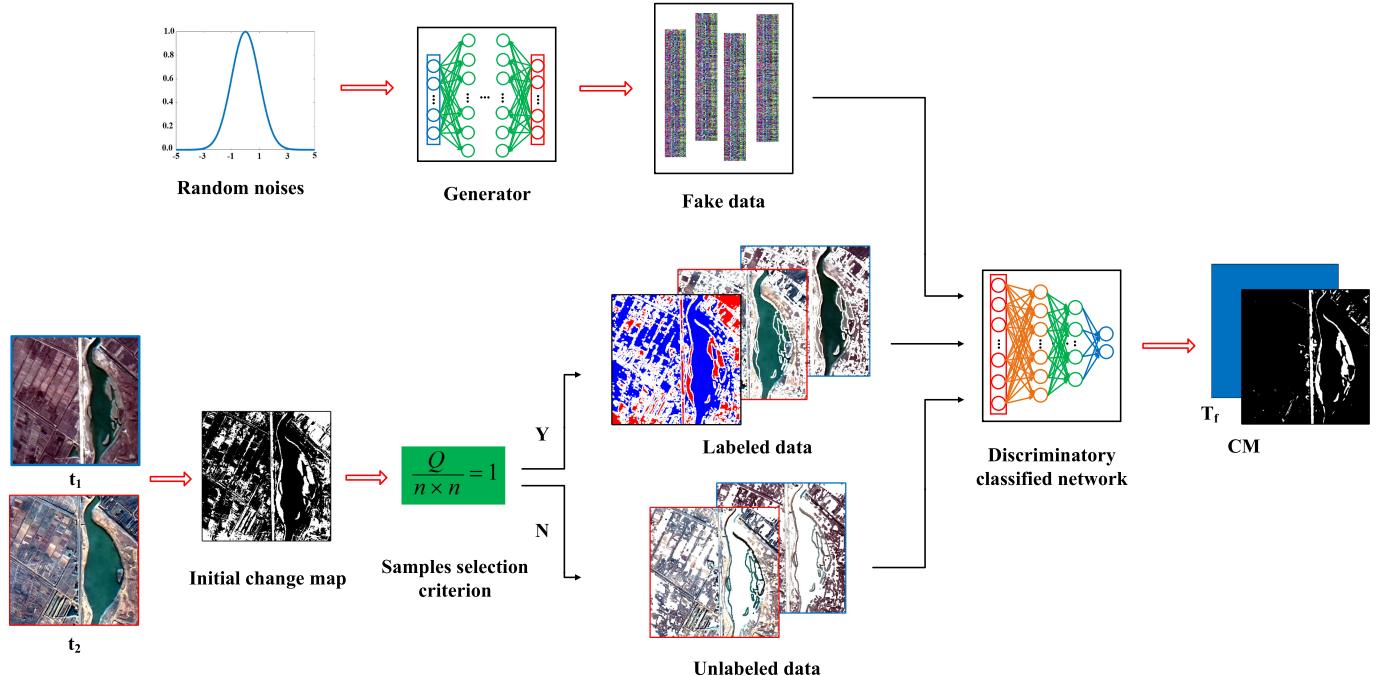


Fig. 2. Flowchart of the proposed GDCN-based method for multispectral image change detection. “Fake data” represent the pixel neighborhood data generated by G in a batch, and  $T_f$  represents fake class. First, initial change detection result obtained by preclassification and reliable labeled data and unlabeled data are selected according to the result. Then, labeled data, unlabeled data, and fake data generated by G are used to train the GDCN. Finally, bitemporal multispectral remote sensing images are fed into the DCN when it is trained well to get the final CM.

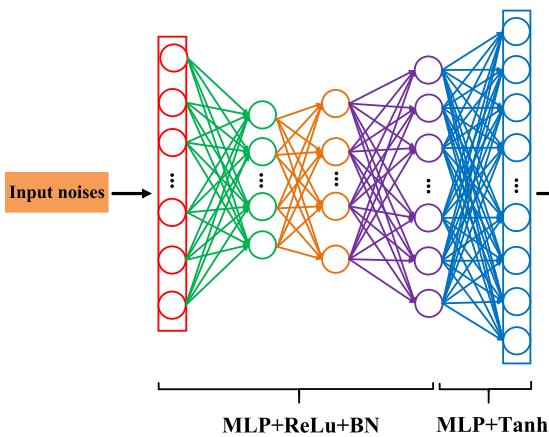


Fig. 3. G network structure of the GDCN.

noise dimensions,  $\omega$  represents the neighborhood size of the pixel, and  $N$  represents the number of image bands.

The network structure of “MLP + ReLu + BN” is also applied to the DCN; in addition, dropout is applied to it after BN so as to prevent the problem of overfitting during the training of the DNN [37]. The network structure is represented as “MLP + ReLu + BN + Dropout” in Fig. 4. Besides, the last layer of the DCN is the softmax layer, which is used to output the probability that input data belong to each class. It is represented as “MLP + Softmax” in Fig. 4. The network architecture of the DCN is set to  $\omega \times \omega \times N - 100 - 50 - 25 - 2$  in the proposed GDCN. It should be noted that we set the dimension of the output in the DCN as 2 in order to perform the task of change detection: one is used to determine whether the data generated by G are fake or not, and the other is used to denote whether

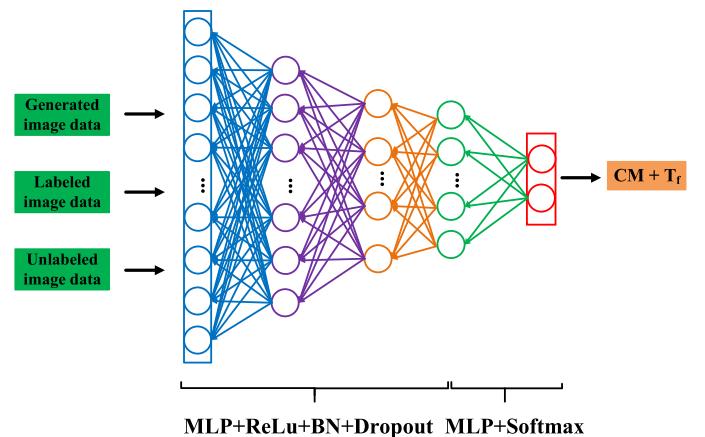


Fig. 4. DCN network structure of the GDCN.

the pixel has changed. This is different from typical GANs, whose discriminator is a binary classifier to discriminate true or fake input data, while our DCN performs a multiclass pixel classification task. By doing so, we can find that the DCN has the ability to accomplish two different tasks simultaneously, namely, discrimination and classification. This is mainly due to the following two reasons. On the one hand, it is easy to learn the approximate statistical distribution of the real image data, thus enhancing the robustness of the network to noises based on the discriminant analysis. On the other hand, it can learn the concepts of the changed and unchanged pixels by using sample data to train this network.

In the GDCN, the process of image pixel classification is completed by the DCN; G only plays a catalytic role. The more hidden layers of the DCN, the better the effect of learning the

concepts of changed and unchanged pixels. The effect of the DCN network structure on the change detection performance will be discussed in Section V.

### C. Network Learning

We will elaborate on the entire learning process of the GDCN in this section. First, the G loss is defined as follows:

$$\mathcal{L}_G = \min_G \mathbb{E}_{z \sim p_z(z)} [\log(1 - \text{DCN}(G(z)))] \quad (4)$$

where  $z$  are the additive random Gaussian noises, its distribution follows  $\mathcal{N}(0, 1)$ , and the influence of input noises on change detection results is discussed in Section V.  $z$  are input to G to generate fake data  $G(z)$ .  $G(z)$  are then input to the DCN to get  $\text{DCN}(G(z))$ . The purpose of  $\mathcal{L}_G$  is to reduce the probability that generated data belong to the fake class. G is trained to convert the input noises into data that are similar to the real image data, aiming to force the DCN to divide it into one of the real classes.

Besides, three types of data are fed into the DCN: labeled data, unlabeled data, and fake data. Accordingly, a pixelwise DCN loss (i.e.,  $\mathcal{L}_{\text{DCN}}$ ) is defined as follows:

$$\mathcal{L}_{\text{DCN}} = \max_{\text{DCN}} \lambda (\mathcal{L}_{\text{unlabeled}} + \mathcal{L}_{\text{fake}}) - \mathcal{L}_{\text{labeled}} \quad (5)$$

where

$$\mathcal{L}_{\text{unlabeled}} = \mathbb{E}_{x \sim p_x(x)} [\log(\text{DCN}(x))] \quad (6)$$

$$\mathcal{L}_{\text{fake}} = \mathbb{E}_{z \sim p_z(z)} [\log(1 - \text{DCN}(G(z)))] \quad (7)$$

$$\mathcal{L}_{\text{labeled}} = - \sum_i e_i \log(\tilde{e}_i) - \sum_i (1 - e_i) \log(1 - \tilde{e}_i). \quad (8)$$

The first term of  $\mathcal{L}_{\text{DCN}}$  is designed for unlabeled data that aim to reduce the probability that pixels belong to the fake class, where  $x$  are the unlabeled real image data and  $x$  are input to the DCN to get  $\text{DCN}(x)$ . The second term of  $\mathcal{L}_{\text{DCN}}$  is to prompt the DCN to distinguish between real image data and fake data and increase the probability that generated data belong to fake class.  $\lambda$  controls the weight of  $\mathcal{L}_{\text{unlabeled}} + \mathcal{L}_{\text{fake}}$  in (5). The last term is a cross-entropy function; it is used to process all the pixels in labeled data, aiming to classify them correctly into changed and unchanged classes, where  $e_i$  is the label of the labeled data  $i$  and  $\tilde{e}_i$  is the classification result in the DCN. The class that corresponds to the highest probability of softmax's output is the classification result.

Not only labeled data, but also unlabeled data are used to train the GDCN, which can improve the performance of the classifier. And this point has been verified in [38]–[40]. Although unlabeled data do not contain class label information, they belong to one of the changed and unchanged classes. By adding unlabeled data to the training set, it is beneficial to improve the diversity of the training data and make G generate the data much closer to the real image data. In addition, it is convenient to use unlabeled data to estimate the appropriate prior information used by the classifier. These further enhance the ability of the DCN in discrimination and classification and make it easier to learn the concepts of changed and unchanged pixels.

---

### Algorithm 1: Learning Procedure of the GDCN.

---

1. Initialization: randomly initializing the parameters of G and DCN.
  2. Updating network G: fixing network DCN, updating the parameters of network G by optimizing (4) with the minibatch stochastic gradient descent algorithm.
  3. Updating network DCN: fixing network G, updating the parameters of DCN by optimizing (5) with the minibatch stochastic gradient descent algorithm.
  4. Loop: loop from step 2 to step 3 until the value of (5) is convergent.
- 

The learning procedure of the GDCN is summarized in Algorithm 1. First of all, the parameters of G and DCN networks are randomly initialized, including the network weights and biases. Then, the minibatch stochastic gradient descent algorithm is used to train G and DCN by optimizing (4) and (5), respectively. The DCN is fixed, and G is trained by optimizing (4) with labeled data and unlabeled data. Followed by optimizing (5) to train the DCN with labeled data, unlabeled data, and fake data, at this point, G is fixed. In the process, the standard Adam optimizer with momentum is used to optimize (4) and (5). In our method, G and DCN are alternately trained like this until the objective function in (5) converges.

In the testing phase, we only use the DCN as the image change detection network. When the network is trained well, original bitemporal multispectral remote sensing images are fed into this network. The DCN softmax layer outputs the probability that each pixel belongs to each class, and the final change detection result is obtained according to it. It should be pointed out that although labeled and unlabeled data are obtained by preclassification, they do not require any human intervention. Therefore, the proposed change detection method is completely unsupervised on the whole process so that it meets the needs of practical applications.

## IV. EXPERIMENTAL STUDY

In order to verify the effectiveness of the proposed method, four real high-resolution multispectral remote sensing images datasets are selected to test the proposed framework based on the GDCN. Some representative algorithms are used as comparative methods, such as PCA based [7] and IR-MAD based [8]. In addition, the change detection method based on the DNN that consists of MLPs is also performed in these experiments. The network structure of the DNN is basically the same as the DCN, and the entire labeled data are used to train the DNN. The GAN-based change detection (GAND) [28] is also tested on these datasets. We choose the best combination in the GAND as a comparative method. We first introduce the datasets used in our experiments in brief. Then, some important parameters of the proposed method are analyzed, which will be used in our experiments. Finally, the corresponding assessment criteria are described in detail to demonstrate the performance of the proposed method.

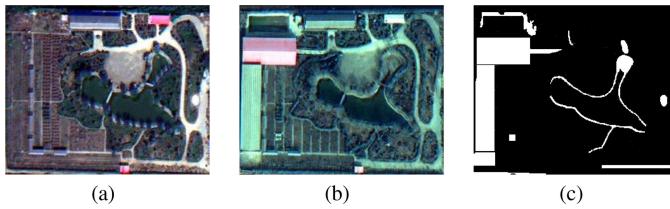


Fig. 5. Yandu Village dataset. (a) Image obtained on September 19th, 2012. (b) Image obtained on February 10th, 2015. (c) Reference image.

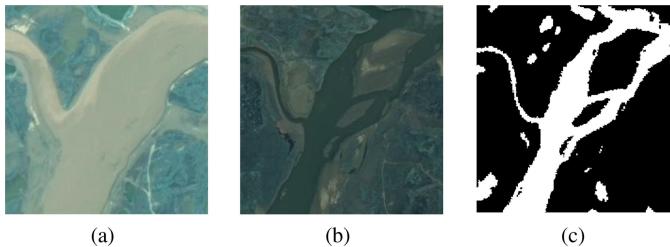


Fig. 6. Weihe River dataset. (a) Image obtained on August 19th, 2013. (b) Image obtained on August 29th, 2015. (c) Reference image.

#### A. Datasets

In this paper, four real high-resolution multispectral remote sensing image datasets are selected to assess the performance of the proposed method. Here, it is worth noting that each dataset consists of four different spectral bands, namely, Red, Green, Blue and Near infrared, which capture the information of the ground objects in different spectral ranges. The first dataset is the Yandu Village dataset, which is acquired by the WorldView-2 satellite over the southern area of Xi'an, China. These two images correspond to September 19, 2012 and February 10, 2015, respectively, as shown in Fig. 5(a) and (b). They have the same size  $322 \times 266$ , and their spatial resolution is 0.5 m. The reference image is marked by prior information and interpretation, as shown in Fig. 5(c).

The second dataset is the Weihe River dataset that consists of two multispectral remote sensing images obtained by the GF-1 satellite, which mainly records changes in the basin of Weihe River near Xi'an China. The panchromatic image (2 m) and the multispectral image (8 m) with four bands are fused to obtain these images. These two images correspond to August 19, 2013 and August 29th, 2015, respectively. They have same spatial resolution of 2 m and size  $321 \times 330$ , as shown in Fig. 6(a) and (b). Fig. 6(c) shows the reference image, which is obtained manually by using expert knowledge and prior information.

The third and fourth datasets are multispectral remote sensing images obtained by the GF-1 satellite with the same spatial resolution of 2 m, and they record the changes occurred on the estuary area of Yellow River in China. The former is the Minfeng dataset, which contains building changes information in Minfeng area. These images correspond to December 9, 2013 and October 16, 2015 with size  $651 \times 461$ , respectively, as shown in Fig. 7(a) and (b). The reference image is shown in Fig. 7(c). The latter is the Hongqi Canal dataset, which records the watercourse changes in Hongqi Canal. Fig. 8(a) and (b) shows the two images that correspond to December 9, 2013 and October

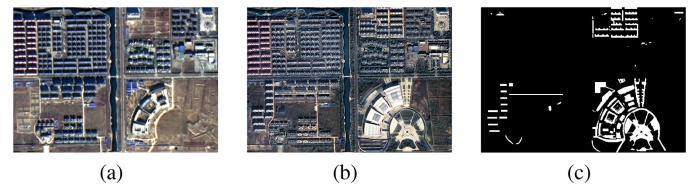


Fig. 7. Minfeng dataset. (a) Image obtained on December 9th, 2013. (b) Image obtained on October 16th, 2015. (c) Reference image.



Fig. 8. Hongqi Canal dataset. (a) Image obtained on December 9th, 2013. (b) Image obtained on October 16th, 2015. (c) Reference image.

16, 2015 with the same size  $539 \times 543$ , respectively. Fig. 8(c) represents the reference image obtained by manual annotation with expert knowledge.

#### B. Evaluation Criteria

Generally, the change detection result is often represented in the form of a binary image, wherein the white pixels denote the changed areas, while the black pixels indicate the unchanged regions. To assess the performance of the proposed method, some evaluation criteria are needed [41]. The false negative (FN) indicates the pixels that belong to change in the reference map and are erroneously classified into unchange in the CM. The false positive (FP) represents the pixels that belong to unchange in the reference image and are incorrectly classified into change in the CM. The true positive (TP) represents the pixels that belong to change in both the reference map and the CM. Conversely, the true negative (TN) represents the pixels that belong to unchange in both the reference map and the CM. Next, the overall error (OE) can be obtained by  $OE = FP + FN$ . The larger the value of the kappa coefficient (KC), the better the test result [42]. It can be calculated as  $KC = (OA - PRE)/(1 - PRE)$ , where OA is the percentage of the overall accuracy and  $OA = (TP + TN)/(TP + TN + FP + FN)$ ; the PRE represents the ratio of expected agreement, which can be calculated by  $PRE = ((TP + FP)(TP + FN)/(TP + TN + FP + FN)^2) + ((FN + TN)(FP + TN)/(TP + TN + FP + FN)^2)$ .

#### C. Parameter Settings

In this section, we analyze the influences of  $\lambda$ ,  $\alpha$ , and  $\omega$  on the performance of the proposed change detection method in order to obtain the optimal parameter settings.  $\lambda$  controls the weight of  $\mathcal{L}_{unlabel} + \mathcal{L}_{fake}$  in (5).  $\alpha$  represents the proportion of labeled data in the training set to all the labeled data. The  $\omega \times \omega$  represents the neighborhood size.

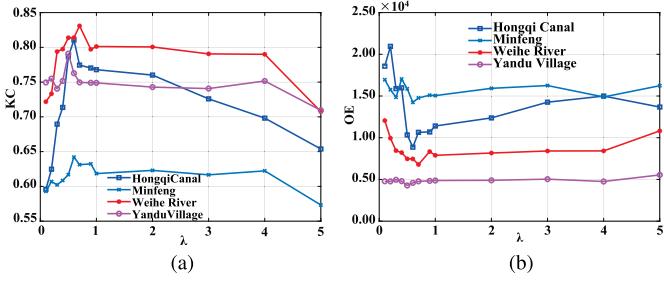


Fig. 9. Relationship between change detection results and  $\lambda$ . (a) Values of the KC for different  $\lambda$ . (b) Values of the OE for different  $\lambda$ .

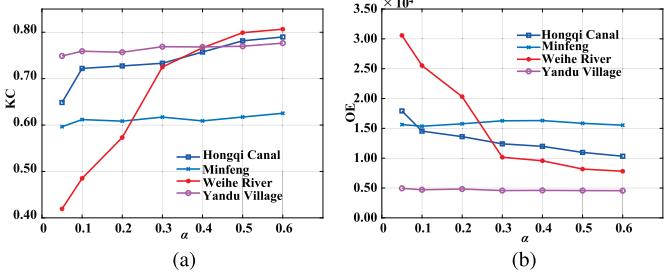


Fig. 10. Relationship between change detection results and  $\alpha$ . (a) Values of the KC for different  $\alpha$ . (b) Values of the OE for different  $\alpha$ .

1) *Effects of Parameter  $\lambda$ :* We test  $\lambda$  from 0 to 5 on different datasets with the GDCN. Fig. 9 shows the relationship between change detection results and  $\lambda$ . If  $\lambda$  is too small, the DCN mainly learns the concepts of changed and unchanged pixels from labeled data. The performance of the DCN will depend on the accuracy of labeled data. If  $\lambda$  is too large, the proportion of  $\mathcal{L}_{unlabel} + \mathcal{L}_{fake}$  in (5) is larger than  $-\mathcal{L}_{labeled}$ ; the DCN mainly distinguishes between real image data and generated data, and it cannot accurately learn the semantic information of the changed and unchanged pixels. Therefore,  $\lambda$  needs to be set to a reasonable value so that the DCN can accurately learn what is change and what is unchange from labeled data and unlabeled data. When  $\lambda$  is greater than 0.05 and less than 1, the KC first rises and then falls for these datasets. When  $\lambda$  is greater than 1, the KC presents an overall downward trend. As can be seen from Fig. 9, when  $\lambda$  is 0.6, 0.6, 0.7, and 0.5, the GDCN can get better results on the Hongqi Canal, Minfeng, Weihe River, and Yandu Village datasets. Therefore, it is appropriate that  $\lambda$  is set to 0.6 for different image datasets through the above analysis.

2) *Effects of Parameter  $\alpha$ :* In this section, some labeled data and all unlabeled data are used to train GDCN, and  $\alpha$  represents the ratio of labeled data in the training set to all labeled data. Fig. 10 shows the relationship between  $\alpha$  and change detection results obtained by the proposed method. It can be seen from Fig. 10 that the KC gradually increases and the OE gradually decreases with the increase of  $\alpha$ . The larger  $\alpha$  indicates the more labeled data in the training set, which allows the DCN to learn more semantic information of changed and unchanged pixels. It is beneficial for the DCN to classify bitemporal image data into changed and unchanged classes accurately. Moreover, the GDCN can get favorable results by using few labeled data and all unlabeled data in the Hongqi Canal, Minfeng, and Yandu Village

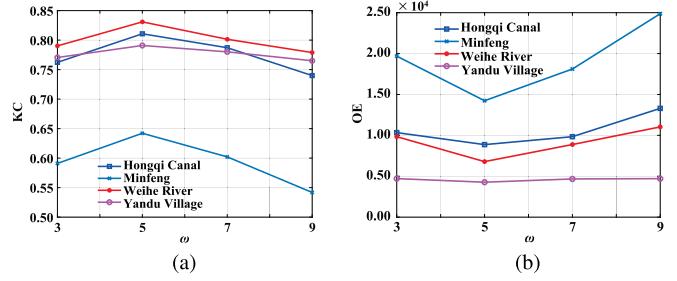


Fig. 11. Relationship between change detection results and  $\omega$ . (a) Values of the KC for different  $\omega$ . (b) Values of the OE for different  $\omega$ .

datasets. When  $\alpha$  is equal to 0.1, the KC values of the GDCN on the Hongqi Canal, Minfeng, and Yandu Village datasets are 0.7220, 0.6118, and 0.7591, respectively. This shows that the use of unlabeled data and generated data reduces the dependence of network change detection performance on labeled data in the proposed change detection method. It further proves the validity and reliability of the proposed method. The following experiments in real datasets are done by using all labeled and unlabeled data.

3) *Effects of Parameter  $\omega$ :* In section III, the neighborhood data of pixels as input vector to input into DCN. For each pixel location in bitemporal multispectral images, the neighborhood data is  $\omega \times \omega \times N$ ,  $\omega$  represents the neighborhood size of the pixel, and  $N$  represents the number of image bands. The relationship between  $\omega$  and change detection results obtained by the proposed method is shown in Fig. 11.  $\omega$  is set smaller and will make the neighborhood of the pixel contain less feature information, which is not conducive to learning the image intrinsic feature. The change detection results are not good when  $\omega$  is 3, as shown in Fig. 11. On the contrary, if  $\omega$  is set larger, the pixel neighborhood contains sufficient feature information, which can make the network fully learn the effective representation of the image data and better learn the concepts of changed and unchanged pixels, but it also contains some redundant information. From Fig. 11, we can see that a satisfactory change detection result can be obtained on the four real image datasets when  $\omega$  is set to 5.

#### D. Experiments on the Yandu Village Dataset

The first experiment is carried out on the Yandu Village dataset over the area of Xi'an, China. The CMs with the proposed method and the comparison methods are shown in Fig. 12. The CM obtained from the PCA contains many white noise points, as shown in Fig. 12(a). Besides, some unchanged regions are erroneously detected as changed areas, for example, the road near the right side in the CM. Fig. 12(b) shows the result of IR-MAD, which also contains many white noise points. The main changes can be detected by IR-MAD, but this method does not detect some small changed regions, such as the changed roads at the bottom of the image on the right side of the corner.

Furthermore, the result obtained by the DNN is shown in Fig. 12(c). The DNN has the strong learning ability, and it can learn the semantic information of changed and unchanged pixels so as to obtain better results. GAND can get a better DI

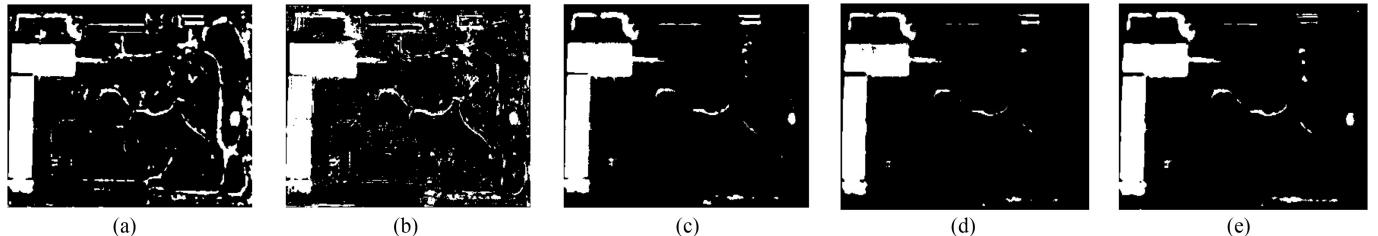


Fig. 12. Change detection results of the Yandu Village dataset by different methods. (a) PCA. (b) IR-MAD. (c) DNN. (d) GAND. (e) GDCN.

TABLE I  
EVALUATION COMPARISON OF EXPERIMENTAL RESULTS OBTAINED BY DIFFERENT METHODS ON THE YANDU VILLAGE DATASET

Methods	FP	FN	OE	OA	KC
PCA	5290	3132	8422	0.9017	0.6358
IR-MAD	4747	<b>2609</b>	7356	0.9141	0.6817
DNN	1242	3501	4743	0.9446	0.7693
GAND	338	4381	4719	0.9449	0.7538
<b>GDCN</b>	<b>911</b>	3360	<b>4271</b>	<b>0.9501</b>	<b>0.7909</b>

through GANs, so it gets better change detection result than the traditional method, as shown in Fig. 12(d). However, this method is affected by the clustering algorithm. Compared to the DNN, the GDCN is trained by adversarial training with labeled data and unlabeled data; it can better learn the concepts of changed and unchanged pixels; its CM is shown in Fig. 12(e). We can see that the GDCN obtains the best change detection result compared with other methods from the Table I, namely, the OA is 0.9501 and the KC is 0.7909. Thus, qualitative and quantitative analyses prove that the proposed change detection method is superior to other methods.

#### E. Experiments on the Weihe River Dataset

The Weihe River dataset mainly reflects the changes of the Weihe River valley at different times. The change detection results obtained by a variety of methods on the Weihe River dataset are shown in Fig. 13. The results of PCA and IR-MAD are shown in Fig. 13(a) and (b), respectively. We can see that some changed regions are detected by PCA and IR-MAD. However, many unchanged areas are mistakenly detected as changed areas and some changes in the area have not been detected in the CM produced by PCA and IR-MAD, respectively. Both PCA and IR-MAD are traditional pixel-based methods, they do not get accurate DI resulting in poor results. The results obtained by the DNN are better than PCA and IR-MAD, but the training samples contain much incorrect information that makes the DNN have poor performance, as shown in Fig. 13(c).

Although GAND can obtain a more accurate DI by adversarial training, the original DI contains too much error information that makes the DI obtained by GAND equally inaccurate, so the results obtained by GAND also contain a number of error messages, as shown in Fig. 13(d). For the proposed approach, the GDCN can learn exactly what has changed and unchanged based on the labeled data and unlabeled data, despite much

erroneous information contained in the labeled data. Therefore, the GDCN can detect the main changed areas with few noise points; the result is shown in Fig. 13(e). The specific evaluation criteria of different methods are shown in Table II. We can see that the GDCN gets the highest OA (i.e., 0.9359) and KC (i.e., 0.8309), and its performance is better than other methods.

#### F. Experiments on the Minfeng Dataset

This experiment is executed on the Minfeng dataset. The Minfeng dataset focuses on the changes in buildings during urbanization. The results obtained by traditional methods, including PCA and IR-MAD, are shown in Fig. 14(a) and (b), respectively. We can see that the PCA does not recognize the main changed areas, and the CM contains many white noise points. In contrast, the CM obtained by IR-MAD contains less white noise points, and it can detect most of the changed areas. However, it classifies the residential areas into changed areas incorrectly in the upper left corner of the CM.

Compared with traditional change detection methods, the DNN can learn the differences between the changed and unchanged pixels. However, the preclassification result of the Minfeng dataset contains much incorrect information, which leads to a trained network that cannot correctly learn the meaning of changed and unchanged pixels. So, the DNN has poor performance on the Minfeng dataset, as shown in Fig. 14(c). In addition, the DI obtained by GAND is more accurate than that by the traditional approach, and the better CM can be acquired by GAND, as shown in Fig. 14(d). Compared to the DNN, not only the labeled data, but also the unlabeled data are used to train the GDCN, which can reduce the impact of the labeled data on the change detection result and make the GDCN have better performance; its change detection result is shown in Fig. 14(e). Furthermore, detailed quantitative analysis of the evaluation criteria for these methods is listed in Table III. We can find that the GDCN outperforms other change detection methods with the highest OA (i.e., 0.9526) and KC (i.e., 0.6421).

#### G. Experiments on the Hongqi Canal Dataset

The final experiment is carried out on the Hongqi Canal dataset, which reflects the changes in the river and the land near the Xijiu Village. The CMs acquired by a variety of methods are shown in Fig. 15. Fig. 15(a) shows the result by PCA. PCA is able to detect major changes in the riverway, but it does not detect small changed areas, such as changes in the lower left corner. Besides, the CM contains some white noise points. The

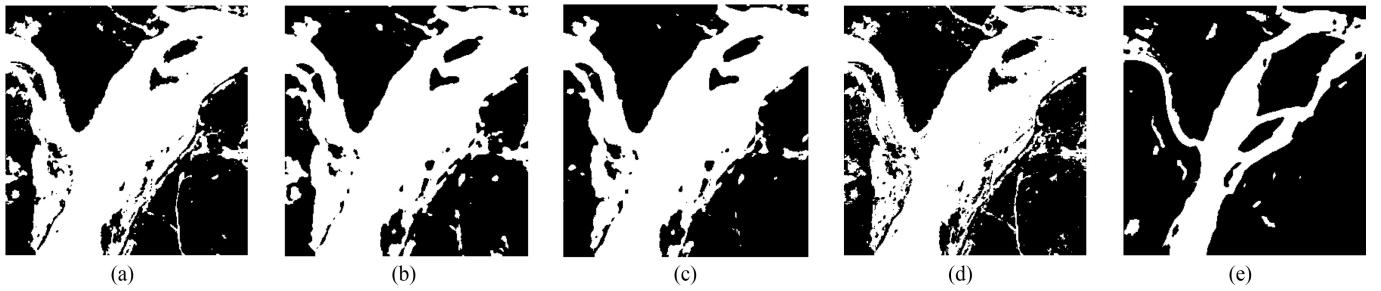


Fig. 13. Change detection results of the Weihe River dataset by different methods. (a) PCA. (b) IR-MAD. (c) DNN. (d) GAND. (e) GDCN.

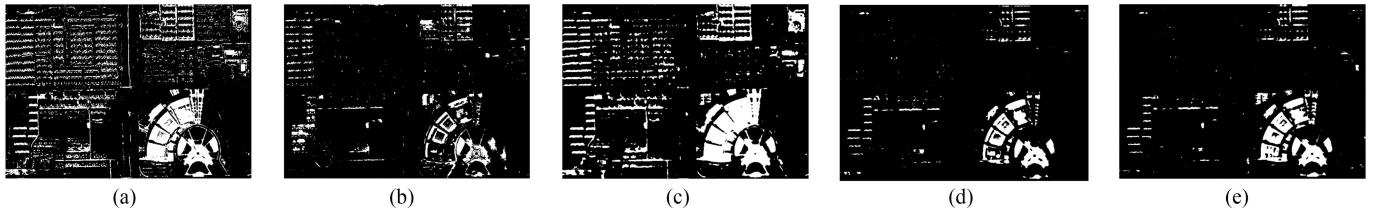


Fig. 14. Change detection results of the Minfeng dataset by different methods. (a) PCA. (b) IR-MAD. (c) DNN. (d) GAND. (e) GDCN.

TABLE II  
EVALUATION COMPARISON OF EXPERIMENTAL RESULTS OBTAINED BY  
DIFFERENT METHODS ON THE WEIHE RIVER DATASET

Methods	FP	FN	OE	OA	KC
PCA	30456	3063	33519	0.6836	0.3782
IR-MAD	31595	<b>2622</b>	34217	0.6770	0.3723
DNN	27108	3130	30238	0.7145	0.4215
GAND	29339	3047	32386	0.6943	0.3933
<b>GDCN</b>	<b>2772</b>	4017	<b>6789</b>	<b>0.9359</b>	<b>0.8309</b>

TABLE III  
EVALUATION COMPARISON OF EXPERIMENTAL RESULTS OBTAINED BY  
DIFFERENT METHODS ON THE MINFENG DATASET

Method	FP	FN	OE	OA	KC
PCA	36172	<b>4395</b>	40567	0.8648	0.3943
IR-MAD	7252	11476	18728	0.9376	0.5221
DNN	30971	4744	35715	0.8810	0.4531
GAND	5287	8728	<b>14015</b>	<b>0.9533</b>	0.6151
<b>GDCN</b>	<b>5100</b>	9134	14234	0.9526	<b>0.6421</b>

CM generated by IR-MAD is shown in Fig. 15(b). IR-MAD can detect the main changes in the region, and the CM has less noise points in comparison with PCA, but there are still some changes in the region are not detected, and some detailed information are lost.

In addition, the DNN can learn the notions of changed and unchanged pixels owing to the cogent feature learning capabilities. Compared with the traditional change detection methods, the DNN can obtain better change detection result with some noise points, as shown in Fig. 15(c). However, the performance of the DNN can be affected by the labeled data. Although GAND can detect the main changed areas due to the better DI obtained, GAND is still based on the DI approach, which has limitations

in accuracy; its CM is shown in Fig. 15(d). The training samples of the GDCN include all unlabeled data, which can reduce the impact of labeled data on the change detection result and improve the final detection accuracy. Thus, the GDCN can detect the main change area more accurately with less noise points, as shown in Fig. 15(e). From Table IV, we can see that the highest OA (i.e., 0.9697) and KC (i.e., 0.8107) can be obtained by the GDCN. The effectiveness and advantages of the proposed multispectral image change detection method are proved by the above analysis.

## V. DISCUSSION

In this section, we use Gaussian noises and uniform noises as inputs to G, respectively, to assess their effect on the change detection result. We discuss the effect of noise dimensions on the change detection result. Then, the influence of the DCN network structure on the performance of the GDCN can be discussed. In addition, the GDCN with different training data is performed on the Minfeng dataset to verify the effectiveness of the proposed method. Finally, the learning process of the DCN is discussed.

### A. Effects of Input Noises

G can transform Gaussian noises or uniform noises into a realistic generated image that has been verified in [30], [32], and [43]. However, multispectral remote sensing images have different properties than ordinary optical images, so we evaluate the change detection results on the Yandu Village and Hongqi Canal datasets by using Gaussian noises and uniform noises as the input of G, respectively. The distribution of Gaussian noises follows  $\mathcal{N}(0, 1)$ , and the distribution of uniform noises follows  $\mathcal{U}(-1, 1)$ . From Table V and Fig. 16, we can see that Gaussian noises as the input of G can get better results than uniform noises for these datasets. The noises in multispectral

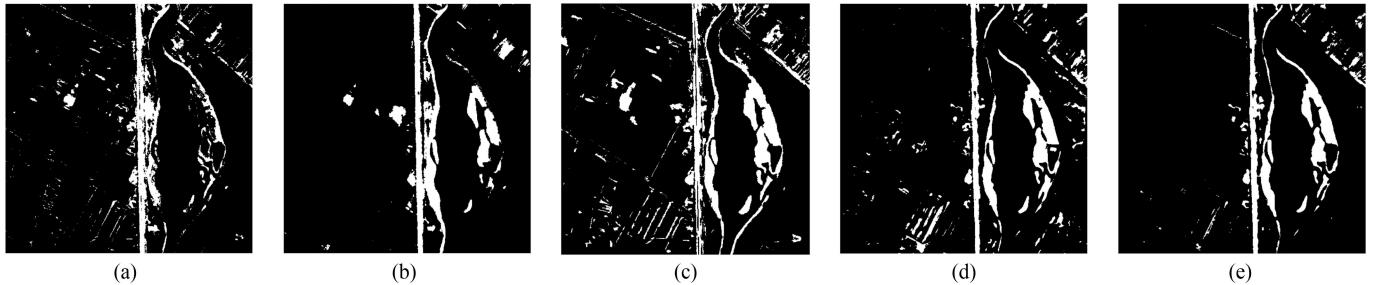


Fig. 15. Change detection results of the Hongqi Canal dataset by different methods. (a) PCA. (b) IR-MAD. (c) DNN. (d) GAND. (e) GDCN.

TABLE IV  
EVALUATION COMPARISON OF EXPERIMENTAL RESULTS OBTAINED BY  
DIFFERENT METHODS ON THE HONGQI CANAL DATASET

Method	FP	FN	OE	OA	KC
PCA	7906	10422	18328	0.9374	0.6092
IR-MAD	6501	8400	14901	0.9491	0.6856
DNN	7142	6546	13688	0.9532	0.7231
GAND	3236	12100	15336	0.9476	0.7252
GDCN	<b>2660</b>	<b>6208</b>	<b>8868</b>	<b>0.9697</b>	<b>0.8107</b>

TABLE V  
EVALUATION COMPARISON OF EXPERIMENTAL RESULTS OBTAINED BY  
DIFFERENT INPUT NOISES ON THE YANDU VILLAGE AND HONGQI  
CANAL DATASETS

Data sets	Noises	OE	OA	KC
Yandu Village	Gaussian	4271	0.9501	0.7909
	uniform	4519	0.9472	0.7688
Hongqi Canal	Gaussian	8868	0.9697	0.8107
	uniform	19504	0.9334	0.6629

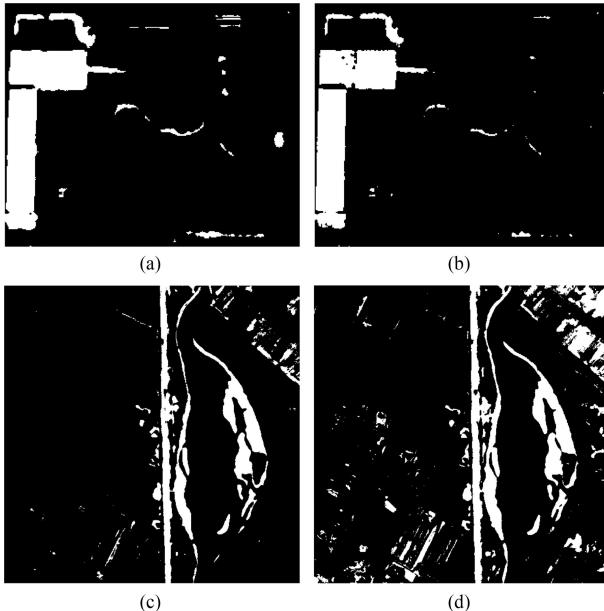


Fig. 16. CMs for Yandu Village and Hongqi Canal datasets with different input noises. (a) CM for the Yandu Village dataset with input Gaussian noises. (b) CM for the Yandu Village dataset with input uniform noises. (c) CM for the Hongqi Canal dataset with input Gaussian noises. (d) CM for the Hongqi Canal dataset with input uniform noises.

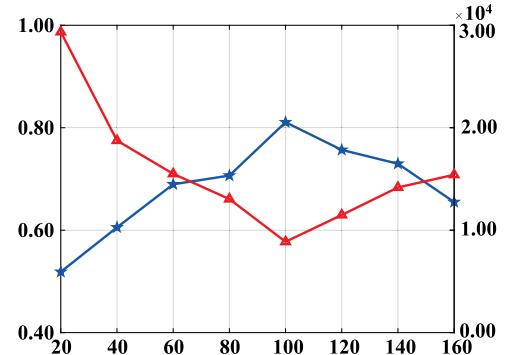


Fig. 17. Relationship between the change detection result and noise dimensions for the Hongqi Canal dataset. The blue line shows the values of the KC for different noise dimensions. The red line shows the values of the OE for different noise dimensions.

remote sensing images follow Gaussian distribution [44], and G can better convert Gaussian noises into image data that are similar to the real multispectral imagery data so as to improve the performance of the DCN. Therefore, we use Gaussian noises as input to G in the GDCN through the above analysis.

In addition, we assess the effect of noise dimensions  $D$  on the change detection result. Gaussian noises follow  $\mathcal{N}(0, 1)$  as the input noises to G. The size of target data is  $\omega \times \omega \times N$ , where  $\omega$  is 5. G maps input noises to target data. We test  $D$  from 20 to 160 on the Hongqi Canal dataset with the GDCN. Fig. 17 shows the relationship between the change detection result and noise dimensions. In [32], the dimension of input noise is much smaller than the size of the target image because the generated image contains much redundant information. In the proposed method, the target data of G also contain some redundant information. As can be seen from Fig. 17, the value of the KC is larger when  $D$  is 80–120. Therefore, we set  $D$  is 100 in the proposed to obtain better performance of the GDCN.

### B. Effects of the DCN Network Structure

Through the analysis in Section III, the network structure of the DCN has a major impact on the change detection performance compared with G. Therefore, we discuss the effect of the DCN network structure on the change detection performance in this section. For the Hongqi Canal dataset, G with three hidden layers is fixed; different DCN network structures are used for experiments to assess their impact on the change detection

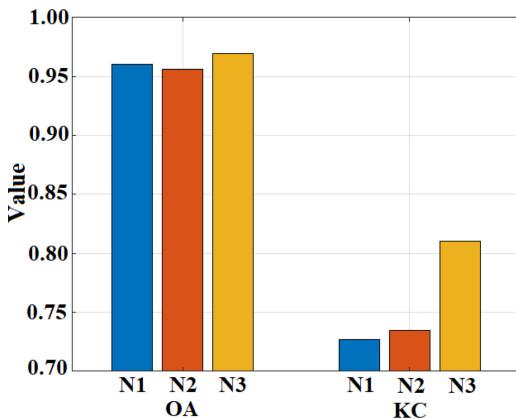


Fig. 18. Relationship between change detection results and different DCN network structures on the Hongqi Canal dataset. The values of OA and KC with different DCN network structures are shown. N1 represents the DCN with one hidden layer, N2 represents the DCN with two hidden layers, and N3 represents the DCN with three hidden layers.

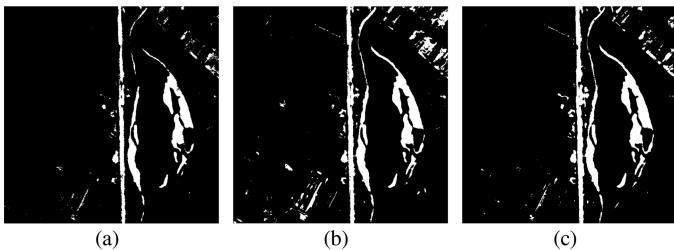


Fig. 19. CMs for the Hongqi Canal dataset obtained by different DCN network structures. (a) CM obtained by the DCN with one hidden layer. (b) CM obtained by the DCN with two hidden layers. (c) CM obtained by the DCN with three hidden layers.

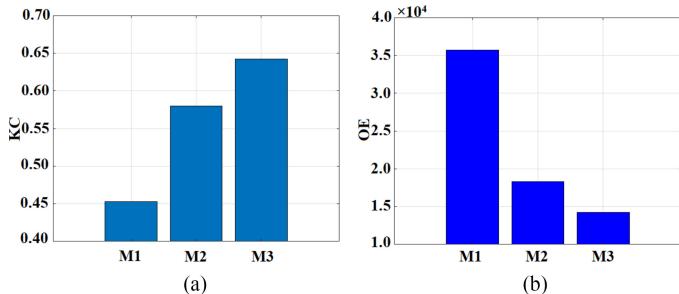


Fig. 20. Relationship between change detection results and different methods with different training data on the Minfeng dataset. M1 represents the DNN with all labeled data, M2 represents the GDCN with all labeled data (GDNN), and M3 represents the GDCN with all labeled and unlabeled data.

result. Fig. 18 shows the relationship between change detection results and different DCN network structures. It is obvious that the higher OA and KC are obtained when there are more hidden layers in the DCN. This is because deeper networks allow the DCN to learn more efficient features of the input data and thereby facilitate the DCN to better learn the concepts of changed and unchanged pixels. This enhances DCN's ability to classify pixels into changed and unchanged classes.

In addition, Fig. 19 demonstrates the CMs obtained by different DCN network structures. As can be seen from the CMs, the more accurate change detection result can be obtained when

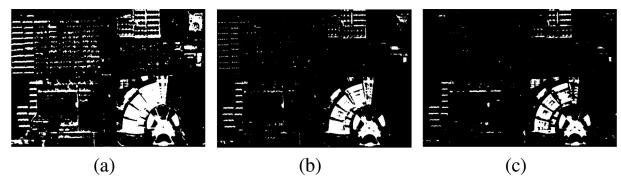


Fig. 21. CMs for the Minfeng dataset by different methods. (a) DNN. (b) GDNN. (c) GDCN.

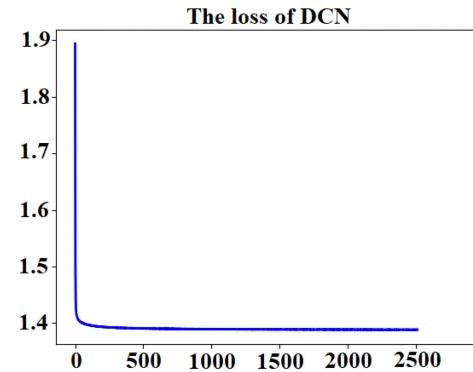


Fig. 22. Loss of the DCN corresponding to different epochs.

the network structure of the DCN is deeper. Although the neural network with multiple hidden layers can promote the change detection performance of the DCN, overfitting can occur when the number of network layers is too large. Therefore, the DCN network structure with three hidden layers is used in our experiments through the above analysis.

### C. Effects of Training Data

The effect of different training data on the change detection performance is discussed in this section. The DNN with all labeled data, the GDCN with all labeled data (GDNN), and the GDCN with all labeled and unlabeled data are used for the experiment on the Minfeng dataset. Fig. 20 shows the relationship between change detection results and different methods with different training data. Fig. 21 shows the CMs obtained by different methods. Although both the DNN and the GDNN are trained by all labeled data, G in the GDNN can force the DCN to better identify what are real labeled data, which can help the DCN to better learn the meaning of changed and unchanged pixels. Thus, we can see from Figs. 20 and 21 that the GDNN has better detection performance than that of DNN. The GDCN is trained with labeled and unlabeled data. The unlabeled data increase the diversity of training data. Compared with the GDNN, the GDCN can better distinguish the real images data, thereby enhancing the ability to learn the concepts of changed and unchanged pixels. Therefore, the GDCN with all labeled and unlabeled data can get the best change detection result compared with other methods.

### D. Learning Process

The experiments are run by Python 3.6 on AMAX workstation with Tesla K40c 12-GB GPU. Fig. 22 shows the loss values of the DCN for different epochs. From Fig. 22, we can see that at

the beginning of the training, the loss value of the DCN drops sharply, and then, the decrease amplitude gradually decreases. The loss value of the DCN gradually stabilizes, and the DCN converges when the epoch reaches 2000. Therefore, the DCN network that trains 2000 epochs is the final change detection network.

## VI. CONCLUSION

In this paper, we develop a change detection method for multispectral imagery based on the GDCN. In the proposed method, random noises are input to the generator for generating plausible synthetic data. Labeled data, unlabeled data, and fake data are used to train the proposed method. The performance of the DCN can be enhanced by adversarial training with these data. When the network is trained well, the raw bitemporal multispectral images are input into the DCN to get the final CM. We have demonstrated that this proposed method trained with unlabeled data and a limited number of labeled data outperforms some existing change detection methods, as well as its validity and reliability over several real multispectral remote sensing datasets with high resolution.

Furthermore, although labeled and unlabeled data are obtained by preclassification, they do not require any human intervention. Thus, the proposed multispectral image change detection method is completely unsupervised from the whole process.

## REFERENCES

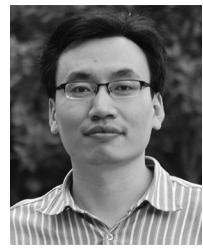
- [1] D. Lu, P. Mausel, E. Brondizio, and E. Moran, "Change detection techniques," *Int. J. Remote Sens.*, vol. 25, no. 12, pp. 2365–2401, 2004.
- [2] B. Gokaraju, A. C. Turlapati, D. A. Doss, R. L. King, and N. H. Younan, "Change detection analysis of tornado disaster using conditional copulas and data fusion for cost-effective disaster management," in *Proc. IEEE Appl. Imagery Pattern Recognit. Workshop*, Oct. 2015, pp. 1–8.
- [3] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: A systematic survey," *IEEE Trans. Image Process.*, vol. 14, no. 3, pp. 294–307, Mar. 2005.
- [4] Q. Liu, L. Liu, and Y. Wang, "Unsupervised change detection for multispectral remote sensing images using random walks," *Remote Sens.*, vol. 9, no. 5, 2017, Art. no. 438.
- [5] F. Bovolo and L. Bruzzone, "A theoretical framework for unsupervised change detection based on change vector analysis in the polar domain," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 1, pp. 218–236, Jan. 2007.
- [6] Q. Wang, P. Yan, Y. Yuan, and X. Li, "Multi-spectral saliency detection," *Pattern Recognit. Lett.*, vol. 34, no. 1, pp. 34–41, 2013.
- [7] J. Deng, K. Wang, Y. Deng, and G. Qi, "PCA-based land-use change detection and analysis using multitemporal and multisensor satellite data," *Int. J. Remote Sens.*, vol. 29, no. 16, pp. 4823–4838, 2008.
- [8] A. A. Nielsen, "The regularized iteratively reweighted mad method for change detection in multi-and hyperspectral data," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 463–478, Feb. 2007.
- [9] J. Kittler and J. Illingworth, "Minimum error thresholding," *Pattern Recognit.*, vol. 19, no. 1, pp. 41–47, 1986.
- [10] Z. Yetgin, "Unsupervised change detection of satellite images using local gradual descent," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 5, pp. 1919–1929, May 2012.
- [11] S. Krinidis and V. Chatzis, "A robust fuzzy local information c-means clustering algorithm," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1328–1337, May 2010.
- [12] M. Gong, Z. Zhou, and J. Ma, "Change detection in synthetic aperture radar images based on image fusion and fuzzy clustering," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2141–2151, Apr. 2012.
- [13] Y. Yuan, L. Mou, and X. Lu, "Scene recognition by manifold regularized deep learning architecture," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2222–2233, Oct. 2015.
- [14] Q. V. Le, "Building high-level features using large scale unsupervised learning," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2013, pp. 8595–8598.
- [15] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.
- [16] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, Apr. 2015.
- [17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [18] H. Zhang, M. Gong, P. Zhang, L. Su, and J. Shi, "Feature-level change detection using deep representation and feature change analysis for multispectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 11, pp. 1666–1670, Nov. 2016.
- [19] J. Liu, M. Gong, J. Zhao, H. Li, and L. Jiao, "Difference representation learning using stacked restricted Boltzmann machines for change detection in SAR images," *Soft Comput.*, vol. 20, no. 12, pp. 4645–4657, 2016.
- [20] M. Gong, J. Zhao, J. Liu, Q. Miao, and L. Jiao, "Change detection in synthetic aperture radar images based on deep neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 1, pp. 125–138, Jan. 2016.
- [21] M. Gong, T. Zhan, P. Zhang, and Q. Miao, "Superpixel-based difference representation learning for change detection in multispectral remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2658–2673, May 2017.
- [22] N. Souly, C. Spanipanato, and M. Shah, "Semi supervised semantic segmentation using generative adversarial network," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 5689–5697.
- [23] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
- [24] F. Huszár, "How (not) to train your generative model: Scheduled sampling, likelihood, adversary?" 2015, arXiv:1511.05101.
- [25] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5967–5976.
- [26] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," 2017, arXiv:1703.10593.
- [27] J. T. Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," 2015, arXiv:1511.06390.
- [28] M. Gong, X. Niu, P. Zhang, and Z. Li, "Generative adversarial networks for change detection in multispectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2310–2314, Dec. 2017.
- [29] L. Bruzzone and D. F. Prieto, "An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 452–466, Apr. 2002.
- [30] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, arXiv:1411.1784.
- [31] Z. Yi, H. Zhang, P. Tan, and M. Gong, "DualGAN: Unsupervised dual learning for image-to-image translation," in *Proc. IEEE Int. Conf. Comput. Vision*, Oct. 2017, pp. 2868–2876.
- [32] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015, arXiv:1511.06434.
- [33] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [34] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.
- [35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
- [36] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.
- [37] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [38] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 2234–2242.
- [39] N. Papernot, M. Abadi, Ú. Erlingsson, I. Goodfellow, and K. Talwar, "Semi-supervised knowledge transfer for deep learning from private training data," 2016, arXiv:1610.05755.

- [40] A. Odena, "Semi-supervised learning with generative adversarial networks," 2016, arXiv:1606.01583.
- [41] P. L. Rosin and E. Ioannidis, "Evaluation of global image thresholding for change detection," *Pattern Recognit. Lett.*, vol. 24, no. 14, pp. 2345–2356, 2003.
- [42] G. H. Rosenfield and K. Fitzpatrick-Lins, "A coefficient of agreement as a measure of thematic classification accuracy," *Photogrammetric Eng. Remote Sens.*, vol. 52, no. 2, pp. 223–227, 1986.
- [43] X. Wang and A. Gupta, "Generative image modeling using style and structure adversarial networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 318–335.
- [44] L. Liu, Y. Jiang, and C. Wang, "Noise analysis and image restoration for optical sparse aperture systems," in *Proc. Int. Workshop Educ. Technol. Training/Int. Workshop Geosci. Remote Sens.*, Dec. 2008, vol. 1, pp. 353–356.



**Tao Zhan** received the B.E. degree in electronic information engineering from the Henan University of Technology, Zhengzhou, China, in 2013. He is currently working toward the Ph.D. degree in pattern recognition and intelligent systems with the School of Electronic Engineering, Xidian University, Xi'an, China.

His research interests include computational intelligence and image understanding.



**Maoguo Gong** (M'07–SM'14) received the B.S. degree in electronic engineering and the Ph.D. degree in electronic science and technology from Xidian University, Xi'an, China, in 2003 and 2009, respectively.

Since 2006, he has been a Teacher with Xidian University, where he became an Associate Professor and a Full Professor, both with exception admission, in 2008 and 2010, respectively. His research interests include computational intelligence with applications to optimization, learning, data mining, and image understanding.

Dr. Gong received the prestigious National Program for the support of Top-Notch Young Professionals from the Central Organization Department of China, the Excellent Young Scientist Fund from the National Natural Science Foundation of China, and the New Century Excellent Talent in University from the Ministry of Education of China. He is the Vice Chair of the IEEE Computational Intelligence Society Task Force on Memetic Computing, an Executive Committee Member of the Chinese Association for Artificial Intelligence, and a senior member of the Chinese Computer Federation.



**Xudong Niu** received the B.E. degree in electronic engineering in 2016 from Xidian University, Xi'an, China, where he is currently working toward the M.E. degree in pattern recognition and intelligent systems with the School of Electronic Engineering.

His research interests include computational intelligence and image understanding.



**Yuelei Yang** received the B.E. degree in electronic information science and technology from Shandong Agricultural University, Tai'an, China, in 2016. He is currently working toward the M.E. degree in pattern recognition and intelligent systems with the School of Electronic Engineering, Xidian University, Xi'an, China.

His research interests include computational intelligence and image understanding.



**Shuwei Li** was born in 1997. He is working toward the B.S. degree in electronic science and technology with Southeast University, Nanjing, China.

He is an Analyst of the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an, China.