SAR image classification using adaptive neighborhood-based convolutional neural network

Anjun Zhang, Xuezhi Yang, Lu Jia, Jiaqiu Ai and Zhangyu Dong

School of Computer and Information, Hefei University of Technology, Hefei, P. R. China; Anhui Province Key Laboratory of Industry Safety and Emergency Technology, Hefei, P. R. China

Abstract

The convolutional neural network (CNN)-based pixel-wise synthetic aperture radar (SAR) data classification does not take fully use the spatial neighborhood information due to the fact that the impact of neighborhood pixels is not taken into consideration. The flaw of CNN-based classification method may lead to misclassification under some conditions. In this paper, we propose a novel adaptive neighborhood-based convolutional neural network (AN-CNN) for the single polarimetric synthetic aperture radar data classification. In the convolution layer, the neighborhood pixels are adaptively weighted based on their bilateral distance (spatial and feature distance) to the central pixel. In this way, different pixels have different impact on the classification result of the central pixel. The spatial distance-based weighting can reduce the misclassifications in the homogenous regions which are caused by speckle noise and the feature distance-based weighting is beneficial for the classification in the boundary regions. As a result, the misclassification is obviously reduced by the proposed AN-CNN which has a new cost function. Experimental results on simulated and real SAR data show that our proposed AN-CNN can notably improve the classification accuracy in both boundary regions and homogeneous regions compared with conventional CNN in different scenes especially when limited training samples are explored.

Keywords

Deep learning; convolutional neural network; synthetic aperture radar image classification; adaptive neighborhood; bilateral distance-based weighting

Introduction

Synthetic aperture radar (SAR) can provide a plenty of information of the land cover under all-weather and all-time conditions. As a result, SAR plays an important role in many fields, such as geology, agriculture and oceanology. Consequently, it is of great importance to interpret the SAR data accurately and effectively. Topics about SAR image interpretation such as SAR image retrieval (Tang & Jiao, 2017; Tang, Jiao, & Emery, 2017), object recognition and SAR image classification has been rising recent years. The pixel-wise SAR image classification, where each pixel in the SAR image is assigned to one class, is the most fundamental problem in SAR image interpretation. Thus, the pixel-wise SAR image classification has been widely investigated (Dekker, 2003; Ferro-Famil, Pottier, & Lee, 2000; Fukuda & Hirosawa, 1999; Tison, Nicolas, Tupin, & Maître, 2004; Tseng & Chen, 1998).

Recently, a lot of methods have been proposed for the classification of SAR data (Chapelle et al., 1999; Hou, Kou et al., 2016; McNairn, Kross, Lanen, Caves, & Shang, 2014; Ressel, Frost, & Lehner, 2015; Xiang, Tang, Hu, Li, & Su, 2014; Zhao et al., 2017). The methods for SAR data classification can be broadly divided into two categories: supervised methods and unsupervised methods. For unsupervised methods, the clustering algorithm is often applied. For example, the kernel fuzzy c-mean clustering method, which combines the pixel intensity and location information, is used to reduce the influence of speckle noise in the SAR image (Xiang et al., 2014). For supervised methods, the labeled pixels are used to generate training samples. The training samples are then employed to train the classifiers. There are series of supervised classifiers for the classification of SAR image, such as support vector machine (SVM) classifier (Chapelle et al., 1999), sparse represented classifier (SRC) (Hou, Ren et al., 2016), neural network (NN) (Ressel et al., 2015), and random forest (RF) (McNairn et al., 2014). Classification performances of these methods rely on the representative and discriminative of the extracted image features. Consequently, some carefully designed features are used for the training of classifiers. However, the designing of features requires prior knowledge and domain expertise, which reduces the robustness of the extracted features.

The feature learning, instead of feature designing, is an applicable solution to the problems mentioned above. Recently, deep learning, as one kind of attractive artificial intelligence technique, has been applied to SAR data to extract robust and discriminative features automatically. The deep learning can be realized by deep neural networks that has multiple hidden layers and...
nonlinear active function. This technique has been successfully used in the field of image classification (Krizhevsky, Sutskever, & Hinton, 2012), object recognition (He, Liu, Liu, Huang, & Yang, 2015; Simonyan & Zisserman, 2015), and remote sensing (Chen, Jiang, Li, Jia, & Ghamisi, 2016; Chen, Zhao, & Jia, 2015; Hou, Kou et al., 2016; Liu, Jiao, Hou, & Yang, 2016; Liu et al., 2017; Zhang, Ma, & Zhang, 2016, Zhao et al., 2017). There are mainly three kinds of models developed for deep learning, the stacked autoencoder (SAE) (Vincent, Larochelle, Bengio, & Manzagol, 2008), the deep belief network (DBN) (Hinton & Salakhutdinov, 2006) and the convolutional neural networks (CNNs) (Lécun, Bottou, Bengio, & Haffner, 1998). For the SAE and DBN, they are usually optimized by two steps: the layer-by-layer unsupervised pretraining and the supervised fine-tuning. For the CNN, the parameters of the neural networks are trained by the forward pass and supervised backpropagation. For the DBN and SAE, the 2-D spatial neighborhood information is inevitably destroyed because of the 1-D input of these models (Masci, Meier, Cireşan, & Schmidhuber, 2011). As for the CNN, it can preserve the spatial context because that the input of the CNN is a 2-D patch. Consequently, the CNN is superior to DBN and SAE in the image processing (Zhao & Du, 2016b). CNN has also been successfully used for the SAR image classification (Zhou et al., 2016), and SAR sea-ice concentration estimation recently (Wang, Scott, Xu, & Clausi, 2016). For the CNN-based pixel-wise SAR data classification, all the pixels from the whole patch are used to predict the label of the central pixel of the patch, which takes the local spatial information into consideration. However, in the conventional CNN, all the pixels in the patch have the same influence on the classification result. In fact, pixels with different spatial and feature distance to the central pixel should have different influence on the classification result. As a result, the classification result in boundary regions will be worse for the conventional CNN (Zhou et al., 2016).

To address the problem mentioned above, an adaptive neighborhood CNN (AN-CNN) model is proposed for the terrain classification. In this model, a new cost function is proposed containing adaptive neighborhood. The adaptive neighborhood is formed by weighting every pixel adaptively in the image patch based on the bilateral distances (spatial distance and feature distance) to the central pixel and pixels with smaller bilateral distance have larger adaptive weights. In this way, pixels with different bilateral distance to the central pixel would have different impact on the classification result. The spatial distance-based weighting can reduce the misclassification caused by the speckle noise in the homogenous regions and the feature distance-based weighting can improve the classification accuracy in the boundary regions. Experimental results on the simulated single look SAR image and real SAR image show that the AN-CNN can improve the classification accuracy in both boundary and homogenous regions notably.

The outline of this paper is organized as follows. The background and the related work are introduced in the Section 2. Section 3 presents the proposed AN-CNN model and the corresponding classification method in detail. Experimental results on the simulated and real SAR images are represented in Section 4. Finally, the conclusion and the future work are shown in Section 5.

**Background**

CNN is a machine learning model proposed by LeCun (Lécun et al., 1998). The CNN consists of a stack of convolution layers and pooling layers with the fully connected layers on the top. The structure of the CNN is shown in Figure 1. Filters and biases in the convolution layer and the weight matrixes in fully connected layers are the parameters to be learned.

The convolution layer is used to extract the features of the input. The filter moves with a certain step within the input image to generate a convolved image. The convolved image is then transformed by a nonlinear active function to generate the "feature map." Through multiple filters, multiple feature maps can be generated. The process mentioned above is described as follows:

![Figure 1.](image-url)
\begin{equation}
F_{ij} = g((W * X)_{ij} + b), \quad i = 1, 2, ..., C_x; \\
\quad \quad j = 1, 2, ..., C_y
\end{equation}

\begin{equation}
C_x = \frac{I_x - K_x}{S} + 1
\end{equation}

\begin{equation}
C_y = \frac{I_y - K_y}{S} + 1
\end{equation}

In Equations (1–3), the input image is represented as \( X \). The width and height of the input image is denoted by \( I_x \) and \( I_y \). The filter \( W \) whose width and height are \( K_x \) and \( K_y \) is used to convolve with the input image and \( S \) denotes the moving step of the filter. The operation of convolution is represented as “\( * \)”. The bias of the convolution layer is denoted as \( b \) and the nonlinear active function is expressed as \( g \). After the nonlinear transform, a feature map will be generated which is represented as \( F \).

The generated feature map is then subsampled by the average pooling layer. In this layer, each nonoverlapped rectangle image patch is mapped as the average value of the corresponding patch. After the average pooling, the data size will be reduced and the noise of the input will be smoothed with the statistical property of the input feature map being unchanged (Geng, Fan, Wang, & Ma, 2015). Through the operation of pooling, the upper convolution layer will have larger receptive fields than the lower ones, which enables the upper convolution layer to model larger scale of spatial context information.

The fully connected layer locates on the top of the convolutional and pooling layers which stack alternatively upon each other. The feature maps subsampled by the last pooling layer are reshaped into a 1-D feature vector. The generated feature vector will then be used for the classification of the category of the terrain.

**Proposed method**

In our work, the adaptive neighborhood-based CNN (AN-CNN) is proposed for the classification of SAR image. In the proposed framework, the adaptive neighborhood is constructed to highlight the impact of neighborhood pixels on the classification result of the central pixel, in which way the misclassification can be notably reduced. The structure of AN-CNN is shown in Figure 2 (a). An example of real image is used to illustrate the strategy of AN-CNN. The real image is used as the input and the corresponding adaptive neighborhoods are used to represent the AN-CNN layers. In Figure 2(b), a real image has been used to illustrate the mechanism of adaptive neighborhood of AN-CNN. The creating of the neighborhood weights matrix and the constructing of the adaptive neighborhood are shown. For the neighborhood weights matrix, the higher weight value is represented with lighter color.

**The construction of the adaptive neighborhood**

The conventional CNN utilizes image patches to perform the pixel-wise SAR image classification. Although the spatial neighborhood information is used, the difference of the impacts of the neighborhood pixels on the classification results of the central pixels has not been considered. This may lead to misclassification under some conditions, take Figure 3 as an example:
Figure 3 illustrates image patches with the corresponding groundtruth of the pixels. There are two pictures in Figure 3. The Figure 3(a) illustrates an image patch extracted from a homogenous region which is heavily contaminated by speckle noise and Figure 3(b) illustrates the image patch containing a boundary region. In Figure 3(a), the central pixel and its nearest neighborhoods are labeled as 1 while most of the pixels in the patch are labeled as 2 due to the disturbance of speckle noise. The conventional CNN may misclassify the central pixel as 2 because most of the pixels in the image patch are labeled as 2. To overcome this problem, the spatial distance-based weighting is utilized, that is to say, the labeled pixels as 1 have larger weights while the pixels labeled as 2 have smaller weights, in this way, the misclassification can be eliminated. Through the spatial distance-based weighting, the misclassification in the homogenous region, which is cause by the speckle noise, can be reduced. However, the spatial distance-based weighting cannot work well in boundary region, such as Figure 3(b). It can be easily seen that the central pixel may still be misclassified as 2 although the spatial distance-based weighting is applied. Under this condition, the feature distance-based weighting should be exploited, where the pixels labeled as 1 owns larger weights and the pixels labeled as 2 have smaller ones. Taking these two conditions into consideration, a kind of jointly weighting strategy is implemented, the weight of an arbitrary pixel in the image patch is calculated based on the bilateral distance to the central pixel. The bilateral distance is described as follow:

\[ d_{i,j} = \sqrt{(i-i_c)^2 + (j-j_c)^2}, \quad i = 1, \ldots, w, \]

\[ j = 1, \ldots, h \]  \hspace{1cm} (4)

Equation (4) describes the spatial distance of an arbitrary pixel in the image patch to the central pixel. The Euclidean distance is applied to represent the spatial distance. The \((i, j)\) represents the coordinate of the pixel and the constants \(w\) and \(h\) represents the width and the height of the image patch. Equation (5) illustrates the intensity feature distance of an arbitrary pixel to the central pixel and the directed-divergence distance (Kullback, 1959) is applied to describe the feature distance. In Equation (5), \(g_{ij}\) represents the intensity of an arbitrary pixel and the \(g_{kc}\) denotes the intensity of the central pixel. Based on the spatial and feature distance, a kind of jointly weighting strategy is implemented, which can be described as:

\[ W_s(i,j) = e^{-d_{i,j} / dow_1} \]  \hspace{1cm} (6)

\[ W_f(i,j) = e^{-d_{i,j} / dow_2} \]  \hspace{1cm} (7)

\[ W_b(i,j) = \lambda W_s(i,j) + (1 - \lambda) W_f(i,j) \]  \hspace{1cm} (8)

Equations (6) and (7) describe the spatial distance-based and feature distance-based weights of the pixel \((i, j)\). The weights are constructed using the negative exponential function, which makes the weights nonnegative and monotonically decreasing and the maximum of the weights equal one when the distances equal zero. The constants dow_1 and dow_2 are set to control the degree of weighting (DOW), which makes the local feature within the receptive fields undamaged. Through the controlling of the DOWs, the weighting of the pixels is equivalent to the weighting of local features. The bilateral distance-based weight of an arbitrary is calculated using the weighted-sum approach, which is described in Equation (8). The constants \(\lambda\) and \(1-\lambda\) are the coefficients of the spatial
weights ($SW$) and feature weights ($FW$). Through calculating the weight of every pixel in an image patch, the neighborhood weights matrix of this image patch will be obtained. By weighting every pixel in the image patch using the created neighborhood weights matrix, the adaptive neighborhood of this patch can then be constructed (see Figure 2(b)). The adaptive neighborhood is demonstrated in Figure 4 with examples of a real image patch. Figure 4(a) shows the original image patch which consists of boundary region and homogenous region. Figure 4(b–d) illustrate the neighborhood weights matrices with varies $\lambda$ values ($\lambda = 0.5$, $\lambda = 0.9$, and $\lambda = 0.1$). For the neighborhood weights matrices, the lighter color represents the higher value. Figure 4(e–g) demonstrate the adaptive neighborhoods with $\lambda$ values mentioned above. It can be seen from Figure 4 that the different neighborhood weights matrices can be constructed by using different values of $\lambda$. Through changing $\lambda$, the influence of spatial distance-based weight and feature distance-based weight will be changed. According to Equation (8), when $\lambda$ is higher, the spatial distance-based weighting is more important. With smaller $\lambda$, the feature distance-based weighting plays a more important role. This conclusion can be demonstrated by the neighborhood weights matrices with different $\lambda$ values shown in Figure 4(b)–(d).

### Training of AN-CNN

Adaptive neighborhood is introduced into the convolution layer of CNN to construct the AN-CNN. The parameters of the AN-CNN should be optimized for the feature extraction and classification of SAR data. In the course of training, the outputs of the AN-CNN are forced to approach the labels of the training samples. In the AN-CNN, a new cost function which should be minimized during the course of training is proposed and the new cost function is defined as:

$$J(F(X; W), y) = \frac{1}{2N} \sum_{n=1}^{N} (F(P_i X_i; W) - y_i)^2$$

In Equation (9), $F()$ represents the feed-forward function of AN-CNN which is used to predict the category labels of input image patches. The AN-CNN takes $X$ as the input to estimate the categories of the terrains based on its parameters $W$ and the labels of the $X$ are represented as $y$. The training samples are divided into mini-batches. The total number of training samples in a mini-batch is denoted as $N$. The $i$th training sample in a mini-batch is denoted as $X_i$ and the corresponding adaptive weight is illustrated as $P_i$. The corresponding adaptive neighborhood is then formed by element-wised multiplication (denoted as “$\cdot$”) between $X_i$ and $P_i$. The adaptive
weights matrix of the $i$th training sample in a mini-batch, which are denoted as $P_i$, are described as follows:

$$
P = \begin{pmatrix}
W_{b(1,1)}^i & \cdots & \cdots & W_{b(1,n)}^i \\
\vdots & \ddots & \vdots & \vdots \\
W_{b(p,q)}^i & \cdots & \cdots & \vdots \\
W_{b(n,1)}^i & \cdots & \cdots & W_{b(n,n)}^i
\end{pmatrix}
$$

where $W_{b(p,q)}^i$ is the adaptive weight of the point ($p, q$) in the $i$th training sample, which can be calculated according to Equation (8).

The backpropagation algorithm and the mini-batch SGD algorithm are used to train the AN-CNN. Parameters of the AN-CNN should be optimized to minimize the cost function and the parameters are updated according to the derivatives of the parameters to the cost function, which can be described as:

$$
\Delta W_{lm}^i = \text{rot180}(\text{conv}(h_{l-1}^{(l-1)}, \text{rot180}(\sigma_n^{()}))) / N_{\text{batchsize}}
$$

In Equations (11) and (13), the update functions of weights and biases are given. The numerical expresses of the derivatives of weights and biases to the cost function are described in Equations (12) and (14), respectively. In Equation (11), $W_{l+1}^i$ is the weight of AN-CNN in the $l$th layer at the iteration $i + 1$ and $W_l^i$ is the weight of the AN-CNN in the same layer at the iteration $i$. In Equation (13), the biases of AN-CNN in the $l$th layer update in the same way as the weight. In Equation (12), $l$ represents the $l$th layer, $m$ and $n$ represent the $m$th and $n$th feature map in the $l-1$th and $l$th layer. $h$ and $\sigma$ denote the hidden layer and error term respectively. The error term of each layer can be calculated by BP algorithm (Bouvrie, 2006). The function rot180() means rotting a matrix by 180 degrees and conv() represents the convolution operation. $N$ represents the number of training samples in a mini-batch. In Equation (14), $\sigma_j$ represents error term in the $l$th layer and $j$ denotes the $j$th feature-map. The $(p,q)$ represents the location of an arbitrary pixel. Learning rate for the updating of parameters is represented as $\alpha$. The learning rate is an important parameter controlling the updating step of the parameters. In this study, the learning rate is experimentally set as 0.05. After the training of AN-CNN, the testing samples will be used to evaluate the classification performance of the trained models and the models with the

$$
\Delta b = \sum_p \sum_{q} \sigma_j^{(p,q)} / N_{\text{batchsize}}
$$

Figure 5. Flow chart of the SAR image classification using AN-CNN.
optimized parameters will be picked out for the classification of SAR data. The procedure is shown in Figure 5.

**Experiments and results**

**Parameters and datasets**

In this study, a simulated SAR image and two real SAR images are used to validate the superiority of AN-CNN over the conventional methods. The single-look simulated SAR data is synthesized by adding the speckle noise to the hyperspectral image. The statistical property of the speckle noise obeys the Gamma distribution and the speckle noise is used to prove the robustness of the AN-CNN to the speckle noise. For the simulated SAR data, the widely used “University of Pavia” hyperspectral dataset, which is available online,¹ is applied to synthesize the simulated SAR image. This dataset is chosen to validate the superiority of CNN in the classification of heterogeneous regions because that there exists lots of heterogeneous regions in this dataset. Note that the noisy bands of the “University of Pavia” are removed (Chen et al., 2016). In this study, the 100th band, whose visual effect is clean, is chosen to synthesize the simulated SAR data, which is shown in Figure 6 (a). The size of the simulated image is 610 × 340. Standard groundtruth (Chen et al., 2016) of this dataset is applied and there are nine classes in the groundtruth, as shown in Figure 6(b).

The Radarsat-2 SanFrancisco-Bay (RS2-SF-Bay) dataset (Samat, Gamba, Du, & Luo, 2015) and the Radarsat-2 Flevoland (RS2-Flevoland) dataset (Liu et al., 2017; Samat et al., 2015) are adopted as the real SAR data. Note that both of these datasets are available online². The format of the SAR data is single-look-complex and the HH components are extracted from the PolSAR data. For the RS2-SF-BAY dataset, a subregion with 1010 × 1160 pixels is selected, as shown in Figure 6(c). The groundtruth of the dataset shown in Figure 6(d) is referred from (Samat et al., 2015). It includes mainly five types of terrains: water, vegetation, build-up1, build-up2, and build-up3. For the RS2-Flevoland, a subregion with 1000 × 1400 pixels is shown in Figure 6(e). The groundtruth data is referred from (Samat et al., 2015), which includes five terrains: water, forest, farmland1, farmland2, and urban as illustrated in Figure 6(f).

For all these three datasets, we split labeled samples into two subsets: training samples and testing samples. One thousand or two thousand samples are randomly selected as the training samples. Note that the selected training samples distribution in the whole part of the simulated and real SAR images, which is reasonable for the training of classifiers. The remained samples are used to test the classification performance of the classification methods.

The structural parameters of CNN should be firstly determined. The inputs of the CNN are image patches with the size of 27 × 27. The size of image patches is referred to in (Chen et al., 2015), which shows a balance between the classification accuracy and the computational cost. The other structural parameters which include the number of layers (NOL), the number of feature-maps (NOFM) in each layer and the number of iterations (NOI) are optimized experimentally. The methodology of parameters optimization of CNN is taken from (Zhao & Du et al. 2016a). The number of feature-maps in each layer are firstly determined. In this step, the number of layers of CNN is fixed as two (Note that a couple of convolution-pooling layers is defined as one layer). During the course of optimization, the NOFM in each layer varies from 16 to 22 with a step of 1 and the optimized NOFM will be picked out. Note that the NOFMs in each layer are the same. The results of the optimization are illustrated in Figure 7.

According to the optimization result, the CNN achieves the highest classification accuracy when there are 20 feature-maps in each layer of CNN. When the NOFM is too small, the extracted features are not representative and discriminative enough. However, too large NOFM may lead to redundancy of features which can also reduce the classification accuracy. Based on the analysis above, the NOFM of each CNN layer is optimized as 20.

The NOL and NOI are then jointly optimized as is done in (Zhao & Du, 2016a). During the course of optimization, the NOL varies from 1 to 3 with a step of 1 and NOI varies from 50 to 200 with a step of 50. For CNN with three layers, the kernel size of convolution in the third layer is set as 4 and the pooling rate in this layer is set as 1.

The optimization results are shown in Figure 8. The optimized NOL is set as 2 and the NOI is chosen as 100, which shows a trade-off between the time cost and classification accuracy. For NOL, the deeper CNN can extract higher level of features. If the structure of CNN is too shallow, the extracted features are not representative and discriminative enough for SAR image classification. However, the number of parameters will be larger as the CNN goes deeper, which may lead to over-fitting and higher time cost. For CNN with 1 layer, the obtained highest classification accuracy is 71.07% due to the limited representation of extracted features. For CNN with 3 layers, it takes 150 iterations to obtain the highest classification accuracy (80.01%) which is slightly higher than that of CNN with 2 layers (79.43%). However, it takes much longer time to train the CNN. Thus, the NOL

¹Available online: www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes.
²Available online: ftp://rsat2:yvr578MM@ftp.mda.ca/.
is optimized as 2 considering the trade-off between classification accuracy and time cost. For CNN with two layers, the optimized NOI is 100. When NOI equals 50, the CNN cannot be sufficiently trained. In reverse, too large NOI may lead to overfitting which limited the classification of CNN. The structural parameters of AN-CNN are the same with those of CNN for the sake of a fair comparison.

Based on the optimized parameter setting, the size of feature maps in each layer can be determined. The size of filters in the first convolution layer is $4 \times 4 \times 20$. Thus, the size of the convolved image is $24 \times 24 \times 20$ (Wang et al., 2016). An even value ($4 \times 4$) is chosen for the kernel size of the first convolution layer because that only by setting the kernel size of the first layer as an even value, the size of the feature maps which are subsampled by the pooling layer can be an integer. When performing the convolution operation, the top left pixel can be regarded as the central pixel. The first convolution layer is followed by the average-pooling layer with the pooling ratio of $2 \times 2$. The average pooling layer outputs the average value of every unoverlapping $2 \times 2$ patches of

![Figure 6. Datasets and their corresponding groundtruths.](image1)

![Figure 7. The optimization of the number of feature maps in each layer.](image2)
the feature maps. The size of the feature maps becomes 12 × 12 × 20 after the average pooling. The size of the filters in the second convolution layer is 5 × 5 × 20. Thus, it outputs feature maps of size 8 × 8 × 20. The second pooling layer is the same as the first pooling layer and the size of the corresponding output feature map is 4 × 4 × 20. In the fully connected layer, the feature maps will be reshaped to form a feature vector with the dimension of 4 × 4 × 20 (320). Finally, a single layer perceptron is used to predict the labels of the samples, which show a tradeoff between classification accuracy and computational cost. ReLU is chosen to be the active functions of the convolution layer because it has been demonstrated to have better performance than saturating function such as sigmoid (Krizhevsky et al., 2012). However, the active function of the classifier is chosen to be sigmoid function due to the fact that the sigmoid function can normalize the feature vectors into [0,1].

Overall accuracy (OA) is utilized as statistical metric to evaluate the classification accuracy of the proposed method and compared methods. The OA is described as the following equation:

$$\text{OA} = \frac{N_{\text{correct}}}{N_{\text{total}}} \times 100\% \quad (15)$$

In Equation (15), $N_{\text{total}}$ refers to the total number of samples and $N_{\text{correct}}$ denotes the number of correctly classified samples.

For comparison, six methods which includes the conventional methods and related state-of-art methods are exploited. The conventional CNN whose structural parameters are the same with that of AN-CNN is used for comparison. As referred to (Zhao et al., 2017), the DBN and SAE are also applied as the conventional. The DBN exploited in this study includes two hidden layers and both of them owns 100 neurons. The structure of SAE is in the same with that in (Zhang et al., 2016). The SAE used for comparison has three hidden layers. Each of the first two hidden layers has 200 neurons while the third layer has 100 neurons. The patch vector which is formed by extending the image patch into a vector is used as the input of DBN and SAE. The support vector machine (SVM) is applied as the base-line method. According to (Zhao et al., 2017), both the patch vector and the GLCM+Gabor feature vector are used as the input of SVM. The patch vector is formed by extending the image patch into a vector. The GLCM+Gabor features include the texture features and the mutiorientation boundary features. The GLCM texture features consist of the mean and variance of the energy, entropy, inertia moment and correlation of the GLCM. For Gabor filters, features with 8 different orients which includes [0, (π/8), (π/4), (3π/8), (π/2), (5π/8), (3π/4), (7π/8)] are extracted. After the feature extraction, GLCM features and Gabor features are fused using principle component analysis (PCA) and the first principle component is selected as the fused features (GLCM+Gabor feature vector). For the SVM, the radial basis function kernel is utilized. The gamma parameter of kernel is experimentally set to be 0.02. In (Zhang et al., 2016), a spatial distance-based weighting strategy is proposed to utilize the neighborhood information in Stacked Autoencoder (SAE), which can be described as:

$$W_{(i,j)} = \frac{1}{1 + d_{(i,j)}} \quad (16)$$

This weighting strategy is also used in CNN to form the spatially weighting-based CNN (SW-CNN) for the comparison. The SW-CNN is explored as the related state-of-art method for comparison.
Results on simulated SAR data and discussions

The simulated SAR data is used to evaluate the classification performance of the AN-CNN. The main parameters for the AN-CNN are determined as follow. The DOWs (See Equations 6 and 7) of spatial distance-based weight and feature distance are set to be 40 and 1 respectively according to the optimized results of grid research method with the step of 10. Coefficients of SW(λ) and FW(1−λ) are also optimized through grid research method to further improve the classification performance of AN-CNN. Figure 9 illustrates the relationship between OA and the value of λ. The optimal value for λ is 0.9, which demonstrates that the bilateral distance-based weighting outperforms the unilateral distance-based weighting (when λ = 1 or λ = 0). According to Equation (8), only the spatial distance-based weighting works when λ equals 1. In the same way, only the feature distance-based weighting works when λ equals 0. In contrast, the bilateral distance-based weighting works when λ does not equals 1 or 0.

In this experiment, 32,800 samples are labeled while only 1000 or 2000 samples are used for the training of CNN. The experimental results are shown in Table 1 in detail. With 1000 training samples, the AN-CNN achieves the best classification performance, which gets 81.35% OA and the conventional CNN obtains 79.34% classification accuracy. The SW-CNN achieves 79.12% overall accuracy, which is not higher than that of CNN. This can be explained by the fact that the spatial weighting may destroy the local features. The classification accuracy achieved by DBN is 64.17% and the SAE obtains 63.25% OA. This result demonstrates that the CNN outperforms the DBN and SAE. This is because that the DBN and SAE is vulnerable to speckle noise. The GLCM+Gabor+SVM shows the higher classification accuracy (69.76%) than patch-vector+SVM (65.56%). This is because that the constructing of GLCM+Gabor features utilizes spatial neighborhood information so that the disturbance of speckle noise can reduced. It can be seen that the AN-CNN improves the classification accuracy notably compared with CNN. This can be explained by the fact that the AN-CNN can use the neighborhood information more reasonably. As the AN-CNN can extract the robust and discriminative features, thus the classification accuracy of AN-CNN is much higher than that of SVM. The classification results with 1000 samples are shown in Figure 10(a–g). One can see that the AN-CNN shows higher classification accuracy in the boundary regions (highlighted by the rectangles) comparing with CNN, which demonstrates the effect of the adaptive neighborhoods. Compared with the classification results of DBN and SAE, the classification result of AN-CNN is much smoother, which indicates that the AN-CNN is robust to the speckle noise.

Table 1. The OAs of all the explored algorithms on PaviaUniversity, SanFrancisco Bay, and Flevoland datasets.

|                     | CNN       | AN-CNN    | SW-CNN    | Patch-vector+SVM | DBN       | SAE       | GLCM+Gabor+SVM |
|---------------------|-----------|-----------|-----------|------------------|-----------|-----------|----------------|
| 1000 training samples (PaviaUniversity) | 79.34%    | 81.35%    | 71.32%    | 65.70%           | 64.17%    | 64.25%    | 69.51%         |
| 2000 training samples (PaviaUniversity) | 85.82%    | 87.20%    | 77.01%    | 68.34%           | 65.82%    | 66.11%    | 71.87%         |
| 1000 training samples (SanFrancisco Bay) | 77.96%    | 80.48%    | 70.68%    | 68.23%           | 64.61%    | 64.97%    | 71.31%         |
| 2000 training samples (SanFrancisco Bay) | 81%       | 83.90%    | 74.41%    | 71.17%           | 66.23%    | 66.71%    | 74.43%         |
| 1000 training samples (Flevoland)       | 84.40%    | 85.41%    | 83.74%    | 82.35%           | 81.30%    | 81.03%    | 83.51%         |
| 2000 training samples (Flevoland)       | 86.20%    | 87.13%    | 86.81%    | 83.48%           | 81.53%    | 81.40%    | 84.73%         |

Figure 9. The overall accuracy as a function of λ for the simulated SAR data.
noise. There are more isolated misclassification points in the classification result of patch-vector +SVM method than that of GLCM+Gabor+SVM method, which demonstrates the speckle immunity of GLCM+Gabor features.

Results on Radarsat-2 SanFrancisco Bay dataset and discussions

To further prove the superiority of the proposed AN-CNN, the real SAR data should be used for the experiment. The Radarsat-2 SanFrancisco Bay dataset is firstly used to evaluate the performance of AN-CNN. The parameters of the adaptive neighborhood are also optimized by experiments. Based on the experimental results, the DOWs (see Equations 6 and 7) of spatial distance-based weighting and feature distance-based weighting are set to be 40 and 10, respectively. The optimized value for $\lambda$ is 0.1, which demonstrates that the bilateral distance-based weighting is superior over unilateral distance-based weighting in the real SAR image dataset. The classification accuracy as a function of $\lambda$ is shown in Figure 11.

The experimental results of different methods are listed in Table 1. With 1000 and 2000 training samples, the AN-CNN shows the best performance and the corresponding are 80.48% and 83.90%, respectively. The conventional CNN illustrates the secondly best classification accuracy. The SW-CNN achieves lower classification accuracy than CNN because of the unsuitable weighting of neighborhood pixels, which destroys the local features of the input image patch. The classification accuracies of DBN and SAE are even lower than that of patch-vector+SVM, which might be caused by
the disturbance of speckle noise. The patch-vector + SVM method shows the lower classification accuracy than that of GLCM+Gabor+SVM method. It can be concluded that the feature selection is crucial for conventional classifier. The GLCM features and the Gabor features are extracted based on image patches, so that the influence of speckle can be reduced. The classification results with 1000 training samples are illustrated in Figure 12(a)–(g). The proposed AN-CNN has the best visual effect for each of the five terrains. The results indicate that the AN-CNN shows better classification performance in both the homogeneous region (highlighted by circles) and the boundary regions (highlighted by rectangles) when compared with CNN. In the homogeneous regions, the superiority of AN-CNN derives from the spatial distance-based weighting, which can improve the robustness to speckle noise. In the boundary regions, the feature distance-based weighting can effectively reduce the misclassification. Consequently, the AN-CNN induces obvious improvement of classification performance in this dataset. On the classification results of SAE and DBN, it can be seen that there are lots of isolated misclassified points, which demonstrates the disturbance of speckle noise. Compare Figure 12(c) with Figure 12(g), it can be found that the Figure 12(g) is smoother than Figure 12(c). This is because that the GLCM+Gabor features can resist the influence of speckle noise due to the exploration of spatial information.

**Results on Radarsat-2 Flevoland dataset and discussions**

The Radarsat-2 Flevoland dataset, with a resolution of 6 m, is also used to evaluate the robustness of the adaptive weighting strategy. For this dataset, the

![Figure 11](image1.png)  
*Figure 11. The overall accuracy as a function of λ for the RS2-SF-BAY SAR data.*

![Figure 12](image2.png)  
*Figure 12. The classification results of the Radarsat-2 SanFrancisco Bay dataset. (a) Results with CNN. (b) Results with AN-CNN. (c) Results with SVM. (d) Results with SW-CNN. (e) Classification results with DBN. (f) Classification results with SAE. (g) Classification results with GLCM+Gabor+SVM.*
parameters of adaptive neighborhood are also optimized by experience and experiments. The optimized value of $DOW_1$ (see Equation 6) is 50 and that of $DOW_2$ (see Equation 7) is 30. Coefficients of the weights are also optimized and the $\lambda$ is set to be 0.1. The classification accuracy as a function of $\lambda$ is demonstrated in Figure 13. It can be concluded that the bilateral distance-based weighting is more advanced than unilateral distance-based weighting.

The classification accuracy of different methods is shown in Table 1. It can be seen that the AN-CNN still improve the classification accuracy obviously compared with the other methods, which indicates the robustness of the proposed AN-CNN. The overall accuracy of AN-CNN is more than 85% with only 1000 samples (less than 1% of the labeled samples), which is much higher than the OA of SVM, DBN and SAE. Compared with the SVM, the DBN, and the SAE, the advantage of AN-CNN can be attributed to the preservation of the spatial neighborhood information. The AN-CNN can still outperform the CNN, which demonstrates the robustness of the adaptive neighborhoods on different scenes. The Gabor +GLCM+SVM achieves much higher OA than those of the DBN, the SAE and the patch-vector+SVM, which demonstrates the importance of speckle immunity for SAR image classification. The classification results with 1000 training samples are shown in Figure 14(a–g). The result of the AN-CNN shows the best visual effect. In Figure 14(c), (e), and (f), there exist lots of isolated points because that the patch-vector+SVM, the DBN and the SAE is susceptible to speckle noise. However, the result of AN-CNN is much smoother due to the fact that the AN-CNN is robust to the speckle noise, which can be attributed to the use of neighborhood information. Compared with the CNN, the AN-CNN shows higher classification accuracy in each kind of terrain because of the superiority of the adaptive neighborhoods.

**Selecting the training samples according to a certain proportion**

The experimental results mentioned above demonstrates the superiority of AN-CNN over CNN when the training samples are limited. However, in some recent papers, the training samples are randomly according to a certain proportion (Zhao et al., 2017). In this section, the classification performance of CNN and AN-CNN are investigated when a certain proportion of samples are selected as training samples. In this section, the PaviaUniversity simulated SAR image, the Radarsat-2 SanFrancisco-Bay real SAR image and Radarsat-2 Flevoland real SAR image are explored for evaluation. Parameters of AN-CNN are experimentally optimized on these datasets using grid research respective. On PaviaUniversity image, 10% of samples are used for training. For the two real SAR image, 1% of samples are selected as training samples, which shows a trade-off between classification accuracy (about 90%) and time cost. As for the real SAR image, the number of samples is large. As a result, 1% of samples are enough to train the CNN and the AN-CNN. For all the simulated SAR image, the $\lambda$ is optimized as 0.4. For the Radarsat-2 SanFrancisco-Bay dataset and Radarsat-2 Flevoland real SAR images, the $\lambda$ is optimized as 0.6 and 0.1, respectively.

The overall accuracies of AN-CNN and CNN on these three datasets are shown in Figure 15. On the dataset of PaviaUniversity, the OA of CNN is 92.75% while that of AN-CNN is 94.57%. On real SAR datasets, the AN-CNN can still outperforms the CNN. For the Radarsat-2 SanFrancisco-Bay dataset, the AN-CNN achieves 90.67% OA and the OA obtained by CNN is 89.78%. For the Radarsat-2 Flevoland dataset, the OA of AN-CNN (90.54%) can still surpass that of CNN (89.35%). This result indicates that the AN-CNN can still outperform CNN when a relatively abundant number of training samples are explored. However, the superiority of AN-CNN is less obvious when a certain percent of labeled samples are exploited as training samples. The reason of this phenomenon is that the disturbance of speckle noise will be reduced when more training samples are exploited. The speckle noise can also be regarded as redundancy of information. When the number of training samples are limited, the features learned by CNN is not representative enough due to the influence of redundancy. With the increasing of the number of training samples, the disturbance of redundancy will be reduced so that the features learned by conventional CNN will be
more representative and discriminative. Due to discussions made above, the advantage of AN-CNN over conventional CNN will be reduced with the increasing of the number of training samples. The advantage of the adaptive neighborhood strategy still maintains when the number of training samples increases because of its benefits on narrow (boundary) regions caused by the feature distance-based weighting and higher boundary locating accuracy caused by the spatial distance-based weighting (Pan & Zhao, 2017).

![Figure 14. The classification results of the Radarsat-2 Flevoland dataset. (a) Results with CNN. (b) Results with AN-CNN. (c) Results with SVM. (d) Results with SW-CNN. (e) Classification results with DBN. (f) Classification results with SAE. (g) Results with GLCM+Gabor+SVM.](image-url)
Conclusion

In this paper, a novel deep learning framework, which is called AN-CNN, is proposed to extract discriminative features and classify the SAR data in this paper. In our proposed AN-CNN, both the spatial context information and the impact of neighborhood pixels on the classification results are taken into consideration, which leads to the reduction of misclassification. Experiment results on the simulated and real SAR image show the following:

1. The AN-CNN can notably improve the overall classification accuracy on both simulated and real SAR data with little amount of training samples when compared with the conventional methods.
2. Classification result indicates that the AN-CNN can improve the classification accuracy in both narrow regions and homogenous regions.

In the future, we will develop and investigate more deep learning-based methods for SAR data classification.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the Anhui Province Natural Science Foundation [1608085QF142]; National Natural Science Foundations of China [41601452,61371154]; Fundamental Research Funds for the Central Universities [JZ2017YYPY0237]; Key Research and Development Program of Anhui province [1704a0802124].

References

Bouvrie, J. (2006) Notes on convolutional neural networks. Neural Networks. 1–8.
Chapelle, O., Haffner, P., & Vapnik, V.N. (1999). Support vector machines for histogram-based image classification. IEEE Transaction on Neural Network, 10(5), 1055–1064. doi:10.1109/72.788646
Chen, Y., Jiang, H., Li, C., Jia, X., & Ghamisi, P. (2016). Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. IEEE Transactions on Geoscience and Remote Sensing, 54(10), 6232–6251. doi:10.1109/TGRS.2016.2584107
Chen, Y., Zhao, X., & Jia, X. (2015). Spectral–spatial classification of hyperspectral data based on deep belief network. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 8(6), 2381–2392. doi:10.1109/JSTARS.2015.2388577
Dekker, R.J. (2003). Texture analysis and classification of ERS SAR images for map updating of urban areas in the Netherlands. IEEE Transactions on Geoscience and Remote Sensing, 41(9), 1950–1958. doi:10.1109/TGRS.2003.814628
Ferro-Famil, L., Pottier, E., & Lee, J.S. (2000). Unsupervised classification of multifrequency and fully polarimetric SAR images based on the H/A/Alpha-Wishart classifier. IEEE Transactions on Geoscience and Remote Sensing, 39 (11), 2332–2342. doi:10.1109/36.964969
Fukuda, S., & Hirosawa, H. (1999). A wavelet-based texture feature set applied to classification of multifrequency polarimetric SAR images. IEEE Transactions on Geoscience and Remote Sensing, 37(5), 2282–2286. doi:10.1109/36.789624
Geng, J., Fan, J., Wang, H., & Ma, X. (2015). High-resolution SAR image classification via deep convolutional autoencoders. IEEE Geoscience and Remote Sensing Letters, 12 (11), 2351–2355. doi:10.1109/LGRS.2015.2478256
He, S., Lau, R., Liu, W., Huang, Z., & Yang, Q. (2015). SuperCNN: A superpixelwise convolutional neural network for salient object detection. International Journal of Computer Vision, 115(3), 330–344. doi:10.1007/s11263-015-0822-0
