Generative Adversarial Networks and Conditional Random Fields for Hyperspectral Image Classification

Zilong Zhong, *Student Member, IEEE*, Jonathan Li, *Senior Member, IEEE*, David A. Clausi, *Senior Member, IEEE*, and Alexander Wong, *Senior Member, IEEE*

***Abstract*—In this paper, we address the hyperspectral image (HSI) classification task with a generative adversarial network and conditional random field (GAN-CRF)-based framework, which integrates a semisupervised deep learning and a probabilistic graphical model, and make three contributions. First, we design four types of convolutional and transposed convolutional layers that consider the characteristics of HSIs to help with extracting discriminative features from limited numbers of labeled HSI samples. Second, we construct semisupervised generative adversarial networks (GANs) to alleviate the shortage of training samples by adding labels to them and implicitly reconstructing real HSI data distribution through adversarial training. Third, we build dense conditional random fields (CRFs) on top of the random variables that are initialized to the softmax predictions of the trained GANs and are conditioned on HSIs to refine classification maps. This semisupervised framework leverages the merits of discriminative and generative models through a game-theoretical approach. Moreover, even though we used very small numbers of labeled training HSI samples from the two most challenging and extensively studied datasets, the experimental results demonstrated that spectral–spatial GAN-CRF (SS-GAN-CRF) models achieved top-ranking accuracy for semisupervised HSI classification.**

***Index Terms*—Conditional random fields (CRFs), generative adversarial networks (GANs), hyperspectral image (HSI) classification, semisupervised deep learning.**

I. Introduction

DUE TO their hundreds of spectral bands, the accurate interpretation of hyperspectral images (HSIs) has attracted significant scholarly attention from the machine learning and remote sensing communities [1]–[4]. Recent studies suggest that supervised deep learning models can alleviate challenges caused by the high spectral dimensionality of HSIs and achieve strikingly better classification accuracy [5]–[7]. However, there are still three challenges that prevent deep learning models from offering precise pixel-wise HSI classification maps [8], [9]. First, the high dimensionality of HSI pixels make it hard to directly use the deep learning models for normal optical images in HSI interpretation. Second, the shortage of labeled pixels limits the classification performance of deep learning models. Third, the classification maps generated by deep learning models tend to be noisy and have spurious object edges. In this proposal, I will analyze these challenges and offer our suggestions to mitigate them.

The first challenge derives from the high spectral dimensionality of HSIs, which comprise two spatial dimensions and one spectral dimension. Conventional methods focus on reducing the spectral dimensionality of HSIs. For example, Zhao and Du [10] adopted dimensionality reduction methods to extract discriminative features for the convolutional neural networks (CNNs) that follow to classify. Nonetheless, these methods ignore the inherent dimension reduction capability of deep learning models. Many papers indicate that both spectral and spatial features play important roles in precise HSI interpretation. For instance, Yuan and Tang [9] employed a shared structured learning strategy to construct a discriminant linear projection for spectral–spatial HSI classification. In addition, Chen *et al.* [5] proposed an end-to-end CNN model for HSI classification and achieved promising results, which show the generality of deep learning models. However, most machine learning models for HSI interpretation overlook the characteristics of this remotely sensed data.

The second challenge stems from the high cost of and difficulty in obtaining a large amount of labeled data for HSIs. Deep learning models [11]–[13] are prevailing for HSI classification. Many papers suggest that these models require a large amount of training data. For example, Chen *et al.* [5] provided a method adding noise to HSI pixels in order to increase the number of training samples. In addition, Li *et al.* [11] proposed a pixel-pair approach that samples two pixels independently and couples them as a group for the purpose of

Manuscript received May 9, 2018; revised February 23, 2019; accepted April 29, 2019. This work was supported in part by the Canada Research Chairs Program, in part by the Natural Sciences and Engineering Research Council of Canada, and in part by the China Scholarship Council. This paper was recommended by Associate Editor D. Goldgof. *(Corresponding author: Jonathan Li.)*

Z. Zhong is with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: z26zhong@uwaterloo.ca).

J. Li is with the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada, and also with the Fujian Key Laboratory of Sensing and Computing for Smart City, Xiamen University, Xiamen 361005, China (e-mail: junli@uwaterloo.ca).

D. A. Clausi and A. Wong are with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: dclausi@uwaterloo.ca; a28wong@uwaterloo.ca).

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/TCYB.2019.2915094

enlarging the training data size. The scarcity of training samples is especially the case for some land cover classes in the HSI datasets. However, in contrast to the conventional optical image classification objectives in the computer vision domain [14], [15], which usually contain hundreds or thousands of classes, the land cover classification objectives of HSIs have far fewer classes to recognize. Therefore, the assumption that deep learning models require a high amount of data for training may not hold for HSIs. On the other hand, a large amount of unlabeled data remains an unexploited gold mine for efficient data use. Several works that focus on semisupervised learning used small numbers of labeled and large numbers of unlabeled HSI samples for training. For instance, Mnih *et al.* [16] adopted multilayer neural networks to propagate labels from annotated HSI pixels to unannotated ones. Moreover, Chen *et al.* [12] applied a stacked convolutional autoencoder to use spectral–spatial representation learned from other HSI datasets. Although this paper achieved accurate classification results, these results may originate from the large area of spatial information contained in each training sample rather than deep learning models.

The third challenge is caused by the complexity of HSIs. Multiple works utilize the smoothness assumption that favors geometrically simple classification results [17]–[20]. For example, Tarabalka *et al.* [18] incorporated a probabilistic graphical model as the post-processing step to improve the classification outcomes of kernel support vector machines (SVMs). Zhong and Wang [21] constructed a conditional random field (CRF) with a high-order term to consider more complex relationships between different spectral bands and obtained very promising outcomes. In addition, Zhong *et al.* [19] incorporated a CRF for preprocessing as well as post-processing to stress the *a priori* smoothness and refine the classification maps. The adoption of probabilistic graphical models on top of supervised classification models can also be conceived as a way to take the unlabeled samples into account for HSI classification because this step does not require the ground truth annotation of neighboring pixels. However, most CRF-based models consider only the short-range correlations of pixels and ignore the long-range ones.

In the face of these difficulties, two common semisupervised learning methods—graph-based models and generative models—have been adopted to alleviate them [20], [22]–[24]. Graph-based models are premised on the smoothness assumption that accentuates geometrically simple classification results. For example, Yang *et al.* [20] imposed a manifold regularizer on a Laplacian SVM framework to learn spectral–spatial features for HSI image classification. In addition, Ji *et al.* [22] proposed a dual hypergraph framework that imposes spectral–spatial constraints by jointly calculating a Laplacian matrix. Although these graph-based semisupervised methods take both labeled and unlabeled samples into account, they identify HSI pixels based on hand-crafted features. Generally, these features learned from feature engineering steps are difficult to tune or generalize to other cases. Moreover, the performance of these semisupervised models largely depends on the quality of unlabeled data, which is hard to control or standardize. Recently, a generative model called generative adversarial network (GAN) [25] has attracted a lot of attention for image generation. For instance, Zhan *et al.* [23] proposed a semisupervised 1D-GAN for HSI classification, but ignored the spatial attribute of HSIs that can be used for enhancing classification performance. Moreover, Zhu *et al.* [24] used CNNs to build GANs for HSI classification and achieved very promising results. However, the discriminators used in this paper only use three principal component analysis (PCA) channels of HSIs and therefore do not fully exploit the spectral characteristic of HSIs.

Inspired by [25] and [26], we suggest a semisupervised deep learning framework that consists of a generator, discriminator, and CRF built on top of the discriminator. The discriminator and generator form a GAN based on game theory. Specifically, the discriminator adopts spectral–spatial convolutional layers to learn discriminative features from a small amount of labeled data and unlabeled data, and the generator employs spectral–spatial transposed convolutional layers to reconstruct HSI samples from vectors of Gaussian noise. Unlike traditional semisupervised models, which require a large amount of unlabeled data for training, our proposed framework is data-efficient because the generator creates a high amount of synthetic data and the discriminator takes a small number of unlabeled samples. In this way, the GAN and CRF (GAN-CRF) model estimates the real data distribution, mitigates the shortage of annotated data, and smooths the semisupervised learning process. In addition, the output of the discriminator is the unary input term of the subsequent CRF. The binary term of the CRF imposes an *a priori* smoothness whereby adjacent pixels are more likely to belong to the same categories. More importantly, the CRF takes on a fully connected form that imposes a random field on the whole classification map and considers the long-range relationship between HSI pixels. Thus, by taking a GAN and considering the continuity of neighboring pixels, the designed semisupervised architectures learn local fine-grain representation as well as high-level invariant features of HSI pixels concurrently.

The main contributions of this paper are as follows.

1) We integrate the spectral–spatial attributes of HSIs into convolutional and transposed convolutional layers of a GAN-CRF framework to learn discriminative spectral–spatial features of HSI samples.
2) We construct semisupervised GANs to alleviate the shortage of labeled data through adversarial training, which is a zero-sum game between the discriminators and generators of GANs.
3) We build dense CRFs that impose graph constraints on the softmax predictions of trained discriminators to refine HSI classification maps.

The overall structure of this paper takes the form of five sections. Section II reviews related works with regard to the GAN-CRF framework. Section III introduces fundamental layers, spectral–spatial discriminators and generators, semisupervised GANs, and post-processing CRFs of GAN-CRF models. Section IV offers model parameter settings, comparative experiments, and discussions. Finally, Section V makes some conclusions.

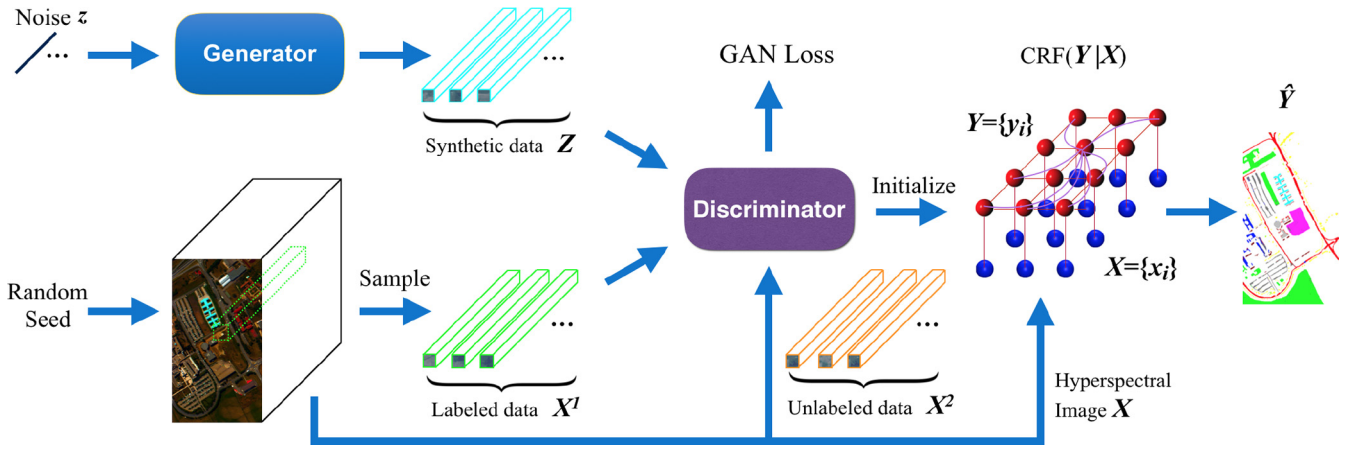


Fig. 1. Semisupervised GAN-CRF framework for HSI classification. First, in the semisupervised GAN, a generator transforms noise vectors z to a set of fake HSI cuboids $\boldsymbol{Z}$, and a discriminator tries to distinguish the categorical information as well as the genuineness of input cuboids that come from $\boldsymbol{X}^1$ or $\boldsymbol{Z}$. Then, a dense CRF is established by using the softmax prediction of the trained discriminator about $\boldsymbol{X}^2$ to initialize random variables $\boldsymbol{Y}$, which is conditioned on the HSI data $\boldsymbol{X}$. Mean-field approximation is adopted to offer a refined classification map $\hat{\boldsymbol{Y}}$ for the post-processing CRF.

II. Related Work

GANs are unsupervised deep learning models that provide a solution to implicitly estimate real data distribution and correspondingly generate synthetic samples. Recently, there has been increasing interest in GANs for unsupervised learning, especially in regards to generating synthetic images that approximate the distribution of real ones [25], [27]. Compared with traditional generative methods, GANs are not constrained by Markov fields or explicit approximation inference. For instance, a deep convolutional GAN [28] that consists of deep convolutional layers has been proposed to generate high-quality images. The original GAN aims for image generation and its variants have generated astonishing controllable and partially explainable images [29]. The GAN employs a discriminator and a generator to compete with each other. Specifically, the generator generates synthetic examples to deceive the discriminator, and the discriminator distinguishes real samples from fake ones. Since their objectives are contradictory, the training of the discriminator and generator of a GAN can be regarded as a process to find a Nash equilibrium through a game-theoretical point of view. Therefore, this GAN training can be formulated as a min–max optimization problem

$$\min_G \max_D \ \text{Loss}(D,G) = E_{\boldsymbol{x}\sim p_{\text{data}}}\left[\log D(\boldsymbol{x})\right] + E_{z\sim p_z}\left[\log\left(1-D(G(z))\right)\right] \quad (1)$$

where $D(\cdot)$ and $G(\cdot)$ represent the softmax outputs of a discriminator and synthetic data generated by a generator, respectively. x and z denote true images and vectors of Gaussian noise, and they follow the distributions of real HSI data and Gaussian noise, respectively. GANs produce very promising image generation results in datasets like the MNIST digit database [30] and the Yale Face database [31], both of which contain compact data distribution and similar image layout.

Graph models have been widely used for remotely sensed image interpretation tasks to effectively impose smoothness constraints on classification or segmentation results [26], [32]. CRFs are graphical models that assume *a priori* continuity whereby neighboring pixels of similar spectral signatures tend to have the same labels [21]. Since CRFs can be regarded as a structured generalization of multinomial logistic regression, the conditional probability distribution of a CRF takes this form

$$\text{Prob}(\boldsymbol{y}|\boldsymbol{x}) = \frac{\exp(-E(\boldsymbol{y}|\boldsymbol{x}))}{\sum_{\boldsymbol{y}}\exp(-E(\boldsymbol{y}|\boldsymbol{x}))} \quad (2)$$

where $\boldsymbol{y}$ and $\boldsymbol{x}$ denote output random variables and their corresponding observed data. $E(\cdot)$ is an energy function that models the joint probability distribution of $\boldsymbol{y}$ and $\boldsymbol{x}$. The optimal random variables can be calculated by the maximum *a posteriori* (MAP) estimation

$$\boldsymbol{y}^{\text{MAP}} = \underset{\boldsymbol{y}}{\text{argmax}}\ \text{Prob}(\boldsymbol{y}|\boldsymbol{x}). \quad (3)$$

However, although (16) usually is an intractable problem, it can be solved through approximation methods [33].

III. Proposed Model

To solve the three challenges of HSI classification, we propose a GAN-CRF-based semisupervised deep learning framework. Suppose an HSI $\boldsymbol{X}$ contains m pixels $\{\boldsymbol{x_i}\} \in \mathbb{R}^{n_x\times m}$, where n_x represents the number of spectral bands. Then, we sample two groups of HSI cuboids from $\boldsymbol{X}$: the labeled group $\boldsymbol{X}^1 = \{\boldsymbol{X_i}^1\} \in \mathbb{R}^{n_x\times w\times w\times m_l}$ and the unlabeled group $\boldsymbol{X}^2 = \{\boldsymbol{X_i}^2\} \in \mathbb{R}^{n_x\times w\times w\times m_u}$, where w, m_l, and m_u are the spatial width of HSI cuboids, the number of labeled, and the number of unlabeled HSI samples, respectively. Since each pixel in $\boldsymbol{X}$ corresponds to an HSI cuboid in $\{\boldsymbol{X_i}^1, \boldsymbol{X_i}^2\}$, therefore $m = m_l + m_u$. The labeled group $\boldsymbol{X}^1$ has its annotation $\boldsymbol{Y}^1 = \{\boldsymbol{y_i}^1\} \in \mathbb{R}^{(1+n_y)\times m_l}$, where n_y is the number of land cover classes and $\boldsymbol{y_i}^1[0]$ (the first entry in a vector $\boldsymbol{y_i}^1$) indicates whether the corresponding HSI cuboid is fake (1/0 means fake/real). As shown in Fig. 1, the whole model is composed of a discriminator, a generator, and a post-processing CRF. Since annotations $\boldsymbol{Y}^1$ of real HSI samples are used for training, the

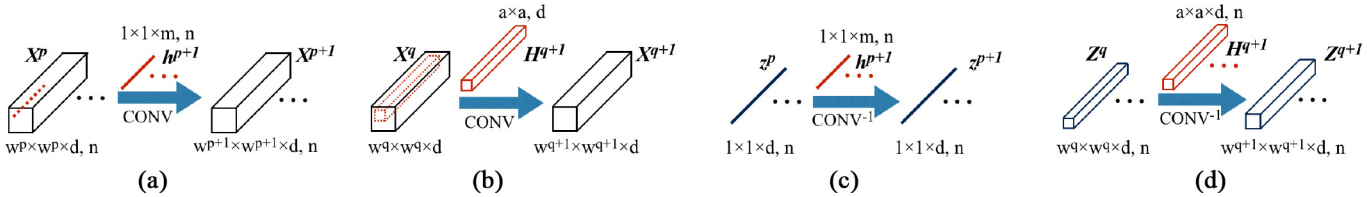


Fig. 2. Four basic convolutional and transposed convolutional layers aiming for hyperspectral features extraction and generation in semisupervised GAN-CRF models. (a) and (b) Spectral and spatial convolutional layers in discriminators. (c) and (d) Spectral and spatial transposed convolutional layers in generators.

discriminator and generator form a semisupervised GAN. The generator transforms noise vectors $\boldsymbol{z}$ to synthetic HSI cuboids $\boldsymbol{Z} = \{\boldsymbol{Z_i}\}$, each sample of which have the same size as those from $\boldsymbol{X}^2$. The discriminator attempts to distinguish real HSI cuboids $\boldsymbol{X}^1$ from fake ones $\boldsymbol{Z}$ and to classify real HSI cuboids.

In contrast to updating one discriminative model in supervised deep learning, the training of a GAN involves searching an equilibrium between the generator and discriminator by using stochastic gradient descent (SGD) or similar methods to optimize the parameters of the GAN. However, GANs are known for their instability in training, and it is almost impossible to find an optimal equilibrium between their generators and discriminators. Therefore, we adopt an alternating optimization strategy that successively updates the parameters of the generator and discriminator in each training iteration to help the discriminator to learn discriminative features using a small amount of labeled data and a large amount of synthetic data produced by the generator. When the training of a GAN is completed, we use the trained discriminator of the GAN to make a prediction about the unlabeled group $\boldsymbol{X}^2$. Then, a CRF is established by using the softmax predictions of the trained discriminator to initialize random variables $\boldsymbol{Y} = \{\boldsymbol{y_i}\} \in \mathbb{R}^{(1+n_y)\times m}$ that are conditioned on the raw HSI $\boldsymbol{X}$. Lastly, we use mean-field approximation to optimize the CRF and get a refined classification map $\hat{\boldsymbol{Y}}$.

A. Spectral–Spatial Discriminator and Generator

Discriminative deep learning models, such as CNNs and their extensions, have been used for HSI feature extraction and they have substantially outperformed traditional machine learning methods given enough training data [5], [6]. However, both these approaches ignore the inherent difference in spectral dimensionality between HSIs and common images used in computer vision tasks. Based on the assumption that the sampled HSI data form a low-dimensional manifold embedded in a higher-dimensional space, multiple models have tried to reduce the high dimensionality of HSI pixels and to learn more efficient representation [10], [34]. However, the dimension reduction process inevitably leads to the loss of useful information.

The specialty of HSI samples lies in its high spectral dimensionality. Recently, in response to this characteristic, Zhong *et al.* [6] implemented a spectral–spatial residual network (SSRN) that considers the characteristics of HSI by consecutively extracting spectral and spatial features and obtained the state-of-the-art supervised classification results. Therefore, as illustrated in Fig. 2(a) and (b), we extend the idea of spectral and spatial convolution from [6] to the discriminator of a GAN-CRF model. If $\boldsymbol{X}^{[p+1]}$ and $\boldsymbol{X}^{[q+1]}$ represent the feature tensors of $[p+1]$th spectral and $[q+1]$th spatial convolutional layers, then the spectral and spatial convolutional layers of a discriminator can be formulated as follows:

$$\boldsymbol{X}^{[p+1]} = \mathrm{LReLU}\left(\boldsymbol{w}^{[p+1]} * \boldsymbol{X}^{[p]} + b^{[p+1]}\right) \tag{4}$$

$$\boldsymbol{X}^{[q+1]} = \mathrm{LReLU}\left(\boldsymbol{W}^{[q+1]} * \boldsymbol{X}^{[q]} + b^{[q+1]}\right) \tag{5}$$

where $\boldsymbol{w}^{[p+1]}$ and $\boldsymbol{W}^{[q+1]}$ represent the $[p+1]$th spectral and $[q+1]$th spatial convolutional kernels, respectively. $b^{[p+1]}$ and $b^{[q+1]}$ are the biases of these two layers. $*$ denotes the convolutional operation. $\mathrm{LReLU}(\cdot)$ is a leaky rectified linear unit function

$$\mathrm{LReLU}(a) = \begin{cases} a, & \text{if } a > 0 \\ 0.2a, & \text{otherwise.} \end{cases} \tag{6}$$

In this paper, we use padding tricks to keep the spatial size of feature tensors in most convolutional layers unchanged. The goal of adopting spectral–spatial convolutional layers in a GAN-CRF model is to exploit as much information as possible from limited labeled HSI samples. Similarly, we stretch the spectral–spatial idea to transposed convolutional layers. As shown in Fig. 2(c) and (d), the spectral and spatial transposed convolutional layers of a generator can be formulated as follows:

$$\boldsymbol{z}^{[p+1]} = \mathrm{ReLU}\left(\boldsymbol{h}^{[p+1]} *^{\mathrm{T}} \boldsymbol{z}^{[p]} + b^{[p+1]}\right) \tag{7}$$

$$\boldsymbol{Z}^{[q+1]} = \mathrm{ReLU}\left(\boldsymbol{H}^{[q+1]} *^{\mathrm{T}} \boldsymbol{Z}^{[q]} + b^{[q+1]}\right) \tag{8}$$

where $\boldsymbol{h}^{[p+1]}$ and $\boldsymbol{H}^{[q+1]}$ represent the $[p+1]$th transposed spectral and $[q+1]$th transposed spatial convolutional kernels. $b^{[p+1]}$ and $b^{[q+1]}$ are the biases of these two layers. $*^{\mathrm{T}}$ denotes the transposed convolutional operation. $\mathrm{ReLU}(\cdot)$ is the rectified linear unit function

$$\mathrm{ReLU}(a) = \begin{cases} a, & \text{if } a > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

As shown in Fig. 2, in contrast to spatial convolutional layers, the transposed convolutional layers expand the spatial size of feature tensors. In both the discriminator and generator of a GAN-CRF model, we apply batch normalization [35] in all convolutional and transposed convolutional layers to stabilize the training of a GAN.

B. Semisupervised GAN

A GAN can be regarded as a combination of discriminative and generative models, where the discriminator focuses

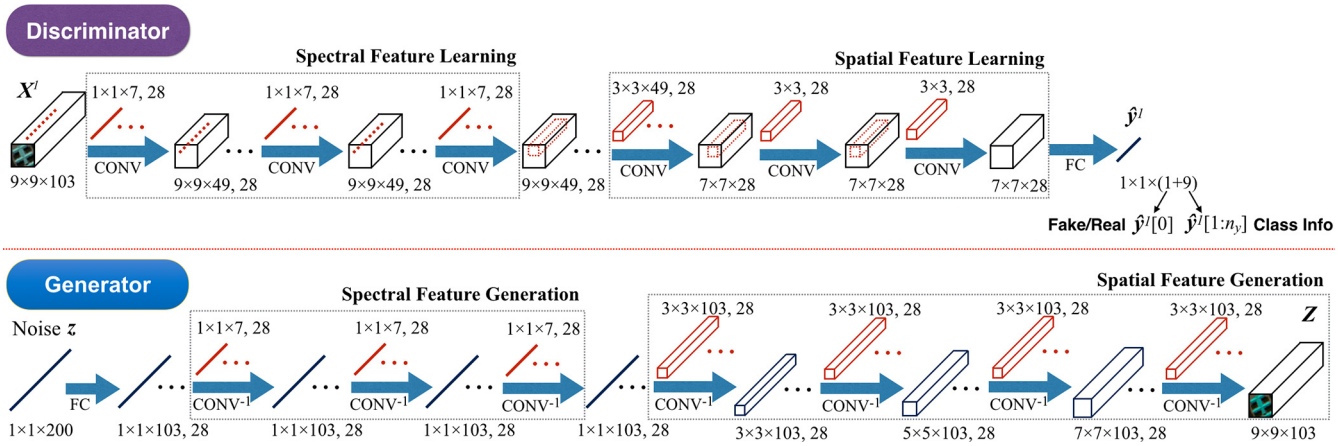


Fig. 3. Spectral–spatial discriminator (upper), which comprises consecutive spectral and spatial feature learning blocks, outputs a vector that contains an indicative entry of fake or real and categorical probabilities; and a spectral–spatial generator (lower), which comprises consecutive spectral and spatial feature generation blocks, transforms a vector of Gaussian noise to a synthetic HSI cuboid.

on learning discriminative features, and the generator concentrates on implicitly reconstructing real data distribution from random noises. As an example of University of Pavia (UP) dataset shown in Fig. 3, the discriminator comprises three spectral convolutional layers, three spatial convolutional layers, and a fully connected layer before a vector of softmax outputs. Conversely, the generator consists of a fully connected layer, three transposed spectral convolutional layers, and four spatial transposed convolutional layers to produce a synthetic hyperspectral cuboid.

As the generator of a GAN can produce reasonable synthetic images and utilize them to train the discriminator of the GAN, many research papers have extended the discriminator of GANs to semisupervised classification [23], [29], [36]. Similarly, we generalize the GAN to the semisupervised HSI classification task. Since the labeled hyperspectral cuboid group $\boldsymbol{X}^1 = \{\boldsymbol{X}_i^1\}$ has its corresponding annotation group $\boldsymbol{Y}^1 = \{\boldsymbol{y}_i^1\}$, the prediction of trained discriminators take this form

$$\hat{\boldsymbol{Y}}^1 = D\left(\boldsymbol{X}^1; \theta_D\right) \tag{10}$$

each element $\hat{\boldsymbol{y}}_i^1$ of which has $(1 + n_y)$ entries. Specifically, $\hat{\boldsymbol{y}}_i^1[0]$ indicates the genuineness of a hyperspectral cuboid, and $\hat{\boldsymbol{y}}_i^1[1 : n_y]$ is a vector of softmax outputs that shows the probabilities of a hyperspectral cuboid belonging to the n_y land cover classes. Compared to the original GAN that discriminates real data from fake ones, a semisupervised GAN recognizes the categorical information of HSI cuboids by adding a supervised term to the loss function of a GAN.

It is worth noting that the objectives of an unsupervised GAN and a semisupervised GAN are different and even partially contradictory. The unsupervised GAN aims for implicitly estimating the true data distribution. On the contrary, the semisupervised GAN focuses on data generation using limited labeled samples. Therefore, training a semisupervised GAN jeopardize its image generation capability. As presented in [36], a good semisupervised GAN requires a bad generator because this generator produces data outside real data distribution, which in turn helps the discriminator recognizes real data more accurately. In this way, the generator that produces synthetic HSI cuboids functions as a regularizer on the discriminator. Therefore, the loss function regarding optimize the discriminator of a GAN for semisupervised HSI classification takes the form

$$L_{\text{SEMI}}(\theta_D, \theta_G) = L_{\text{SUP}}(\theta_D) + L_{D1}(\theta_D) + L_{D2}(\theta_D, \theta_G) \tag{11}$$

where θ_D and θ_G are the parameters of a discriminator and a generator, respectively. L_{SEMI} is the total semisupervised loss for training the discriminator of a semisupervised GAN, L_{SUP}, L_{D1}, and L_{D2} represent the supervised loss of a discriminator, the unsupervised loss of a discriminator, and the unsupervised loss of a generator, respectively. These three terms are formulated as follows:

$$\begin{aligned} L_{\text{SUP}}(\theta_D) &= -E_{\boldsymbol{X}^1 \sim p_{\text{data}}} \log D\Big(\boldsymbol{X}^1; \theta_D\Big)\big[1 : n_y\big] \\ &= -E_{\boldsymbol{X}^1 \sim p_{\text{data}}} \log \hat{\boldsymbol{Y}}^1\big[1 : n_y\big] \end{aligned} \tag{12}$$

$$\begin{aligned} L_{D1}(\theta_D) &= -E_{\boldsymbol{X}^1 \sim p_{\text{data}}} \log\Big(1 - D\Big(\boldsymbol{X}^1; \theta_D\Big)[0]\Big) \\ &= -E_{\boldsymbol{X}^1 \sim p_{\text{data}}} \log\Big(1 - \hat{\boldsymbol{Y}}^{\boldsymbol{1}}[0]\Big) \end{aligned} \tag{13}$$

$$\begin{aligned} L_{D2}(\theta_D, \theta_G) &= -E_{z \sim p_z} \log D(G(\mathbf{z}; \theta_G); \theta_D)[0] \\ &= -E_{z \sim p_z} \log D(\boldsymbol{Z}; \theta_D)[0] \\ &= -E_{z \sim p_z} \log \hat{\boldsymbol{Y}}^1[0]. \end{aligned} \tag{14}$$

It is worth mentioning that $L_{D1} + L_{D2}$ also is the part of the total semisupervised loss L_{SEMI} that aims at training the bad generator of a GAN [36]. Correspondingly, the loss function for training the generator of a semisupervised GAN takes this form

$$\begin{aligned} L_G(\theta_D, \theta_G) &= -E_{z \sim p_z} \log\left(1 - D(G(\mathbf{z}; \theta_G); \theta_D)[0]\right) \\ &= -E_{z \sim p_z} \log\left(1 - D(\boldsymbol{Z}; \theta_D)[0]\right) \\ &= -E_{z \sim p_z} \log\Big(1 - \hat{\boldsymbol{Y}}^1[0]\Big). \end{aligned} \tag{15}$$

The training of a semisupervised GAN involves two alternating steps of SGD or similar optimization methods in each iteration. First, the gradients of a discriminator $-\nabla_{\theta_D} L_{\text{SEMI}}$ are used to update the parameters θ_D of a discriminator for

learning discriminative spectral–spatial HSI features. Second, the gradients of generators $-\nabla_{\theta_D} L_G$ are employed to update the parameters θ_G of a generator for improving the adversarial training of the semisupervised GAN.

C. GAN-CRF Model

CRFs have been widely used to post-process image segmentation results because they can exploit the predictions of large numbers of unlabeled pixels to enhance image interpretation performance [17], [37]. Once a semisupervised GAN has been built, we establish a CRF by using the softmax predictions of the trained semisupervised GAN about unlabeled HSI cuboids to initialize random variables $\boldsymbol{Y} = \{\boldsymbol{y}\}$ that are conditioned on observed raw HSI pixels $\boldsymbol{X}$. According to (2), the conditional probability distribution of this CRF takes the form

$$\text{Prob}(\boldsymbol{y}|\boldsymbol{X}) = \frac{\exp(-E(\boldsymbol{y}|\boldsymbol{X}))}{\sum_{\boldsymbol{y}} \exp(-E(\boldsymbol{y}|\boldsymbol{X}))}. \tag{16}$$

As illustrated in Fig. 1, given that high correlations exist between HSI pixels $\{\boldsymbol{x_i}\}$ in both short- and long-range, we adopt a dense CRF [26] that includes all pairwise connections between HSI pixels in the pairwise term of energy function to filter salt and pepper noises in homogeneous areas. The energy function of the dense CRF can be formulated as

$$E(\boldsymbol{Y}|\boldsymbol{X}) = U(\boldsymbol{Y}, \boldsymbol{X}) + P(\boldsymbol{Y}, \boldsymbol{X}) \tag{17}$$

where $U(\cdot)$ and $P(\cdot)$ are the unary and pairwise terms of the energy function that is used to build the dense CRF. Specifically, the unary term represents the information cost of pixel-wise softmax predictions $\{\boldsymbol{y_i}\}$ and the binary term penalizes the wrong labeling of pixel pairs $\{\boldsymbol{x_i}, \boldsymbol{x_j}\}$ with similar spectral signatures. These two terms are formulated as follows:

$$U(\boldsymbol{Y}, \boldsymbol{X}) = \sum_i U(\boldsymbol{y_i}, \boldsymbol{X_i}) = \sum_i D(X_i; \theta_D) \tag{18}$$

$$\begin{aligned} P(\boldsymbol{Y}, \boldsymbol{X}) &= \sum_{i,j} P\big(\boldsymbol{y_i}, \boldsymbol{y_j}, \boldsymbol{x_i}, \boldsymbol{x_j}\big) \\ &= \sum_{i,j} \mu\big(\boldsymbol{y_i}, \boldsymbol{y_j}\big) K\big(\boldsymbol{x_i}, \boldsymbol{x_j}, \boldsymbol{l_i}, \boldsymbol{l_j}\big) \end{aligned} \tag{19}$$

where $\boldsymbol{l_i}$ and $\boldsymbol{l_j}$ denote the locations of $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$, respectively. $\mu(\cdot)$ is a compatibility function, and $K(\cdot)$ is a bilateral Gaussian kernel function. These two functions take the forms

$$\mu\big(\boldsymbol{y_i}, \boldsymbol{y_j}\big) = \begin{cases} c, & \text{if } \eta(\boldsymbol{y_i}) \neq \eta\big(\boldsymbol{y_j}\big) \\ 0, & \text{otherwise} \end{cases} \tag{20}$$

$$K\big(\boldsymbol{x_i}, \boldsymbol{x_j}, \boldsymbol{l_i}, \boldsymbol{l_j}\big) = \exp\left(-\frac{\big(\boldsymbol{l_i} - \boldsymbol{l_j}\big)^2}{2\theta_\alpha^2} - \frac{\big(\boldsymbol{x_i} - \boldsymbol{x_j}\big)^2}{2\theta_\beta^2}\right) \tag{21}$$

where $\eta(\cdot)$ denotes a one-hot function. θ_α and θ_β are two standard deviations of the bilateral Gaussian function. c is a constant value that could be manually set. Random variables $\boldsymbol{Y} = \{\boldsymbol{y_i}\}$ of the established dense CRF is initialized to the softmax predictions of the trained discriminators $D(\boldsymbol{X}^2; \theta_D)$ of the semisupervised GAN according to (10).

In a GAN-CRF model, a GAN is utilized to produce softmax predictions about unlabeled HSI samples $\boldsymbol{X}^2$, and the post-processing CRF is independent of the GAN. Specifically, the predictions about a large numbers of unlabeled samples are used to initialize the unary term of the energy function that builds a dense CRF, and therefore the GAN-CRF model is more suitable in the case where only limited labeled samples are available. Because the energy function in (17) is an intractable problem, a function $Q(\boldsymbol{Y}|\boldsymbol{X})$ adopted to approximate the conditional probability distribution $\text{Prob}(\boldsymbol{Y}|\boldsymbol{X})$ of the CRF takes the form

$$Q(\boldsymbol{Y}|\boldsymbol{X}) = \prod_i Q(\boldsymbol{y_i}|\boldsymbol{X}) \approx \text{Prob}(\boldsymbol{Y}|\boldsymbol{X}) \tag{22}$$

in which the tractable function $Q(\boldsymbol{Y}|\boldsymbol{X})$ is close to $\text{Prob}(\boldsymbol{Y}|\boldsymbol{X})$ in terms of KL-distribution divergence. Then, the mean-field approximation [33] is used to find an optimal solution of random variables $\hat{\boldsymbol{Y}}$ for the established dense CRF.

IV. Result and Discussion

In this section, we introduce two challenging HSI datasets, set hyperparameters of semisupervised GANs, and evaluate GAN-CRF models and their competitors using performance metrics, including the classification accuracy of each land cover class, overall accuracy (OA), average accuracy (AA), and kappa coefficient (κ). In addition, we record training and testing times of all semisupervised GANs to quantitatively assess their computational complexity.

A. Experimental Datasets

Two most challenging and commonly studied HSI datasets—the Indian Pines (IN) and the UP—are used to evaluate the various types of semisupervised GANs and GAN-CRF models for HSI classification. In both datasets, we randomly selected {100, 150, 200, 250, 300} HSI cuboids with their annotations for training, and used the remaining cuboids for testing.

As shown in Fig. 7(a) and (b), the IN dataset contains 16 vegetation classes and has 145×145 pixels with a spatial resolution of 20 m by pixel. Two hundred hyperspectral bands are used for this paper and they range from 400 to 2500 nm. As illustrated in Fig. 8(a) and (b), the UP dataset includes nine urban land cover types and has 610×340 pixels with a spatial resolution of 1.3 m by pixel. One hundred and three hyperspectral bands are used for this paper and they range from 430 to 860 nm. The numbers of labeled HSI samples for each land cover class for the IN and UP datasets can be found in Figs. 7 and 8, respectively. Given their relatively small numbers, the labeled hyperspectral groups $\boldsymbol{X}^1$ used for training contain at least two samples for each land cover class to avoid the situation that no sampled HSI cuboids are sampled for rare classes, especially in the IN dataset.

B. Semisupervised GAN Setting

Fig. 3 takes the UP dataset as an example to show the discriminator and generator of a semisupervised GAN for HSI classification. In this semisupervised GAN, the generator takes a $1 \times 1 \times 200$ vector of Gaussian noise as the input and outputs a $9 \times 9 \times 103$ fake HSI cuboid aiming to make the discriminator classify it as real data. Concurrently, a real $9 \times 9 \times 103$

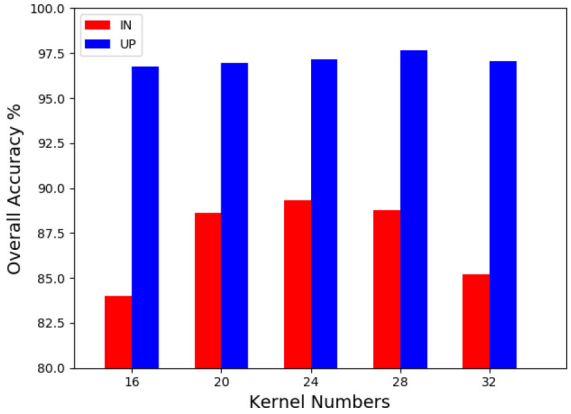


Fig. 4. Overall accuracies of semisupervised GANs with different kernel numbers in their convolutional and transposed convolutional layers using 300 labeled HSI samples for training.

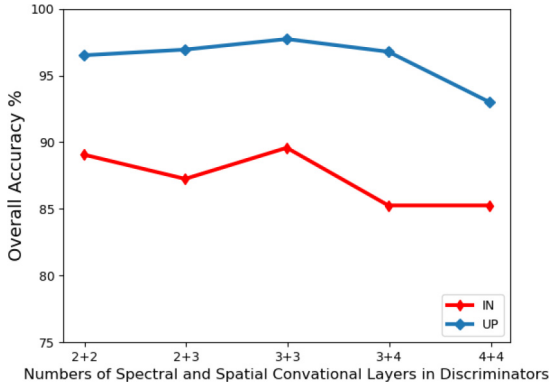


Fig. 5. Overall accuracies of semisupervised GANs that contain varying depths of spectral and spatial convolutional layers in their discriminators using 300 labeled HSI samples for training. The $x + y$ formation in the horizontal axis denotes a discriminator with x spectral and y spatial convolutional layers.

HSI cuboids is randomly sampled from a raw HSI as the input of the discriminator. In this paper, according to the result of a grid search, we set the learning rate to 0.0007, batch size to 50, and the spatial size of sampled HSI cuboids to 9×9. To avoid model collapse, we used Monte Carlo sampling [29] to marginalize noise during training. In addition, we adopted the Adam optimizer [38] to alternatingly train the discriminator and generator. After the hyperparameters of semisupervised GANs are configured, we analyzed three factors that influence the classification performance of semisupervised GANs.

First, the kernel number of convolutional and transposed convolutional layers affects the feature extraction and representation capacity of semisupervised GANs. As illustrated in Fig. 3, the discriminator and generator of a semisupervised GAN have the same kernel number in its convolutional and transposed convolutional layers. We tested different kernel numbers from 16 to 32 in an interval of 4 for all convolutional or transposed convolutional layers of semisupervised GANs. As shown in Fig. 4, the semisupervised GANs with 24 kernels in each layer achieved the highest classification accuracy using the IN dataset, and their counterparts with 28 kernels obtained the best classification performance using the UP dataset. These results are acquired in the 3000-epoch training for both datasets using randomly sampled 300 HSI cuboids.

Second, the depth of the spectral–spatial discriminators in semisupervised GANs also impacts their classification performance. Therefore, we assessed semisupervised GANs with from 4 to 8 layers, which includes spectral and spatial convolutional layers, with the same hyperparameter setting for each dataset. To make a fair comparison, we kept the generators of semisupervised GANs have the same architecture as the generator in Fig. 3. As demonstrated in Fig. 5, the semisupervised GANs with three spectral and three spatial convolutional layers obtained the highest overall accuracies in both datasets. The fact that classification performance of semisupervised GANs decreases with more convolutional layers than the optimal "$3 + 3$" architecture shows discriminators with deeper layers overfit the small number of labeled real HSI samples.

TABLE I
OVERALL ACCURACIES (%) OF SEMISUPERVISED GANS USING DIFFERENT NUMBERS OF UNLABELED AND 200 LABELED HSI SAMPLES IN THE IN AND UP DATASETS

Datasets	Models	0	1000	5000
IN	SPC-GAN	**63.21**	62.12	58.96
	SPA-GAN	**73.48**	71.28	67.62
	SS-GAN	81.12	**82.0**	78.0
UP	SPC-GAN	84.24	**84.69**	79.17
	SPA-GAN	91.01	**91.74**	87.35
	SS-GAN	**96.96**	95.76	93.90

Third, to evaluate the influence of unlabeled real HSI cuboids, we tested three types of semisupervised GANs using different numbers of unlabeled HSI samples for the IN and UP datasets. The three semisupervised GANs are the spectral GAN (SPC-GAN), and the spatial GAN (SPA-GAN), and the spectral-SPA-GAN (SS-GAN). As shown in Fig. 3, the SS-GAN has both spectral and spatial learning blocks in its discriminator, and the SPC-GAN and SPA-GAN contain only spectral and spatial blocks, respectively. Again, we used the same setting of generators for all semisupervised GANs as the generator in Fig. 3. Table I shows that adding real unlabeled HSI samples for training contributes little to and adding more unlabeled samples even jeopardizes the semisupervised HSI classification accuracy, which is caused by the different data distribution between labeled and unlabeled HSI samples.

C. Experimental Result

We compared the proposed semisupervised GANs to state-of-the-art GAN-based models, such as 1D-GAN [23], AE-GAN [12], and CNN-GAN [24]. To demonstrate the effectiveness of the spectral–spatial architecture, we also compared SS-GANs that comprise three spectral and three spatial convolutional layers with their variants: SPC-GANs (three spectral layers) and SPA-GANs (three spatial layers). As shown in Fig. 3, we recorded the HSI classification results of the spectral–spatial CNNs (SS-CNNs) as important benchmarks. We kept the generators of all GANs the same, which consist of

TABLE II
CLASSIFICATION RESULTS, TRAINING, AND TESTING TIMES OF DIFFERENT DEEP LEARNING MODELS USING 300 HSI SAMPLES FOR THE IN DATASET

Class	Samples	1D-GAN	AE-GAN	CNN-GAN	SS-CNN	SPC-GAN	SPA-GAN	SS-GAN
1	3	50.00	0	46.94	83.33	66.67	**100.0**	96.43
2	41	51.98	51.20	46.45	77.88	52.71	64.48	**87.29**
3	29	52.41	38.75	43.17	**81.48**	48.55	61.49	77.84
4	7	35.38	22.37	47.66	76.47	56.45	81.56	**92.35**
5	14	68.83	49.74	47.67	78.81	69.44	82.96	**92.64**
6	20	87.30	81.09	63.37	87.14	86.40	93.98	**95.05**
7	2	45.83	0	20.75	42.85	67.86	**82.35**	76.47
8	15	86.86	87.84	79.13	89.45	91.72	90.75	**98.70**
9	3	33.33	0	34.62	**100.0**	42.86	45.45	57.89
10	36	39.29	51.15	61.37	77.94	59.30	78.83	**90.11**
11	64	54.20	64.83	67.49	80.97	72.96	81.60	**95.19**
12	22	45.57	33.00	34.20	62.52	42.82	53.68	**85.74**
13	4	63.75	81.31	69.41	**97.50**	93.71	87.32	93.30
14	28	80.36	74.63	77.32	88.63	79.80	82.32	**92.59**
15	10	39.24	47.91	64.09	76.92	66.76	70.72	**78.74**
16	2	98.63	0	84.29	**100.0**	77.78	94.44	95.29
OA (%)		59.44	60.26	60.68	81.07	67.92	76.65	**90.28**
AA (%)		58.31	42.74	55.93	81.37	67.23	78.23	**87.85**
$\kappa \times 100$		52.06	54.24	55.03	78.21	63.25	73.30	**88.92**
Training (s)		153.85	217.70	64.87	139.55	932.23	233.32	803.23
Testing (s)		0.59	0.60	0.35	4.117	5.88	1.28	5.09

TABLE III
CLASSIFICATION RESULTS, TRAINING, AND TESTING TIMES OF DIFFERENT DEEP LEARNING MODELS USING 300 HSI SAMPLES FOR THE UP DATASET

Class	Samples	1D-GAN	AE-GAN	CNN-GAN	SS-CNN	SPC-GAN	SPA-GAN	SS-GAN
1	47	84.74	62.51	73.38	**96.07**	84.74	91.10	95.62
2	132	92.50	92.02	90.17	97.57	87.31	96.93	**99.49**
3	15	75.75	39.25	58.09	72.82	60.77	78.84	**89.02**
4	20	93.46	84.55	98.39	**99.37**	97.07	98.94	98.65
5	11	99.55	94.72	99.41	98.97	95.06	99.55	**100.0**
6	35	86.77	62.72	74.21	98.18	86.70	92.71	**99.09**
7	13	82.43	40.46	89.29	96.38	85.86	95.76	**97.10**
8	21	73.79	51.78	83.65	82.81	75.85	86.88	**92.54**
9	6	98.13	66.14	99.30	99.36	96.56	99.79	**100.0**
OA (%)		88.36	75.10	84.23	95.04	85.78	93.97	**97.61**
AA (%)		87.46	66.02	85.10	93.50	85.55	93.39	**96.84**
$\kappa \times 100$		84.41	67.07	78.79	93.40	80.69	91.98	**96.82**
Training (s)		107.27	145.11	64.71	93.45	647.68	159.37	527.46
Testing (s)		2.06	1.34	1.76	14.30	18.38	4.03	15.36

three spectral and four spatial transposed convolutions layers, each of which has 28 kernels. Then, we trained 3000 epochs for all GAN-based models, and set the input HSI cuboids with the same spatial size of 9×9 for all methods that use spatial convolutional layers, and tuned the competitors to their optimal settings.

Tables II and III report the classification performance, including accuracy of all land cover classes, OAs, AAs, and Kappa coefficients, of the IN and UP datasets, respectively. In most cases, the proposed semisupervised GANs perform better than the state-of-the-art GAN-based models. Interestingly, the supervised benchmark SS-CNNs perform slightly better than SPA-GANs, which shows the discriminative feature learning capacity of spectral and spatial convolutional layers. More importantly, the SS-GANs achieved the highest overall classification accuracies (90.28% and 97.61% OAs for the IN and UP datasets, respectively) among all GAN-based models and the SS-CNNs. It is worth noting that the semisupervised SS-GANs outperform fully supervised SS-CNNs in the IN and UP datasets with 9.21% and 2.57%, respectively, which shows that the generated samples are helpful for improving classification accuracy. These results demonstrate the effectiveness of spectral–spatial convolutional architectures and semisupervised adversarial training. In addition, Tables II and III also show the training and testing times of all models, which indicate the computational costs of these models. All experiments

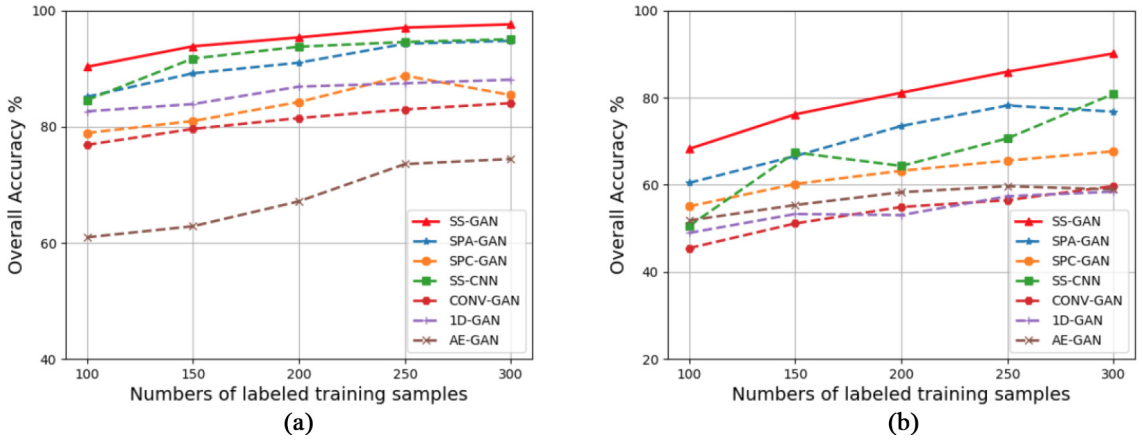


Fig. 6. Overall accuracies of different semisupervised GANs and the supervised benchmark SS-CNNs using from 100 to 300 HSI samples for training. (a) IN dataset. (b) UP dataset.

TABLE IV
OVERALL ACCURACIES (%) OF BASELINE CLASSIFICATION RESULTS (BASE.) AND DIFFERENT POST-PROCESSING METHODS FOR THE IN AND UP DATASETS

	Base.	Mean	Max	Gauss.	Laplace	CRF
IN	86.99	**96.05**	94.53	95.09	94.29	95.16
UP	96.41	96.65	96.47	96.74	96.84	**98.27**

TABLE V
OVERALL ACCURACIES (%) OF DEEP LEARNING MODELS AND THEIR REFINED RESULTS BY ADDING DENSE CRFS USING 300 LABELED HSI SAMPLES FOR TRAINING

	IN Dataset		UP Dataset	
Models	w/o CRF	w/ CRF	w/o CRF	w/ CRF
1D-GAN	59.44	70.41	88.36	94.41
AE-GAN	60.26	76.08	75.10	90.44
CNN-GAN	60.28	73.83	84.23	90.42
SS-CNN	81.07	87.66	95.04	98.05
SPC-GAN	68.92	74.64	85.78	88.13
SPA-GAN	76.65	85.64	93.97	97.57
SS-GAN	**90.28**	**96.30**	**97.61**	**99.31**

were conducted using an NVIDIA TITAN Xp graphical processing unit (GPU). In both datasets, the SPC-GANs are the slowest to train and the SS-GANs take about six times longer for training than SS-CNNs.

To test the robustness of the SS-GANs and their competitors, we randomly sampled different numbers of labeled HSI cuboids in an interval of 50 from 100 to 300 to train these semisupervised GANs and SS-CNNs for the IN and UP datasets. As shown in Fig. 6, the classification performance of SPA-GANs is comparable to that of SS-CNNs. AE-GANs perform clearly worse than other models because their fully connected layers fail to take the spectral–spatial characteristics of HSI samples into account. More importantly, the proposed SS-GANs consistently outperform their semisupervised competitors and SS-CNNs in both datasets. These results demonstrate the importance of accounting for the attributes of training data to design deep learning models, which is in line with the report of [6].

To evaluate the post-processing dense CRFs, we compared semisupervised GANs without CRFs (w/o CRF) with their counterparts with CRFs (w/ CRF). We set standard deviations in (21) $\theta_\alpha = 2$ and $\theta_\beta = 1$ in both datasets, and set constants in (20) $c = 8$ and $c = 10$ for the IN and UP datasets, respectively. Also, we compare the dense CRFs to other alternative post-processing methods, including mean filter, maximum filter, Gaussian filter, and Laplace method. Table IV shows that the CRF delivers comparable OA improvement for post-processing to the best performed mean filter using the IN dataset, and outperform all other methods using the UP dataset. This is caused by the homogeneous spatial layout of the former dataset and more heterogeneous distribution of the latter dataset. Therefore, the long-range correlation emphasized by CRFs facilitates the classification of HSI samples from heterogeneous areas.

In this paper, we used the three most prominent PCA channels of HSI $\boldsymbol{X}$ instead of raw HSI cuboids to facilitate the mean-field approximation of the dense CRF. As shown in Table V, SS-GANs and spectral–spatial GAN-CRF (SS-GAN-CRF) models perform better than their competitors, and GAN-CRF models significantly enhance the classification performance of those models without integrating dense CRFs. Moreover, Figs. 7 and 8 show the classification maps of all semisupervised GANs and all GAN-CRF models. The qualitative results of these classification maps are in line with the quantitative report of Table V. The SS-GAN-CRF models deliver the most accurate overall classification accuracies (96.30% and 99.31% OAs for the IN and UP datasets, respectively) and smoothest classification maps for both HSI datasets, because the SS-GANs learn the most discriminative spectral–spatial features and dense CRFs consider long-range correlations between similar HSI samples. Therefore, these classification outcomes validate the feasibility of integrating semisupervised deep learning and graph models given limited labeled HSI samples for training.

D. Discussion

There are three differences between the GAN-CRF framework and the original GAN proposed in [11]. First, GAN-CRF

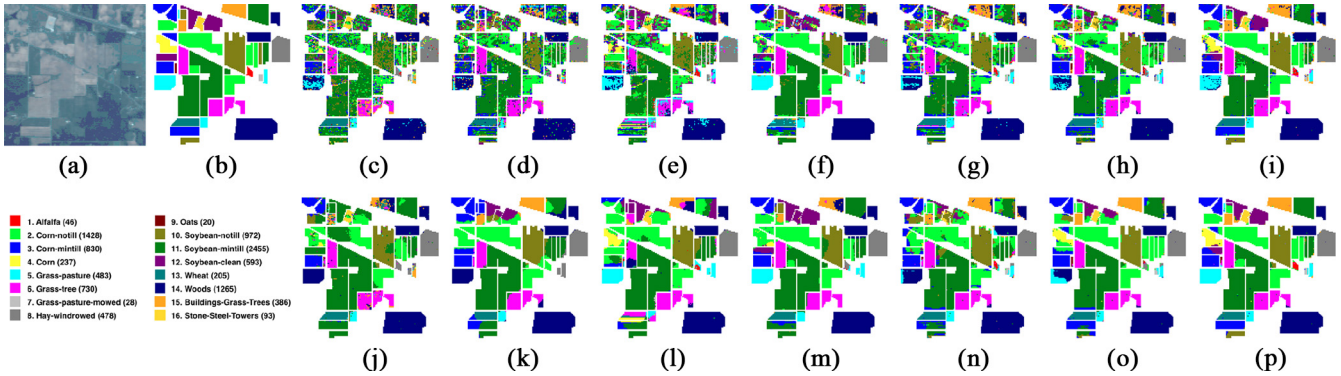


Fig. 7. Classification results of semisupervised GAN models, a supervised CNN, and their refined counterparts by adding dense CRFs using 300 labeled HSI samples for the IN dataset. (a) False color image. (b) Ground truth labels. (c)–(i) Classification maps of 1D-GAN, AE-GAN, CNN-GAN, SS-CNN, SPC-GAN, SPC-GAN, and SS-GAN. (j)–(p) Classification maps of 1D-GAN-CRF, AE-GAN-CRF, CNN-GAN-CRF, SS-CNN-CRF, SPC-GAN-CRF, SPA-GAN-CRF, and SS-GAN-CRF.

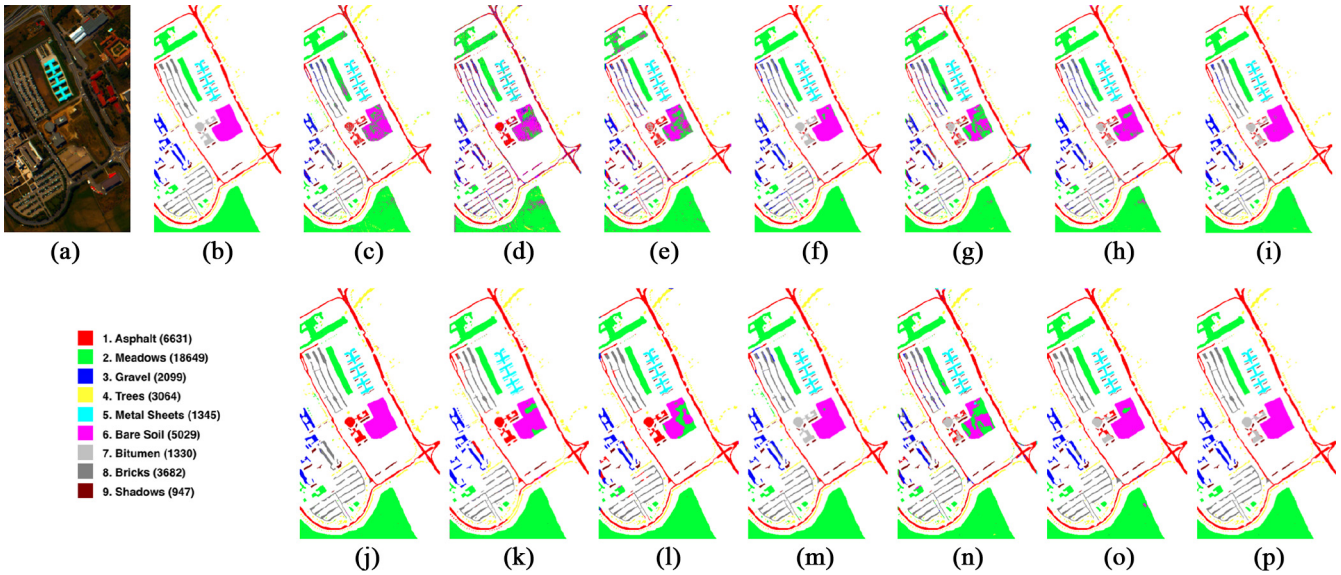


Fig. 8. Classification results of semisupervised GAN models, a supervised CNN, and their refined counterparts by adding dense CRFs using 300 labeled HSI samples for the UP dataset. (a) False color image. (b) Ground truth labels. (c)–(i) Classification maps of 1D-GAN, AE-GAN, CNN-GAN, SS-CNN, SPC-GAN, SPC-GAN, and SS-GAN. (j)–(p) Classification maps of 1D-GAN-CRF, AE-GAN-CRF, CNN-GAN-CRF, SS-CNN-CRF, SPC-GAN-CRF, SPA-GAN-CRF, and SS-GAN-CRF.

models take the spectral–spatial characteristics of HSI data into account for both the discriminators and generators. Second, the discriminators in the semisupervised framework extend the softmax predictions $\hat{\boldsymbol{y}}$ of a GAN from two classes (fake/real) to $1 + n_y$ classes, where n_y represents the number of land cover classes. Third, a post-processing dense CRF has been built on conditional random variables that are initialized to the softmax outputs of the trained GANs to filter salt and pepper noises in homogenous areas.

The GAN-CRF models incorporate the CRF as a post-processing step and build a graph upon the learned features and the softmax outputs of discriminators to refine HSI classification maps. Compared with those CRFs adopted in previous articles [39], [40], the fully connected CRFs consider the long-range correlations between HSI samples. This property helps GAN-CRF models to better filter noises in the homogeneous areas of some land cover classes. Compared to just a supervised discriminator, a GAN-CRF model integrates the advantages of deep learning models and probabilistic graph models and improves HSI classification accuracy. There are two main reasons for this improvement: 1) the synthetic HSI samples produced by generators help discriminators to learn more robust and discriminative features and 2) the subsequent dense CRFs consider the spectral similarity and spatial closeness of HSI samples to refine the softmax outputs conditional on these samples using the trained discriminators of GANs.

We gain four major insights from the semisupervised HSI classification outcomes of GANs and GAN-CRF models in both datasets. First, by taking the characteristics of training data into account, the discriminators of SS-GANs extract discriminative HSI features and achieve better classification accuracy. Second, generators of SS-GANs learn feature representation by producing synthetic HSI samples, and in turn make discriminators more robust to adversaries and learn more

discriminative features. Therefore, this adversarial training enables semisupervised GANs to deliver superior classification outcomes to supervised deep learning models. Third, adding unlabeled real HSI samples to train semisupervised GANs marginally improves or even jeopardizes the HSI classification results. Fourth, dense CRFs take the classification maps generated by semisupervised GANs as an initialization and smooth the noisy classification maps by adding a pairwise term that imposes the correlation between similar or neighboring pixels from input HSIs.

V. Conclusion

In this paper, we have proposed a semisupervised GAN-CRF framework to address three commonly occurring challenges for HSI classification: 1) the high spectral dimensionality of training data; 2) the small numbers of labeled samples; and 3) the noisy classification maps generated by deep learning models. First, we designed four consecutively structured convolutional and transposed convolutional layers to take the spectral–spatial characteristics of HSIs into consideration. Second, we established semisupervised GANs, each of which comprises a generator and a discriminator, to extract discriminative features and to learn feature representation of HSI samples. Third, we integrated a probabilistic graphical model with a semisupervised deep learning model to refine HSI classification maps. The experimental results using two of the most widely studied and challenging HSI datasets demonstrate that the SS-GANs perform the best among all semisupervised GAN-based models and supervised benchmark models, and subsequently that the SS-GAN-CRF models achieved state-of-the-art performance for semisupervised HSI classification.

The GAN-CRF models demonstrate an effective way to integrate two mainstream pixel-wise HSI classification methods—deep learning and probabilistic graphical models—and this framework can be easily generalized to other image interpretation cases. These two models have complementary advantages in the sense that deep learning models focus on discriminative feature extraction and implicit feature representation, and graph models emphasize the smoothness prior of images that is crucial for accurate classification and segmentation. However, the GAN-CRF framework presents a two-step setting because the dense CRFs function as a post-processing step to refine the classification maps generated by GANs.

The contributions of this paper mainly focus on validating the feasibility to integrate these two parts and show a way to implement this target. Therefore, a joint training framework is our future task, and current models need to be redesigned to achieve this goal. For example, we could make the discriminator of GAN a local semantic segmentation network and change the generator accordingly. The reason of the separated training lies in the different roles of semisupervised GAN and fully connected CRF. The GAN aims for training a discriminative model in a semisupervised way and then using the trained model to generate pixel-wise conditional probabilities. In contrast, the adoption of CRF considers the pixel-wise classification prediction holistically and adds structural constraints on top of it. Therefore, we will continue this paper line for imposing graph constraints on the convolutional layers of deep learning models to construct an end-to-end trainable framework.

Acknowledgment

The authors would like to thank N. Fladd for her helpful assistance in improving the quality of this paper and the anonymous reviewers for their valuable suggestions and comments. They would also like to thank NVIDIA for donating the TITAN Xp that was used for this paper through its GPU Grant Program.

References

[1] H. Li, G. Xiao, T. Xia, Y. Y. Tang, and L. Li, "Hyperspectral image classification using functional data analysis," *IEEE Trans. Cybern.*, vol. 44, no. 9, pp. 1544–1555, Sep. 2014.

[2] S. Jia, L. Shen, J. Zhu, and Q. Li, "A 3-D Gabor phase-based coding and matching framework for hyperspectral imagery classification," *IEEE Trans. Cybern.*, vol. 48, no. 4, pp. 1176–1188, Apr. 2018.

[3] Y. Yuan, J. Lin, and Q. Wang, "Hyperspectral image classification via multitask joint sparse representation and stepwise MRF optimization," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 2966–2977, Dec. 2016.

[4] Y. Zhou and Y. Wei, "Learning hierarchical spectral–spatial features for hyperspectral image classification," *IEEE Trans. Cybern.*, vol. 46, no. 7, pp. 1667–1678, Jul. 2016.

[5] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, Oct. 2016.

[6] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018.

[7] F. Luo, B. Du, L. Zhang, L. Zhang, and D. Tao, "Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2406–2419, Jul. 2019.

[8] Y. Tarabalka, J. Chanussot, and J. A. Benediktsson, "Segmentation and classification of hyperspectral images using minimum spanning forest grown from automatically selected markers," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 5, pp. 1267–1279, Oct. 2010.

[9] H. Yuan and Y. Y. Tang, "Spectral–spatial shared linear regression for hyperspectral image classification," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 934–945, Apr. 2017.

[10] W. Zhao and S. Du, "Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4544–4554, Aug. 2016.

[11] W. Li, G. Wu, F. Zhang, and Q. Du, "Hyperspectral image classification using deep pixel-pair features," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 844–853, Feb. 2017.

[12] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014.

[13] Y. Chen, X. Zhao, and X. Jia, "Spectral–spatial classification of hyperspectral data based on deep belief network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2381–2392, Jun. 2015.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1106–1114.

[15] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[16] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[17] X. Cao *et al.*, "Hyperspectral image classification with Markov random fields and a convolutional neural network," *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2354–2367, May 2018.

[18] Y. Tarabalka, M. Fauvel, J. Chanussot, and J. A. Benediktsson, "SVM- and MRF-based method for accurate classification of hyperspectral images," *IEEE Trans. Geosci. Remote Sens. Lett.*, vol. 7, no. 4, pp. 736–740, Oct. 2010.

[19] Y. Zhong, J. Zhao, and L. Zhang, "A hybrid object-oriented conditional random field classification framework for high spatial resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 7023–7037, Nov. 2014.

[20] L. Yang, S. Yang, P. Jin, and R. Zhang, "Semi-supervised hyperspectral image classification using spatio-spectral Laplacian support vector machine," *IEEE Trans. Geosci. Remote Sens. Lett.*, vol. 11, no. 3, pp. 651–655, Mar. 2014.

[21] P. Zhong and R. Wang, "Modeling and classifying hyperspectral imagery by CRFs with sparse higher order potentials," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 2, pp. 688–705, Feb. 2011.

[22] R. Ji *et al.*, "Spectral–spatial constraint hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 3, pp. 1811–1824, Mar. 2014.

[23] Y. Zhan, D. Hu, Y. Wang, and X. Yu, "Semisupervised hyperspectral image classification based on generative adversarial networks," *IEEE Trans. Geosci. Remote Sens. Lett.*, vol. 15, no. 2, pp. 212–216, Feb. 2018.

[24] L. Zhu, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Generative adversarial networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5046–5063, Sep. 2018.

[25] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[26] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[27] T. Salimans *et al.*, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2234–2242.

[28] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[29] Y. Saatci and A. G. Wilson, "Bayesian GAN," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3622–3631.

[30] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[31] J. Yang, D. Zhang, A. F. Frangi, and J.-Y. Yang, "Two-dimensional PCA: A new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, Jan. 2004.

[32] J. Zhao, Y. Zhong, H. Shu, and L. Zhang, "High-resolution image classification integrating spectral–spatial-location cues by conditional random fields," *IEEE Trans. Image Process.*, vol. 25, no. 9, pp. 4033–4045, Sep. 2016.

[33] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with Gaussian edge potentials," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 109–117.

[34] L. Zhang, L. Zhang, D. Tao, X. Huang, and B. Du, "Hyperspectral remote sensing image subpixel target detection based on supervised metric learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4955–4965, Aug. 2014.

[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[36] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. R. Salakhutdinov, "Good semi-supervised learning that requires a bad GAN," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6513–6523.

[37] S. Zheng *et al.*, "Conditional random fields as recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1529–1537.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[39] P. Zhong and R. Wang, "Learning conditional random fields for classification of hyperspectral images," *IEEE Trans. Image Process.*, vol. 19, no. 7, pp. 1890–1907, Jul. 2010.

[40] J. Zhao, Y. Zhong, and L. Zhang, "Detail-preserving smoothing classifier based on conditional random fields for high spatial resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2440–2452, May 2015.

Zilong Zhong (S'15) received the M.Eng. degree in electronic and communication engineering from Lanzhou University, Lanzhou, China, in 2014. He is currently pursuing the Ph.D. degree in machine learning and intelligence with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON, Canada.

His current research interests include deep learning, probabilistic graphical models, and their applications on computer vision tasks that involve large-scale datasets.

Jonathan Li (M'00–SM'11) received the Ph.D. degree in geomatics engineering from the University of Cape Town, Cape Town, South Africa.

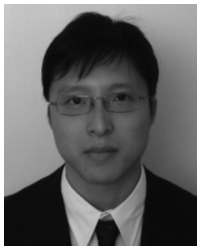
He is a Professor with the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON, Canada. He is also with the Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Information Science and Engineering, Xiamen University, Xiamen, China. He has coauthored over 300 publications, over 150 of which were published in refereed journals, including the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS (IEEE-TITS), the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING (JSTARS), the *ISPRS Journal of Photogrammetry and Remote Sensing*, and the *Remote Sensing of Environment*. His current research interests include information extraction from mobile LiDAR point clouds and from earth observation images.

Dr. Li is the Chair of the ISPRS WG I/2 on LiDAR, Air- and Space-Borne Optical Sensing from 2016 to 2020, the Chair of the ICA Commission on Sensor-Driven Mapping from 2015 to 2019, and an Associate Editor of IEEE-TITS and JSTARS.

David A. Clausi (S'93–M'96–SM'03) received the B.A.Sc., M.A.Sc., and Ph.D. degrees in systems design engineering from the University of Waterloo, Waterloo, ON, Canada, in 1990, 1992, and 1996, respectively.

He is currently a Professor of intelligent and environmental systems with the University of Waterloo. He is an Active Interdisciplinary and Multidisciplinary Researcher. He has authored or coauthored various papers published in refereed journals and conference proceedings in the diverse fields of remote sensing, computer vision, algorithm design, and biomechanics. His research efforts have led to successful commercial implementations.

Prof. Clausi was a recipient of numerous scholarships, paper awards, and two Teaching Excellence Awards; and the Research Excellence and Service to the Research Community by the Canadian Image Processing and Pattern Recognition Society in 2010. He was the Co-Chair of the International Association for Pattern Recognition Technical Committee 7 on Remote Sensing from 2004 to 2006.



Alexander Wong (M'05–SM'16) received the B.A.Sc. degree in computer engineering, the M.A.Sc. degree in electrical and computer engineering, and the Ph.D. degree in systems design engineering from the University of Waterloo, Waterloo, ON, Canada, in 2005, 2007, and 2010, respectively.

He is currently the Canada Research Chair of medical imaging systems, the Co-Director of the Vision and Image Processing Research Group, and an Associate Professor with the Department of Systems Design Engineering, University of Waterloo. He has authored refereed journal and conference papers, and patents, in various fields, such as computer vision, graphics, image processing, multimedia systems, and wireless communications. His current research interests include imaging, image processing, computer vision, pattern recognition, and cognitive radio networks, with a focus on integrative biomedical imaging systems design, probabilistic graphical models, and biomedical and remote sensing image processing and analysis, such as image registration, image denoising and reconstruction, image super-resolution, image segmentation, tracking, and image and video coding and transmission.