remote sensing

MDPI

Article

PolSAR Image Classification via Learned Superpixels and QCNN Integrating Color Features

Xinzheng Zhang [1,*], Jili Xia [1], Xiaoheng Tan [1], Xichuan Zhou [1] and Tao Wang [2]

[1] College of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400044, China

[2] Key Laboratory of Aerocraft Tracking Telementering & Command and Communication, Chongqing University, Chongqing 400044, China

* Correspondence: zhangxinzheng@cqu.edu.cn

Received: 27 June 2019; Accepted: 1 August 2019; Published: 6 August 2019

Abstract: Polarimetric synthetic aperture radar (PolSAR) image classification plays an important role in various PolSAR image application. And many pixel-wise, region-based classification methods have been proposed for PolSAR images. However, most of the pixel-wise methods can not model local spatial relationship of pixels due to negative effects of speckle noise, and most of the region-based methods fail to figure out the regions with the similar polarimetric features. Considering that color features can provide good visual expression and perform well for image interpretation, in this work, based on the PolSAR pseudo-color image over Pauli decomposition, we propose a supervised PolSAR image classification approach combining learned superpixels and quaternion convolutional neural network (QCNN). First, the PolSAR RGB pseudo-color image is formed under Pauli decomposition. Second, we train QCNN with quaternion PolSAR data converted by RGB channels to extract deep color features and obtain pixel-wise classification map. QCNN treats color channels as a quaternion matrix excavating the relationship among the color channels effectively and avoiding information loss. Third, pixel affinity network (PAN) is utilized to generate the learned superpixels of PolSAR pseudo-color image. The learned superpixels allow the local information exploitation available in the presence of speckle noise. Finally, we fuse the pixel-wise classification result and superpixels to acquire the ultimate pixel-wise PolSAR image classification map. Experiments on three real PolSAR data sets show that the proposed approach can obtain 96.56%, 95.59%, and 92.55% accuracy for Flevoland, San Francisco and Oberpfaffenhofen data set, respectively. And compared with state-of-the-art PolSAR image classification methods, the proposed algorithm can obtained competitive classification results.

Keywords: polarimetric synthetic aperture radar (PolSAR); image classification; deep learning; superpixels; quaternion convolutional neural network (QCNN)

1. Introduction

1.1. Background

Synthetic aperture radar (SAR) is an important branch of remote sensing application, which has the imaging capability of all-time, all-weather, and has the ability to penetrate observed objects to a certain extent. As an advanced type of SAR, polarimetric SAR (PolSAR) is able to record the complete scattering information of the observed objects. Since the polarization of electromagnetic waves is very sensitive to the shape, properties and physical structure of the objects, utilizing polarimetric information is benefit to PolSAR image classification [1]. However, due to the unique imaging mechanism and the complex imaging environment, PolSAR image classification is still a challenging task.

The choice of features is an essential part for PolSAR image classification. The scattering matrix and the coherence/covariance matrix are taken as the direct features and are often used in classification. Initially, HH,HV,VV or their combination are utilized as the underlying features to classify PolSAR images [2–6]. The coherence/covariance matrix which can provide more complex polarimetric information, such as amplitude, phase, is widely used for PolSAR classification [1,7–9]. However, these features are easily affected by speckle noise. Polarimetric decomposition , such as Pauli decomposition, Cloude decomposition, and so on [10], describes average backscatter as the sum of independent components, which can provide better PolSAR image interpretation than other features. By mapping polarimetric composition components to color channels, a pseudo-color image can be acquired, which is always as a visual interpretation version of PolSAR data. However, the pseudo-color image can be employed for PolSAR image classification by excavating color features, which are usually ignored [11]. Color features are global features that not only provide a good visual effect, but also describe the surface properties of the observed objects. Besides, they are also with rotation invariance and translation invariance. Most of the exiting methods choose polarimetric features to classify PolSAR images. However, for the pixels and regions which has similar polarimetric features, it is difficult to separate them correctly. Therefore, using color features can avoid the above problem. Uhlmann et al. [12] investigated the application of color features over the Pauli color-coded images in PolSAR image classification. They found that color features could provide useful information for PolSAR data understanding and analyses. Through a large number of experiments, Cheng et al. [13] found that color features are helpful for improving PolSAR image classification accuracy, especially for region-based classification methods. In this paper, superpixel generation and classification are based on PolSAR pseud-color image over Pauli decomposition by means of exploiting color features.

There are several classic PolSAR image classification approaches, such as support vector machine (SVM) [14,15], Wishart maximum likelihood (Wishart ML) [16], random forest (RF) [17], and so on. With the rapid development of deep learning and its extensive application in image classification, many deep networks, such as deep belief networks (DBNs), Auto-encoder, convolutional neural networks (CNN) are widely used in PolSAR image classification. Zhang et al. [18] introduced the local spatial information into the stacked sparse auto-encoder (SSAE) to learn the deep spatial sparse feature for PolSAR image classification. Chen et al. [19]. proposed a PolSAR image classification method based on multilayer projective dictionary pair learning (MDPL) and sparse auto-encoder (SAE). MDPL is used to extract features and SAE is used to get the nonlinear relationship between elements of feature vectors. Hou et al. [20] proposed a semi-supervised DBN PolSAR classification method. They reduce the dimension of features using multilinear principle component analysis (MPCA), and take these features as the input of DBN. Convolutional neural networks have become one of the most papular deep learning models due to their weight sharing, sparse connections. Zhou et al. [21] investigated the suitability and potential of deep convolutional neural network in PolSAR image classification. They found that hierarchical polarimetric spatial features can be learned automatically by deep neural network with two cascaded convolutional layers. Gao et al. [22] improved CNN model and proposed the dual-branch deep convolution neural network (Dual-CNN). One CNN is used to extract the polarization features and the other is used to extract the spatial features. To make use of the phase information of PolSAR image, Zhang et al. [23] proposed a complex-valued CNN (CV-CNN). They extend all elements of CNN to complex domain, including a complex backpropagation algorithm. Shang et al. [24] combined CV-CNN with auto-encoder network and proposed a complex-valued convolutional auto-encoder network (CV-CAE). However, most of the above studies did not adequately explore the role of color features in the PolSAR image classification.

Furthermore, for traditional CNN, each convolution kernel simply sums up the output features of each color channel, which causes the interrelationship among the channels to be ignored, and a greater degree of freedom to overfit [25]. In order to effectively utilize the correlation among the color channels, Zhu et al. [25] proposed the quaternion convolutional neural network (QCNN). They redesign the basic modules, such as convolution layer, full connected layer. Experimental results show that QCNN can obtain more representative features in color image processing tasks. Yin et al. [26] improved the QCNN and apply it to optical color image classification and forensics. For PolSAR image classification, using quaternion neural network (QNN) to extract deep features, some work has been carried out in Poincare-sphereparameter space [27–29]. Different from QNN, QCNN fully explore the correlation between color features. As far as we know, there has been no use of QCNN for PolSAR pseudo-color image classification. In this article, we will explore the application of QCNN to classify PolSAR pseudo-color image over the Pauli decomposition.

Otherwise, the classification methods for PolSAR images can be roughly divided into two categories: pixel-wise methods and region-based methods [11]. Pixel-wise approaches only consider the polarimetric features or other features of a single pixel, which makes the classification process easy and fast. And they also have the ability to preserve terrain details and edges, which are important in PolSAR image classification. Due to speckle noise, filters [30,31] are often adopted to preprocess the PolSAR images for pixel-wise methods. However, the speckle noise is not able to filtered out completely, and some distortion of shapes and texture happens [13]. For region-based methods, the spatial correlation between pixels in the local region can be effectively considered, which helps to improve PolSAR image classification results. One of the most widely used region-based approaches are superpixel-based approach. Superpixels are a number of non-overlapping regions satisfying homogeneity and uniqueness. The commonly used superpixel generation techniques are Normalized-Cuts(N-cut) [32], Meanshift [33], Turbopixel [34], simple linear iteration clustering (SLIC) [35]. Feng et al. [36] combined sparse representation-based classifier (SRCs) with superpixels to classify PolSAR images using polarimetric and contextual information. Hou et al. [37] generated superpixels based on Pauli decomposition and predicted class distributions of each pixel using multilayer autoencoders network. Wang et al. [38] constructed a 27-dimension polarimetric feature space, and classified PolSAR images using Laplacian Eigenmap (LE) and superpixels in this space. Zhang et al. [39] divided the PolSAR image into superpixels using a fast superpixel generation method, and classified the image by consensus similarity network fusion (CSNF). Ge et al. [40] combined improved SLIC and DBN to classify PolSAR images, which taked advantages of both pixel-wise and region-based methods. Gu et al. [41] generated superpixels for PolSAR images by SLIC and classified superpixels using deep CNN by constructing the input pyramid.

However, the superpixel generation methods in the above literatures are mostly based on statistical properties. It is difficult to separate the targets without strong boundary information. And the serious error of superpixel segmentation will lead to the decrease in classification accuracy. Tu et al. [42] realized that pixel affinities measure the likelihood of two neighboring pixels belonging to the same object. They also performed a lot of experiments to find that affinities derived from pre-trained deep features or deep edges are not sensitive to segmentation errors. So they proposed a pixel affinity net (PAN) to learn segmentation-aware affinities for graph-based superpixel segmentation. Abundant experiments in [42] indicated that the PAN by taking deep features and segmentation into account could generated better boundary-preserving superpixels compared to traditional techniques. PAN considers color images as input to learn deep features in color space. In this paper, we investigate superpixel generation of PolSAR pseudo-color images by PAN. Then, on the basis of superpixels, we perform PolSAR image classification with QCNN.

1.2. Contributions and Structure

In this paper, a supervised PolSAR image classification algorithm combining learned superpixels and QCNN is proposed. Three main steps are included: (1) pseudo-color image construction over the Pauli decomposition for PolSAR data; (2) superpixel generation of PolSAR pseudo-color image using PAN; (3) pseudo-color image classification by QCNN combined with superpixels. The framework of the proposed approach is shown in Figure 1. For the PolSAR pseudo-color image classification, QCNN is applied to exploit deep color features that considers the correlation among color channels. Besides, PAN is employed to learn superpixels and obtain reliable superpixel segmentation maps, which is benefit to improve the PolSAR pseudo-color image classification results.

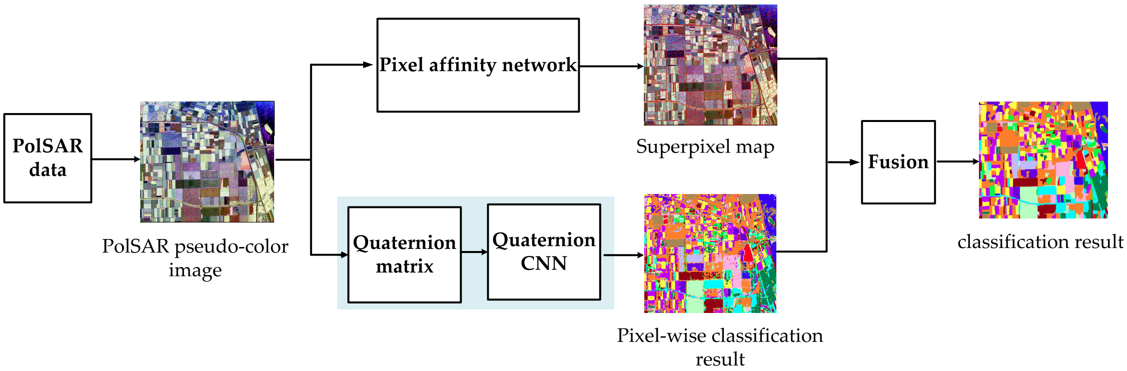


Figure 1. The pipline of the proposed approach. The idea of nearest neighbor (NN) is used to fuse the superpixel map and pixel-wise classification result, which will be described in Section 2.3.2 in detail.

The contributions of this work are as follows:

- Based on the PolSAR pseudo-color image, QCNN is utilized to extract deep color features and to classify the image. Different from traditional CNN processing each color channel independently, QCNN excavates deep color features based on a quaternion matrix constructed by the color channels. Therefore, the interaction among the channels can be fully considered, and the classification result is more reliable.
- A supervised superpixel generation technique of PolSAR pseudo-color images based on deep convolution network is implemented via segmentation-aware pixel affinity measurement. The method utilizes the auto-learned deep semantic features in color space to obtain better segmentation map than traditional techniques avoiding hand-craft features.
- Combining the pixel-wise classification map by QCNN and the superpixel generation map by PAN, a high-precision PolSAR classification map is obtained based on color features.

The remainder of this paper is organized as follows: In Section 2, we introduce the proposed approach in detail. In Section 3, the proposed approach is validated on three real PolSAR image data sets, and the experimental results are represented. In Sections 4 and 5, the discussion and conclusions are reported, respectively.

2. Methodology

2.1. Polarimetric Decomposition of PolSAR Data

The scattering matrix is a simple way to represent the scattering characteristics of PolSAR data. It contains all the polarimetric information, which reflects the relationship between the incident field vector and the scattered field vector sufficiently. A 2 × 2 scattering matrix of PolSAR data can be represented as:

$$\mathbf{S} = \begin{bmatrix} \mathrm{S_{hh}} & \mathrm{S_{hv}} \\ \mathrm{S_{vh}} & \mathrm{S_{vv}} \end{bmatrix} \tag{1}$$

where $\mathrm{S_{ij}}$, $\mathrm{ij} \in \{\mathrm{hh}, \mathrm{hv}, \mathrm{vh}, \mathrm{vv}\}$ is the complex scattering coefficient. The diagonal ones of S are co-polarization terms and the non-diagonal ones are cross-polarization terms. Under the condition that the transmitting antenna and the receiving antenna are interchangeable, and the electromagnetic wave propagates in a homogeneous medium, $\mathrm{S_{hv}} = \mathrm{S_{vh}}$.

The scattering matrix **S** can be decomposed into the weighted sums of Pauli bases by the Pauli decomposition, as shown in Equation (2), where each Pauli base matrix corresponds to a basic scattering mechanism.

$$\mathbf{S} = \frac{a}{\sqrt{2}}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{b}{\sqrt{2}}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} + \frac{c}{\sqrt{2}}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + \frac{d}{\sqrt{2}}\begin{bmatrix} 0 & -j \\ j & 0 \end{bmatrix} \tag{2}$$

where

$$a = \frac{\mathrm{S_{hh}} + \mathrm{S_{vv}}}{\sqrt{2}}, b = \frac{\mathrm{S_{hh}} - \mathrm{S_{vv}}}{\sqrt{2}}, c = \frac{\mathrm{S_{hv}} + \mathrm{S_{vh}}}{\sqrt{2}}, d = j\frac{\mathrm{S_{hv}} - \mathrm{S_{vh}}}{\sqrt{2}} \tag{3}$$

When $\mathrm{S_{hv}} = \mathrm{S_{vh}}$, $d = 0$. The PolSAR pseudo-color image can be constructed over the Pauli decomposition: $|b|^2$, $|c|^2$, $|a|^2$ are assigned to red, green and blue component, respectively [43].

2.2. Pixel-Wise Classification via Quaternion Convolutional Neural Network

Compared with CNN, QCNN considers the internal relationship among color channels and abstract more representative deep features. QCNN was proposed in [25] and has been successfully used to classify color images. The framework of QCNN in this paper is shown in Figure 2. A quaternion matrix constructed by the color channels is as the input of QCNN, and the probability distribution obtained by softmax is as the output. The QCNN model consists of two quaternion convolution layers, a quaternion fully connected layer and a real-value fully connected layer. The size of the former and the latter quaternion convolution kernel is 3 × 3 with 32 channels and 3 × 3 with 64 channels. One max pooling layer with size 2 × 2 is followed by each of the quaternion convolution layer. The quaternion fully connected layer is used to convert channels into 512. After that, the real-value fully connected layer with number of channels same as the number of categories is used to convert the output of the previous layer to a real value, which will be fed into softmax to make predictions.

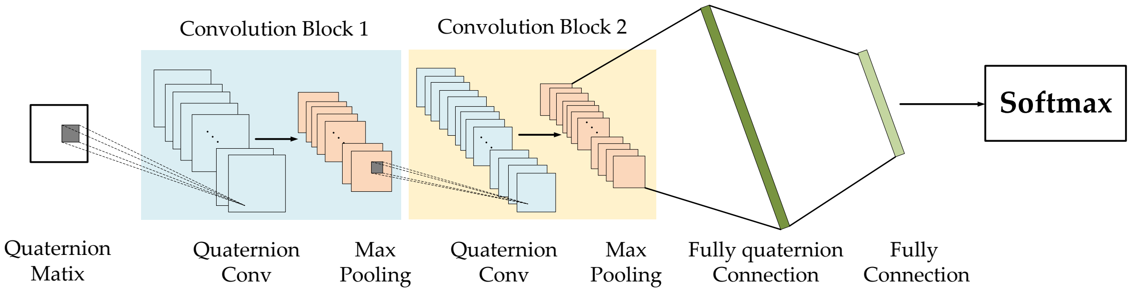


Figure 2. The framework of the QCNN.

2.2.1. Quaternion Matrix

The PolSAR pseudo-color image including three real-value color channels is constructed in Section 2.1. For a pixel, it corresponds to three color values (r, g, b), which can be converted into a quaternion value $\widehat{q} = q_0 + ri + gj + bk$, where i, j, k are the imaginary unit and $i^2 = j^2 = k^2 = ijk = -1$. For a PolSAR pseudo-color image, converting the three color values at each pixel to a quaternion value, a quaternion matrix is obtained: $\widehat{\mathbf{Q}} = [\widehat{q}_{mn'}] \in \mathbb{H}^{M \times N}$

$$\widehat{\mathbf{Q}} = \mathbf{Q_0} + \mathbf{R}i + \mathbf{G}j + \mathbf{B}k \tag{4}$$

where, $\mathbb{H}$ is the quaternion domain. $M \times N$ is the size of quaternion matrix. For a RGB image, $\mathbf{Q_0} = \mathbf{0}$. and **R**, **G**, **B** is the red, green, blue channel, respectively.

In order to understand quaternion convolution layer and quaternion fully connected layer better, some basic operations of the quaternion are briefly introduced. Define two quaternion $\widehat{q} = q_0 + q_1 i + q_2 j + q_3 k$ and $\widehat{p} = p_0 + p_1 i + p_2 j + p_3 k$,

- Addition: $\widehat{q} + \widehat{p} = (q_0 + p_0) + (q_1 + p_1)i + (q_2 + p_2)j + (q_3 + p_3)k$
- Scalar multiplication: $\lambda\widehat{q} = \lambda q_0 + \lambda q_1 i + \lambda q_2 j + \lambda q_3 k$
- Element multiplication: $\widehat{q}\widehat{p} = (q_0p_0 - q_1p_1 - q_2p_2 - q_2p_3) + (q_0p_1 + q_1p_0 + q_2p_3 - q_3p_2)i + (q_0p_2 - q_1p_3 + q_2p_0 + q_3p_1)j + (q_0p_3 + q_1p_2 - q_2p_1 + q_3p_0)k$
- Rotation: if $\widehat{q}$ need to be rotated, and the rotation angle is θ, rotary axis is $\mathrm{w} = [w_1\ w_2\ w_3]^{\mathrm{T}}$ and $w_1^2 + w_2^2 + w_3^2 = 1$. Then the rotated quaternion $\widehat{p} = \widehat{\mathrm{w}}\widehat{q}\widehat{\mathrm{w}}^*$, where $\widehat{\mathrm{w}} = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\,(w_1 i + w_2 j + w_3 k)$.

2.2.2. Quaternion Convolution Layers

Assume that the quaternion convolution operation is denoted as '$\circledast$', and the quaternion convolution kernel is expressed as $\widehat{\mathbf{W}} = [\widehat{\mathrm{w}}_{ll'}] \in \mathbb{H}^{L\times L}$, of which the element is set as:

$$\widehat{\mathrm{w}}_{ll'} = r_{ll'}(\cos\frac{\theta_{ll'}}{2} + \beta \sin\frac{\theta_{ll'}}{2}) \tag{5}$$

where, $\theta_{ll'} \in [-\pi, +\pi]$ is the rotation angle and $\beta = \frac{\sqrt{3}}{3}(i + j + k)$ is the rotary axis. $r_{ll'} \in \mathbb{R}$.

According to rotation operation, the quaternion convolution is defined as:

$$\widehat{\mathbf{Q}} \circledast \widehat{\mathbf{W}} = \widehat{\mathbf{F}} = \left[\widehat{\mathrm{f}}_{kk'}\right] \in \mathbb{H}^{(M-L+1)(N-L+1)} \tag{6}$$

where,

$$\widehat{\mathrm{f}}_{kk'} = \textstyle\sum_{l=1}^{L}\sum_{l'=1}^{L} \frac{1}{r_{ll'}} \widehat{\mathrm{w}}_{ll'} \widehat{q}_{(k+l)(k'+l')} \widehat{\mathrm{w}}_{ll'}^{*} \tag{7}$$

'$*$' is the conjugation of quaternion. For example, the conjugation of $\widehat{q} = q_0 + q_1 i + q_2 j + q_3 k$ is written as $\widehat{q}^* = q_0 - q_1 i - q_2 j - q_3 k$.

From Equation (7), the convolution operation can be seen as a series of rotations and scaling of a quaternion, which helps the QCNN to find the best representation of color features.

2.2.3. Quaternion Fully-Connected Layers

Assume that the input of the quaternion fully connected layer is $\widehat{\mathbf{a}} = [\widehat{\mathrm{a}}_n] \in \mathbb{H}^N$, $n = 1, 2, 3, \ldots, N$, and the kernel is $\widehat{\mathbf{V}}^m = [\widehat{\mathrm{v}}_n^m] \in \mathbb{H}^M$, $m = 1, 2, 3, \ldots, M$. Then the output of the quaternion fully connected layer can be expressed as

$$\widehat{\mathbf{O}} = \sum_{n=1}^{N} \frac{1}{s_n} \widehat{\mathrm{v}}_n^m \widehat{\mathrm{a}}_n \widehat{\mathrm{v}}_n^{m*} \tag{8}$$

where, s_n is the magnitude of $\widehat{\mathrm{v}}_n^m$.

2.3. Superpixel Generation Utilizing the Pixel Affinity Network

Many superpixel generation methods have been proposed for PolSAR images, but these methods mainly explore the statistical characteristics of PolSAR data based on the complex Gaussian distribution [44,45] or the non-Gaussian distributions [46–48]. And they are unsupervised methods. In this paper, a supervised deep learning method is adopted to generate superpixels for PolSAR pseudo-color image. PAN is used to learn superpixels with deep features. The framework of the PAN is shown in Figure 3.

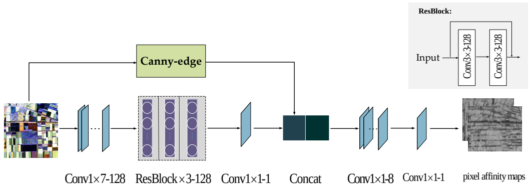


Figure 3. The framework of the PAN. The PAN is utilized to predict horizontal and vertical affinities at every pixel. The gray area in the upper right corner shows the standard structure of a ResBlock [49].

2.3.1. Pixel Affinity Network

Pixel affinities are used to measure the similarity between two adjacent pixels. A better pixel affinity map helps algorithm to generate a better superpixel map. For pixel (x, y), PAN predicts horizontal and vertical affinities. For example, there is a boundary-like change between pixel (x, y) and pixel $(x + 1, y)$, PAN will predict a smaller horizontal affinity value at pixel (x, y), otherwise, a larger horizontal affinity value is predicted. The vertical affinities are predicted in the same way.

As shown in Figure 3, a PolSAR pseudo-color image is feed into the PAN, and two affinity maps are output. One is the horizontal affinity map, and the other is the vertical affinity map. In the first layer, 1×7 convolution kernels (128 channels) are used to capture horizontal changes in the PolSAR pseudo-color image. To capture vertical changes, we only need to rotate the input image along the vertical axis. Three residual blocks (ResBlock) with 128 channels follow the first layer to make the net easier to optimize [49]. An intermediate affinity map predicted by a convolution kernel (1 channel) with size 1×1 is concatenated with an edge map calculated by the Canny detection operator. Canny detector is a powerful edge detector, that is widely used in computer vision. Here, we apply the Canny detector to obtain an accurate edge map. Then we concatenate the edge map with intermediate affinity map to provide an extra, more robust edge feature map into PAN. Experimental result proves that the edge map has been proven helpful for locating the boundaries [42]. Inspired by [50], feature-based method, such as corner detection, can be used to replace the Canny detector. Corner points are important features for feature extraction. The corner detection in an image is to find the points on the contour at which two straight edges meet at a particular angle or the location at which the direction of the contour changes significantly. As edge detection, corner detection can also provides a good feature map which contains the shape information with fewer points. The concatenated map is then passed onto two cascaded 1×1 convolution kernels (8 channels and 1 channel, respectively). Behind the two 1×1 convolution layers whose channel is 1, the sigmoid activations are used to restrict the affinity values in $0 - 1$. And ReLU activations are used behind each convolution layers.

A modified ERS is utilized to accept the horizontal and vertical affinity maps and generate superpixel map. The PAN is trained combining affinity maps, superpixel map with ground-truth segmentation map, the detail can be seen in [42]. The PAN is trained by optical color images. Hence, it is convictive to fine tune the pre-trained PAN with PolSAR data sets. However, the adopted PolSAR data in this work is the PolSAR pseudo-color images, the superpixel segmentation utilizes the color space information. That is to say, we regard the PolSAR pseudo-color images as the color images. Therefore, it is also reliable to generate superpixels of PolSAR pseudo-color images by the pre-trained PAN. In subsequent experiments, we compared the superpixel generation results by the pre-trained PAN and the fine-tuned PAN. The details are described in Section 3.

2.3.2. Pixel-Wise Classification Fusion Superpixels

In this work, we proposed a supervised PolSAR pseudo-color image classification approach fusion pixel-wise and region-based methods. QCNN is applied to obtain the pixel-wise classification map, and PAN is utilized to obtain region-based map, superpixels. By fusing the above maps, the final pixel-wise classification result is obtained. The pixel-wise approaches can learn the features of each pixel independently to preserve terrain details and edges. But They ignore the spacial relations among pixels, and is vulnerable to speckle noise. The region-based approach, such as superpixel, considers the spatial relations and restrains the influence of noise to a certain extent. Therefore, the approach fusion pixel-wise and region-based methods can reduce classification error and restrains noise. In this paper, QCNN is utilized as the pixel-wise approach and PAN is utilized to learn spacial relations among pixels. In principle, the correct class of the pixels in one superpixel should be the same. In other words, the label of a superpixel should be the label of class with the largest numbers or the second-largest numbers in this superpixel [37]. According to this theory, the classification result should be improved via fusing the superpixels generated by PAN and the pixel-wise classification map produced by QCNN. The fusion steps are as follows:

Step 1: For a PolSAR pseudo-color image, 10% reference pixels are selected for each class to calculate the mean intensity of each color channel $Mean(\mathbf{I}, \mathbf{L})$ where $\mathbf{I}$ is the intensity of one of the three color channels, and $\mathbf{L} \in labels$.

Step 2: For each superpixel, the total number of pixels $\boldsymbol{T}$ in this superpixel is counted. The number of pixels belonging to each category in this superpixel is also counted and arranged in descending order: $Num\_1, Num\_2, \ldots, Num\_c$, c is the number of classes in this superpixel.

Step 3: Find the maximum number Num_1 and the second largest number Num_2, and and record their corresponding classes $label_{Num\_1}$, $label_{Num\_2}$.

Step 4: Set the threshold D, According to experimental experience, it is satisfactory to set D as $1/3$. If $Num\_1 \geq \boldsymbol{T} \times D$, the label of the superpixel is $label_{Num\_1}$. Otherwise, the following steps are performed.

Step 5: The intensity of each color channel for the superpixel $Mean\_(I)$ is calculated and compared with corresponding intensity of class $label_{Num\_1}$, $label_{Num\_2}$,respectively:

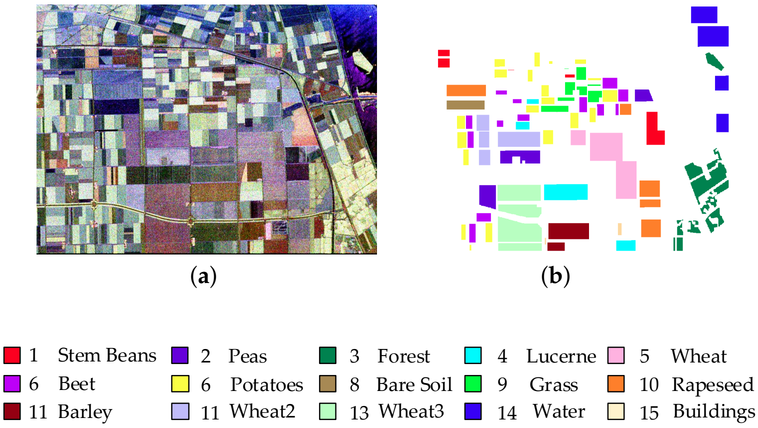
$$M(l) = \sum_{I} |Mean(I, l) - Mean\_(I)| \tag{9}$$

where, $l \in \{label_{Num\_1}, label_{Num\_2}\}$. If $M(label_{Num\_1}) \geq M(label_{Num\_2})$, the label of the superpixel is $label_{Num\_2}$. Otherwise, the label of the superpixel is $label_{Num\_1}$.

3. Experiment and Results

3.1. Description of the Experimental Datasets

The first data set is the four-look L-band PolSAR image with a resolution of $12m \times 6m$, Flevoland, which can be downloaded from https://earth.esa.int/web/polsarpro/data-sources/sample-datasets. The size of this data set is 750 × 1024 pixels. It contains 15 categories: stem beans, peas, forest, Lucerne, wheat, beet, potatoes, bare soil, grass, rapeseed, barley, wheat2, wheat3, water, buildings [22]. The Pauli color code image and its ground truth are given in Figure 4.

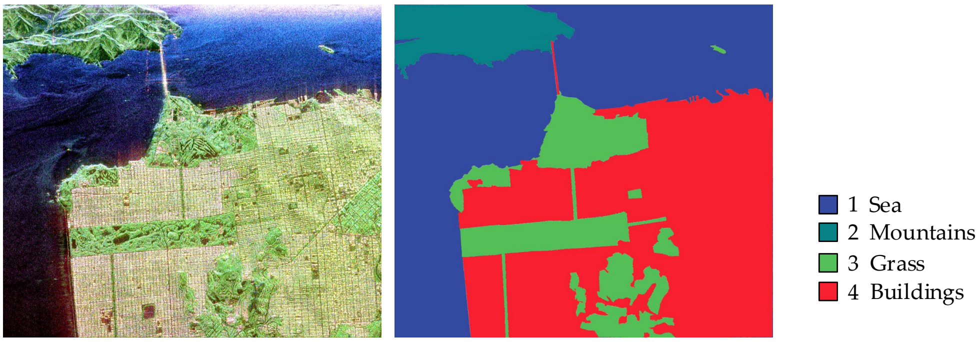


(**a**) (**b**)

(**c**)

Figure 4. (**a**) The Pauli color-code image of Flevoland data set. (**b**) The corresponding ground truth. (**c**) The legend of the ground truth.

The second data set named San Francisco is the four-look L-band PolSAR image with size 900 × 1024 pixels, which can be downloaded from https://earth.esa.int/web/polsarpro/data-sources/sample-datasets. It includes 4 categories: sea, mountains, grass and buildings [51]. The Pauli color code image and its ground truth are shown in Figure 5.



(**a**) (**b**) (**c**)

Figure 5. (**a**) The Pauli color-code image of San Francisco data set. (**b**) The corresponding ground truth. (**c**) The legend of the ground truth.

The third dataset, Oberpfaffenhofen, is the L-band PolSAR image with size 1300 × 1200 pixels. which can be downloaded from https://earth.esa.int/web/polsarpro/data-sources/sample-datasets. It consists of 3 classes: buildings, wood lands, open areas [22]. The Pauli-decomposed pseudo-color image and its ground truth are offered in Figure 6.

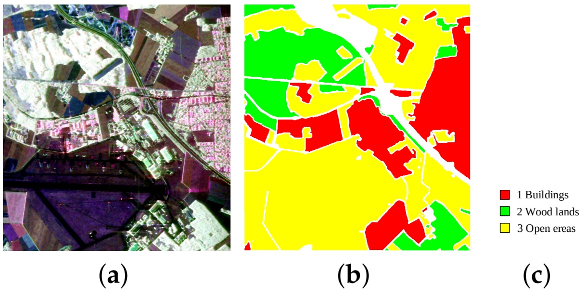


Figure 6. (**a**) The Pauli color-code image of Oberpfaffenhofen data set. (**b**) The corresponding ground truth. (**c**) The legend of the ground truth.

3.2. Superpixel Generation Results

In the experiments in this section, boundary recall (BR) shown in Equation (10) [52] and achievable segmentation accuracy (ASA) shown in Equation (11) [52] are utilized to evaluate the performances of the superpixel generation algorithms. A higher BR score indicates that superpixels maintains more real boundaries. And a higher ASA score indicates superpixels coincide better with real targets .

$$BR = \frac{\sum_{p \in \delta G} F_c(min_{q \in \delta S} \|p - q\| < \varepsilon)}{|\delta G|} \tag{10}$$

where, δS and δG represent the collection of pixels on the superpixel boundaries and real target boundaries, respectively. p is the p-th pixel in δG, and q is the q-th pixel in δS. F_c is the indication function utilized to check if the nearest neighbor pixels is within $\varepsilon = 2$ pixels of the distance, the standard form of F_c is *logical* function. Equation (10) denotes the probability of finding pixels on the superpixel boundaries within ε pixels of real target boundaries.

$$ASA = \frac{\sum_{Z'} max_{Z'} |S_{Z'} \cap G_Z|}{\sum_Z G_Z} \tag{11}$$

where, $S_{Z'}$ is the Z'-th superpixel and G_Z is the Z-th real target covering $S_{Z'}$ most. This equation computes the accuracy by labeling each superpixel with the label of ground truth segmentation that has the biggest overlap area.

Local regions of the above-mentioned three data sets are randomly selected for superpixel generation experiments. The selected region with 512 × 375 pixels of Flevoland is shown in Figure 7a. Figure 7c gives the selected region with 520 × 600 pixels of San Francisco data set. For Oberpfaffenhofen data set, a region with 500 × 500 pixels shown in Figure 7e is chosen.

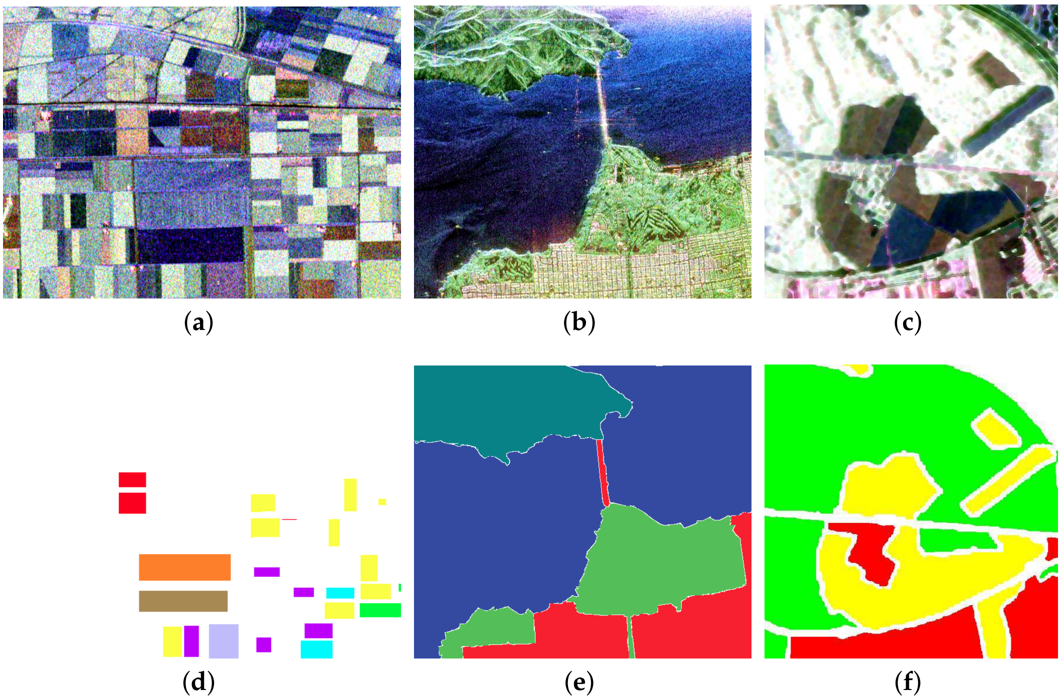


Figure 7. (**a**) The selected local region of Flevoland data set. (**b**) The selected local region of San Francisco data set. (**c**) The selected local region of Oberpfaffenhofen data set. (**d**) The ground truth of (**a**). (**e**) The ground truth of (**b**). (**f**) The ground truth of (**c**).

In this part, two means are adopted for PolSAR pseudo-color image superpixel generation to investigate which one is more suitable. One is to adopt the fine-tuned PAN with PolSAR pseudo-color images. This is no doubt the most reasonable method. The pre-trained PAN provided by [42] was trained with optical color images, but in this work, it is used for superpixel generation of PolSAR pseudo-color images. Because of the different characteristics of optical images and PolSAR data, it is more appropriate to adopt transfer learning when PAN is applied for PolSAR image superpixel generation. The other is to utilize the pre-trained PAN directly. In Section 2.1, PolSAR data is converted into pseudo-color image, superpixel generation of PolSAR is actually based on color features. From this point of view, it is analogous to superpixel generation of optical images. Hence, it is also reasonable to adopt pre-trained PAN for superpixel generation of PolSAR pseudo-color images without fine-tuning. Firstly, we perform superpixel generation by fine-tuned PAN. Since it is difficult and expensive to obtain large scale PolSAR data sets with ground truth, we adopt a 481 × 321 window to randomly sample a 1772 × 1100 PolSAR data, and manually screen and remove these data with higher similarity. By data augmentation, we obtained 500 pairs of PolSAR pseudo-color segmentation data sets for fine-tuning and validation. Figure 8 gives the sampled PolSAR image and its segmentation ground truth, which can be downloaded from https://earth.esa.int/web/polsarpro/data-sources/sample-datasets. 90% of them are randomly selected as the training data set, and the rest is the verification data set. During fine-tuning, the learning rate (LR) is set as 1×10^{-7}. And other parameters are equal to those as in [42]. For test, the data shown in Figure 7a is put to use. To investigate the effect of fine-tuning on the performance of PANs, we test the performances of the networks with different iterations. The ASA scores and the BR scores are shown in Figure 9a,b. From which, it can be illustrated that the performances of the fine-tuned PAN are hardly improved with the increase of iterations. Furthermore, the ASA scores and BR scores of the pre-trained PAN are always higher than those of fine-tuned PANs. That is to say, the best superpixel segmentation result is acquired by pre-trained PAN rather than by fine-tuned PANs. The reasons for this phenomenon will be analyzed and discussed in Section 4. Based on the above description, in the subsequent experiments, we only adopt the pre-trained PAN provided by [42] for superpixel generation of PolSAR pseudo-color images.

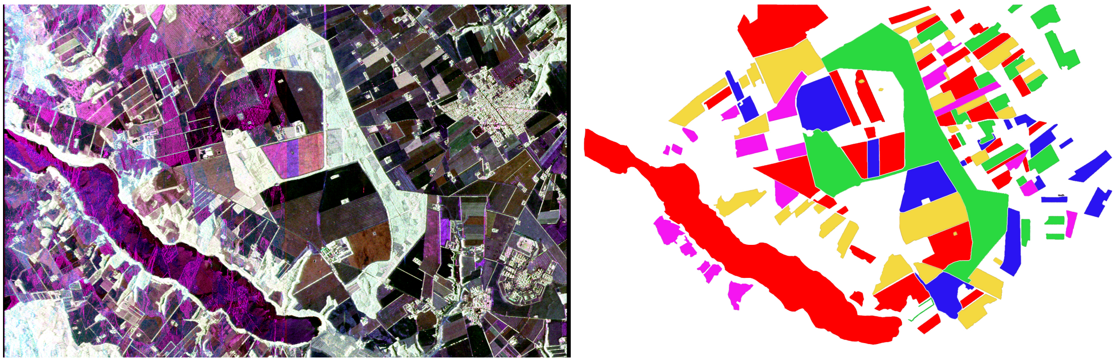
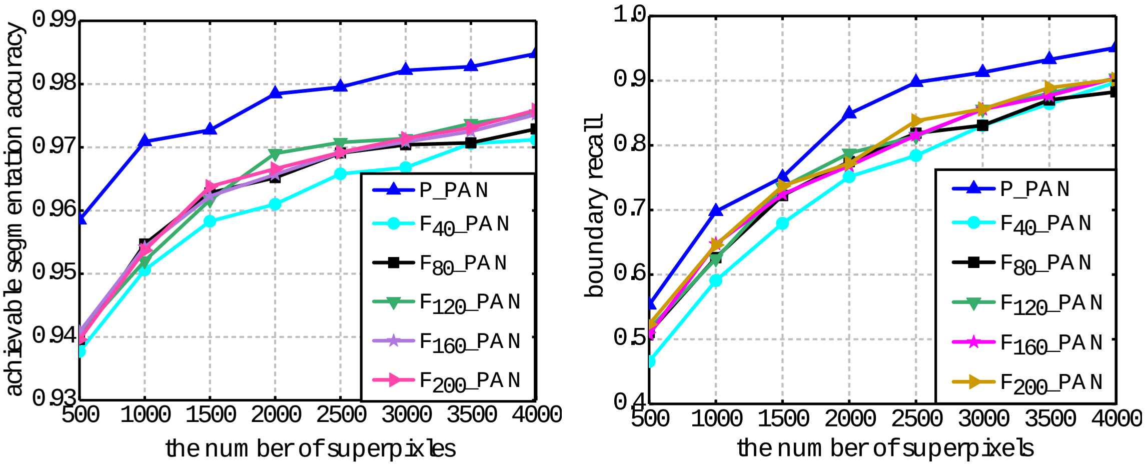

(**a**) (**b**)

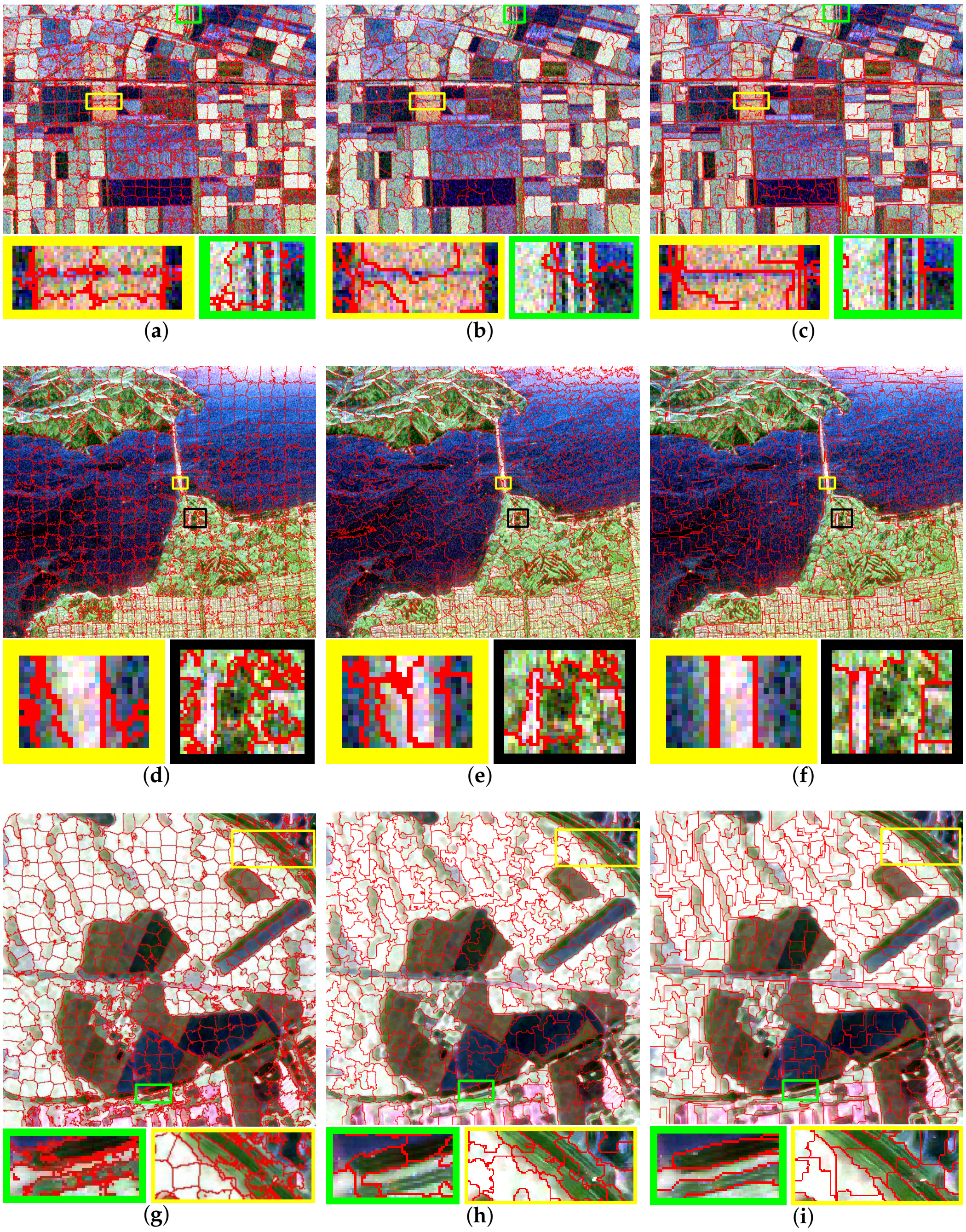
Figure 8. (**a**) The color image of Foulum data set which is used to get sampled data sets. (**b**) The corresponding segmentation ground truth.



(**a**) (**b**)

Figure 9. (**a**) The achievable segmentation accuracy under different iterations on Flevoland data set. (**b**) The boundary recall under different iterations on Flevoland data set. The subscript of F_N_PAN represents epoches.

To illustrate the superiority of deep learning in superpixel generation of PolSAR pseudo-color images, we compare the performances of the pre-trained PAN with SLIC and SEEDs. The visual superpixel generation maps are shown in Figure 10. From the magnified view of the region in box, we can see that the pre-trained PAN shows better boundary-preserving ability than SLIC and SEEDs for superpixel generation of PolSAR pseudo-color images, especially for some narrow, or smaller objectives. Figure 11 shows the ASA scores and the BR scores. For all the data sets, as the number of superpixels increases, the performance of each algorithm is gradually improved. However,the pre-trained PAN possesses the highest ASA scores and BR scores. For Flevoland data set, the BR score of our method improves to 94.8% at 4000 superpixels, which is about 17% and 15% higher than that of SEEDs and SLIC, respectively. For San Francisco data set, our method has the BR score, 94.76% at 4000 superpixels, which increases of 17% and 16% compared to SLIC and SEEDs. But for oberpfaffenhofen data set, due to simple terrain types, the ASA scores and BR scores of the three methods are very close to each other.

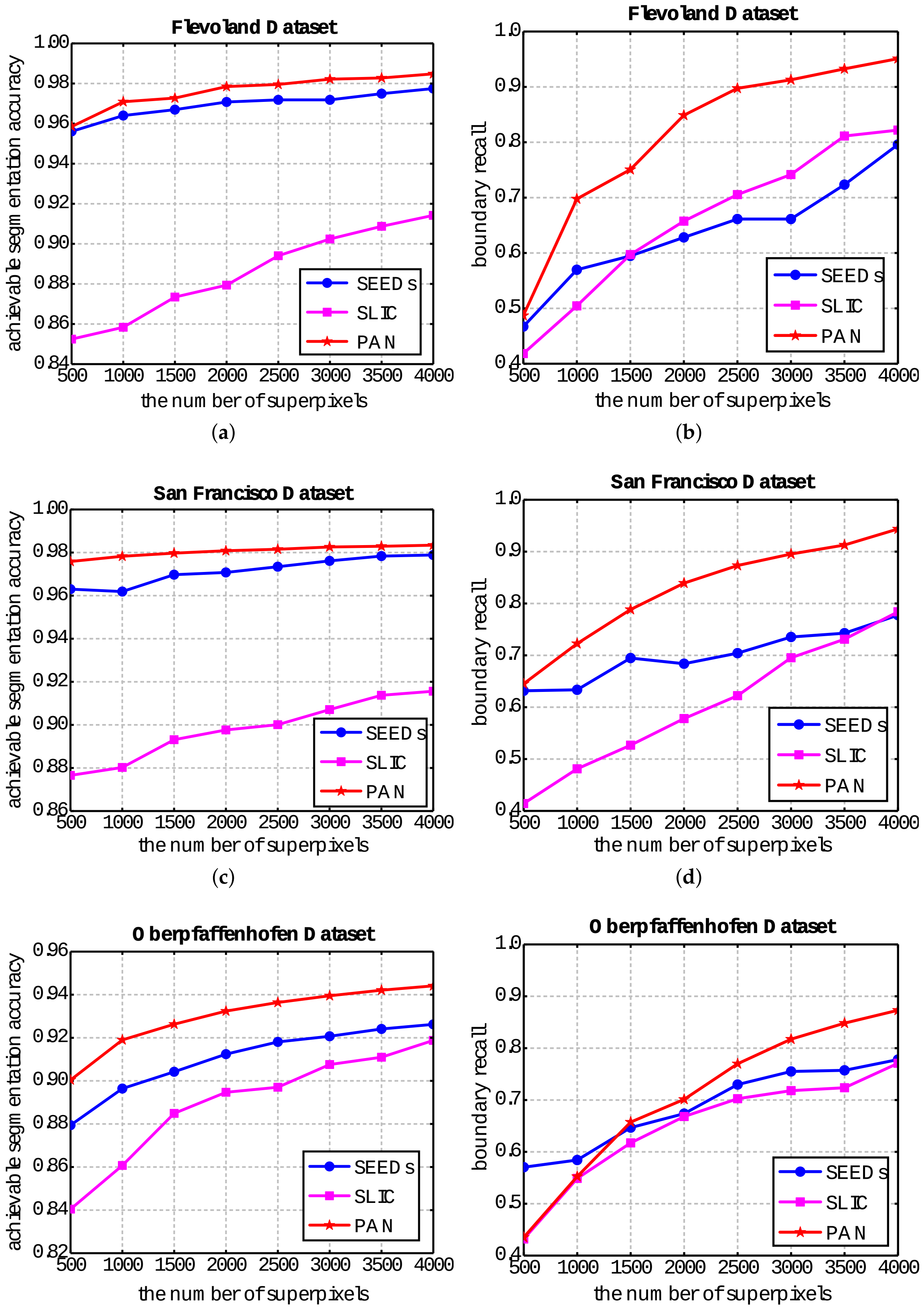


(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

(**g**) (**h**) (**i**)

Figure 10. The visual superpixel generation maps. (**a**) The superpixel generation map and partial enlarged details obtained by SLIC on Flevoland data set. (**b**) The superpixel generation map and partial enlarged details obtained by SEEDs on Flevoland data set. (**c**) The superpixel generation map and partial enlarged details obtained by PAN on Flevoland data set. (**d**) The superpixel generation map and partial enlarged details obtained by SLIC on San Francisco data set. (**e**) The superpixel generation map and partial enlarged details obtained by SEEDs on San Francisco data set. (**f**) The superpixel generation map and partial enlarged details obtained by PAN on San Francisco data set. (**g**) The superpixel generation map and partial enlarged details obtained by SLIC on Oberpfaffenhofen data set. (**h**) The superpixel generation map and partial enlarged details obtained by SEEDs on Oberpfaffenhofen data set. (**i**) The superpixel generation map and partial enlarged details obtained by PAN on Oberpfaffenhofen data set.



(**a**) (**b**)

(**c**) (**d**)

(**e**) (**f**)

Figure 11. The curve of achievable segmentation accuracy and boundary recall. (**a**) The achievable segmentation accuracy on Flevoland data set. (**b**) The boundary recall on Flevoland data set. (**c**) The achievable segmentation accuracy on San Francisco data set. (**d**) The boundary recall on San Francisco data set. (**e**) The achievable segmentation accuracy on Oberpfaffenhofen data set. (**f**) The boundary recall on Oberpfaffenhofen data set.

3.3. Classification Result

In this section, we first compare the classification performance of QCNN and CNN for PolSAR pseudo-color images. And we adopt the overall accuracy (OA) to evaluate the performance of the classification algorithms, and for each calss, precision and recall explained in Equation (12) are also calculated [53].

$$precision = \frac{TP}{TP+FP}, recall = \frac{TP}{TP+FN} \tag{12}$$

where, TP is the true positive or the number of correctly detected class points, FP is the false positive or the number of incorrectly detected class points, and FN is the false negative.

For both QCNN and CNN, a sliding window with size $m \times m$ is used to sample training data and validation data, which meets such requirement that the labels of the pixels within $n \times n$ region around the center pixel are the same as the label of the center pixel. To search the optimal m and n for each data set, we experiment in range $m \in \{4, 6, 8, 10, 12, 14\}$ and $n \in \{1, 2, \dots, m\}$. According to the experimental results, it can be concluded that the sliding window can not contain enough neighborhood information when m is small, which results in poor classification results. Besides, as m increases too much, OA increases slowly, but training time of the network increases rapidly. Balancing OA and training time, we choose relatively appropriate m and n for each data set considering their data characteristics. For all the three data set, m is set as 10. Flevoland contains 15 classes and each class is disperse, so n is set as 2. San Francisco and Oberpfaffenhofen has 4 or 3 classes and the distribution of each class is concentrated, n is set as 4. Besides, we also analyze the effect of sampling rate on classification results. For Flevoland, the number of pixels in each class varies greatly, for example, buildings has 476 pixels in total, while other classes have at least 3000 pixels. If the sampling rate is small, there is no guarantee that adequate training data is sampled for each category. But, if the sampling rate is too large, the training time will increase rapidly. Experiments show that it is appropriate to set the sampling rate as 10% for Flevoland data set, which can ensure to obtain enough training samples and reasonable training time. For San Francisco, the number of pixels in each class is large and the difference is also small. So it is enough to set as sampling rate as 5%. For Oberpfaffenhofen, the number of pixels in each class is huge, so it is sufficient to set the sampling rate as 1%. For each data set, 9% of the sampled data is randomly selected as training data, the remaining is validation data.

In this work, the compared CNN is structured by two convolutional layers, each of which is followed by a max pooling layer. And two fully connected layers are placed at the end of the network. The size and strides of the kernel, the number of neurons in each layer are the same as those of QCNN. During training, the loss functions of both nets are the cross entropy loss. For optimization algorithm, Root Mean Square Prop(RMSProp)is applied by QCNN, and the LR is set as 0.001, the weight decay (WD) is set as 10^{-6}. Moreover, Momentum optimizer is applied by CNN, and the LR is set as 0.001, the WD is set as 10^{-6}.

For Flevoland data set, CNN is trained with 31000 epochs, while QCNN is trained with 35 epochs. After training, a training accuracy, 99.78% and a validation accuracy, 99.53% are achieved for CNN. However, its test OA is only 76.77%. For QCNN, a training accuracy, 99.34% and a validation accuracy, 99.18% are acquired. And its test OA reaches 87.26%, which is 10.49% higher than that of CNN. The visual classification results of CNN and QCNN are shown in Figure 12. From Figure 13, which gives the classification accuracies for each class and the OA, we can see that the classification accuracies for each class of QCNN are higher than those of CNN. Especially for grass, the classification accuracy is 53.45%, which is 39.29% higher than that obtained by CNN. Figures 12 and 13 statistically demonstrate that QCNN is much better than CNN on Flevoland data set.

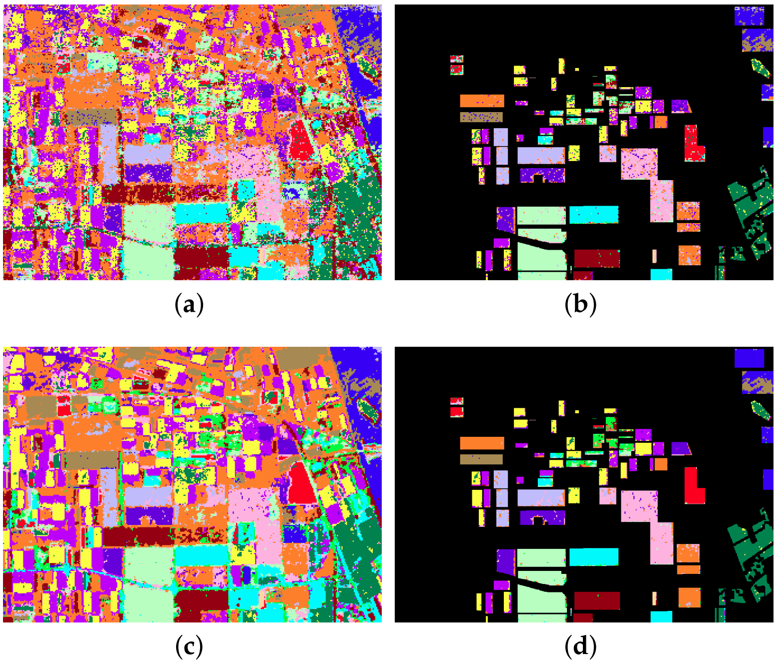


Figure 12. The visual classification results by CNN and QCNN on Flevoland data set. (**a**) The classification result obtained by CNN. (**b**) The result map (**a**) overlaid with the ground truth map. (**c**) The classification result obtained by QCNN. (**d**) The result map (**c**) overlaid with the ground truth map.

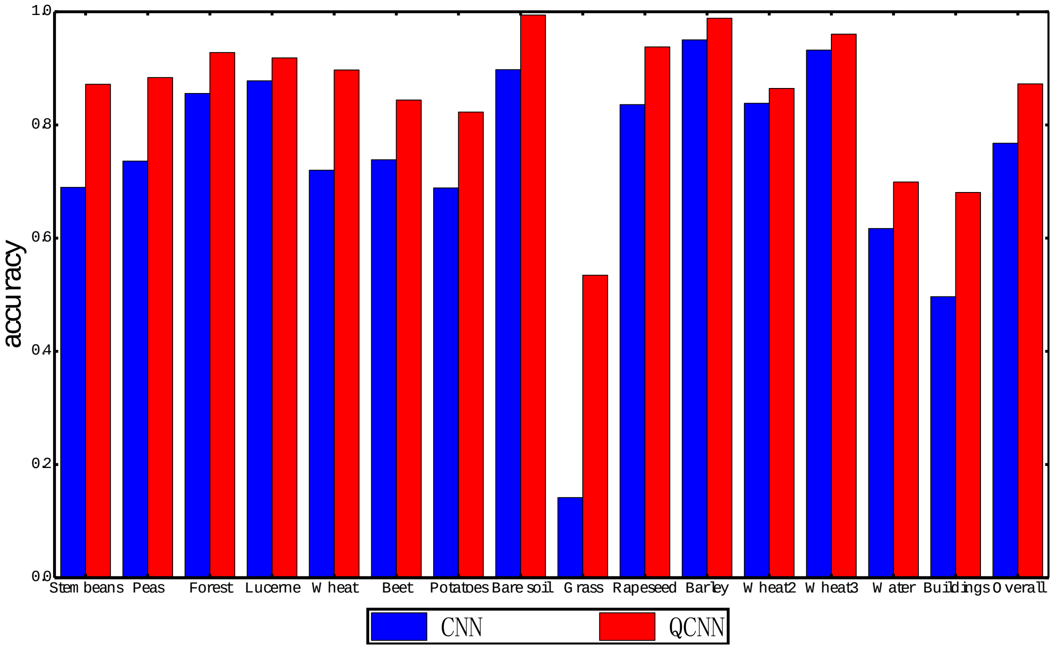


Figure 13. The classification accuracies by CNN and QCNN on Flevoland data set.

For San Francisco data set, we train QCNN with 35 epochs and CNN with 93000 epochs. Finally, a training accuracy, 99.29% and a validation accuracy, 99.05% are achieved for CNN, while the test OA is 85.44%. For QCNN, the training accuracy and the validation accuracy are 99.62% and 99.27%, respectively. And its test OA is 88.51%. The visual classification results of the two methods are shown in Figure 14. And the classification accuracies of the results are shown in Figure 15. From the figure, it can be seen that the classification accuracies of each class by QCNN is higher than those by CNN, except buildings. On the whole, the OA score of QCNN is 3.07% higher than that of CNN. This illustrates that QCNN has better performance.

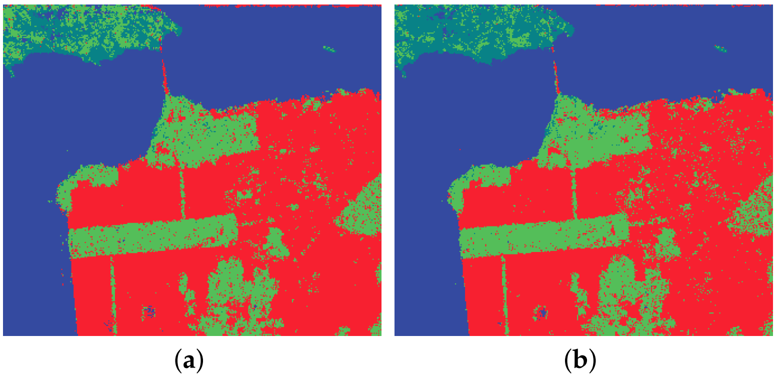


Figure 14. The visual classification results by CNN and QCNN on San Francisco data set. (**a**) The classification results by CNN. (**b**) The classification result by QCNN.

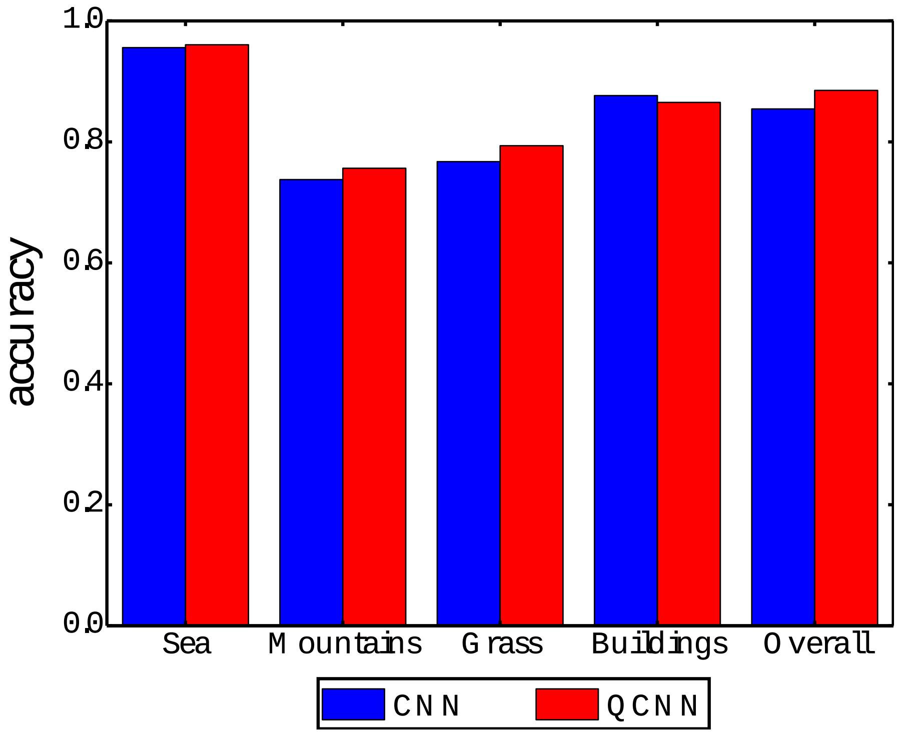
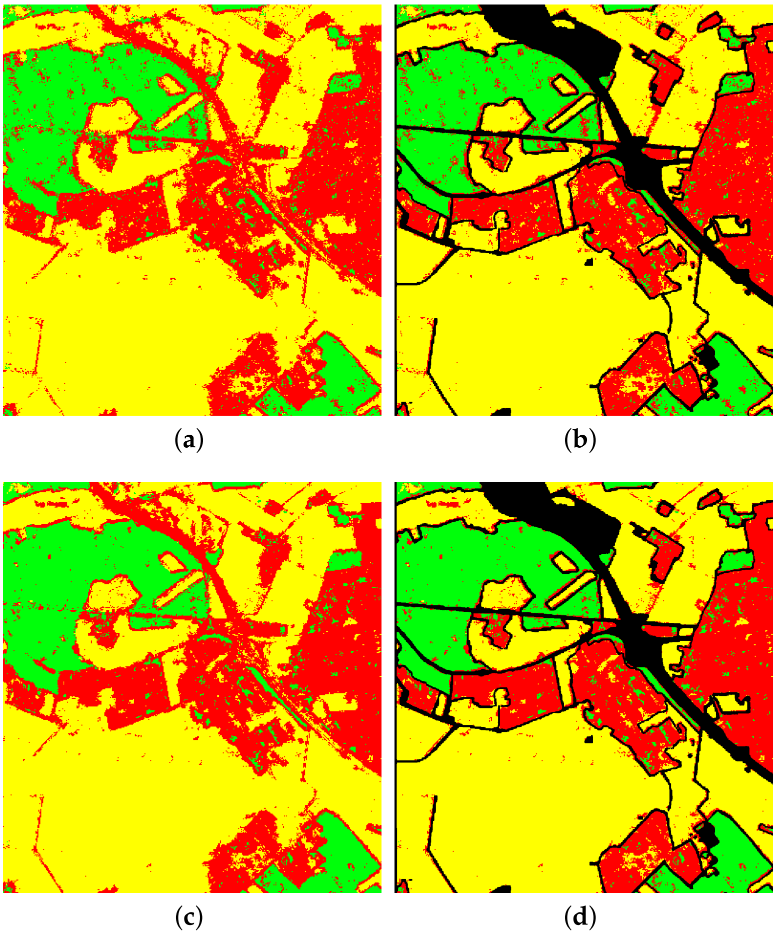


Figure 15. The classification accuracies by CNN and QCNN on San Francisco data set.

For Oberpfaffenhofen data set, QCNN and CNN are trained with 40 epochs and 93000 epochs, respectively. For CNN, the training accuracy and the validation accuracy are 99.38% and 99.66%, respectively. And its test OA is 90.06%. For QCNN, the training accuracy and the validation accuracy are 99.04% and 99.17%, respectively, while its OA is 92.58%. Figure 16 gives the visual classification results by the two methods. And Figure 17 shows the classification accuracies for each class and OA. With regard to the classification accuracies, the performances of QCNN in each class are better than CNN. Moreover, the total accuracy of QCNN is 2.52% higher than that of CNN. This also indicates that QCNN presents better classification results than CNN.



(**a**) (**b**)

(**c**) (**d**)

Figure 16. The visual classification results by CNN and QCNN on Oberpfaffenhofen data set. (**a**) The classification result obtained by CNN. (**b**) The result map (**a**) overlaids with the ground truth map. (**c**) The classification result obtained by QCNN. (**d**) The result map (**c**) overlaids with the ground truth map.

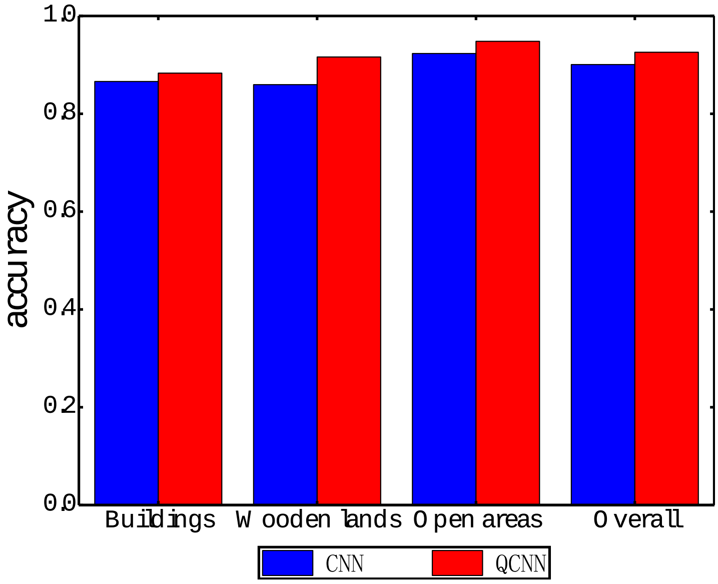


Figure 17. The classification accuracies by CNN and QCNN on Oberpfaffenhofen data set.

The foregoing experiments only represent the classification results by CNN or QCNN, but do not exploit the spatial neighborhood information among pixels. In order to take advantage of the spatial relationship among pixels in color space, the pixel-wise classification approach combined with superpixels is investigated in this part. The above experimental results show that the classification performance of QCNN is superior to CNN, so the subsequent experiments adopt QCNN as classifier.

As shown in [37], different superpixel numbers lead to different superpixel generation maps, which will impact the classification results. Therefore, we first investigate the impact of different superpixel numbers on the classification results. The OA scores of each data set with different superpixel numbers are shown in Figure 18. The optimal OA score, 96.56% is obtained at 1000 superpixels on Flevoland data set. Due to its simple category construction, San Francisco data set reaches a satisfactory OA score, 92.99% at 600 superpixels. Oberpfaffenhofen data set is a relatively large data with simple category construction, a considerable OA scores can be received at 800 superpixels.

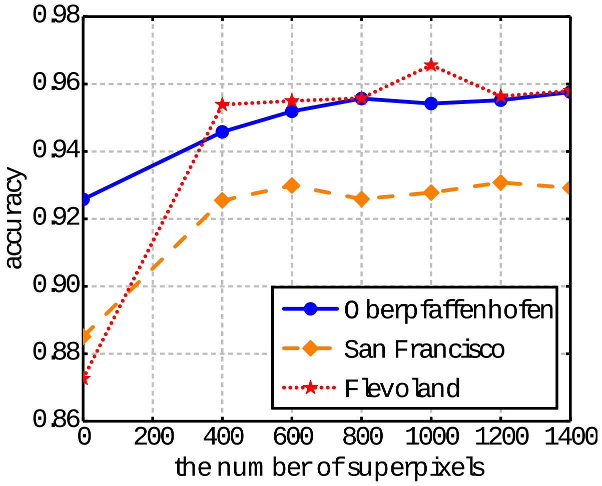


Figure 18. Accuracy curve of Flevoland, San Francisco and Oberpafaffenhofen data sets under different numbers of superpixels.

To further illustrate the effectiveness of the proposed approach, QCNN combined with pre-trained PAN (QPAN), QCNN combined with SLIC (QSLIC) and QCNN combined with SEEDs (QSEEDs) are compared. Refer to the experimental results described in last part, the number of superpixels are set as 1000, 600 and 800 for Flevoland, San Francisco and Oberpfaffenhofen, respectively. The classification maps are shown in Figures 19–21. And Tables 1–3 give the class-specific accuracy, overall accuracy, precision and recall.

From classification results of Flevoland data set in Table 1, QSLIC performs poorly with OA score, 89.83%. And it also gets the worst class-specific accuracy for each class. A better OA score, 93.98% is received by QSEEDs. Besides, the best class-specific accuracies of peas, Lucerne, wheat, grass and rapeseed are also obtained by QSEEDs. However, the best OA score, 95.56% is achieved by QPAN, which is 6.73% and 2.58% higher than that of QSLIC and QSEEDs, respectively. For most of the classes on Flevoland data set, QPAN can classify them correctly with a high accuracy. Especially for water, QPAN improves the classification accuracy to 89.25%. Besides, for each class, the proposed method can get better precision and recall. In summary, our approach performs better on complex terrain classification.

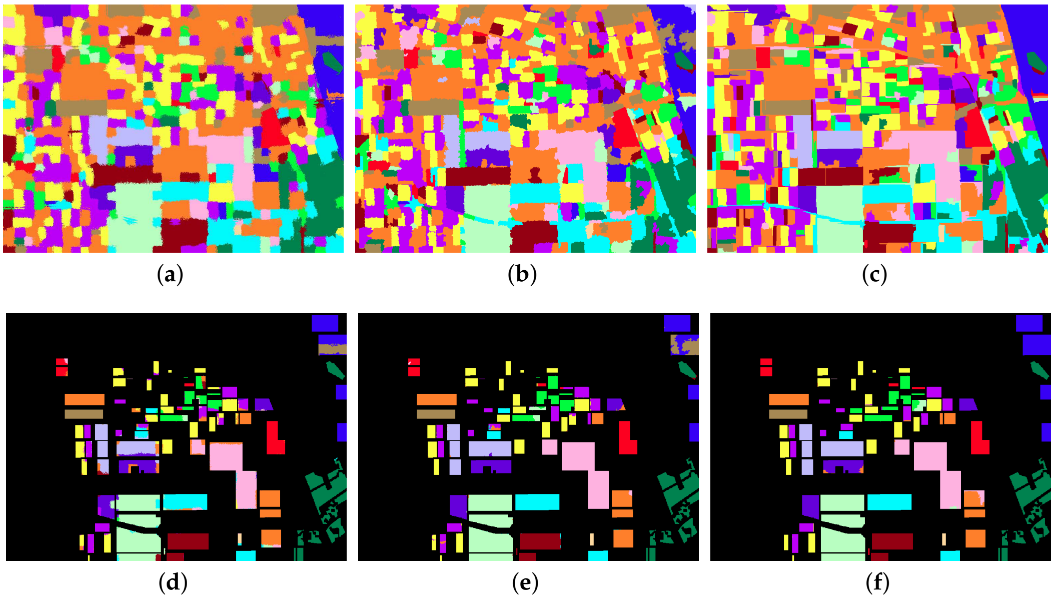


Figure 19. The visual classification results on Flevoland data set. (**a**) The classification map by QSLIC. (**b**) The classification map by QSEEDs. (**c**) The classification map by QPAN. (**d**) The classification map (**a**) overlaid with the ground truth map. (**e**) The classification map (**b**) overlaid with the ground truth map. (**f**) The classification map (**c**) overlaid with the ground truth map.

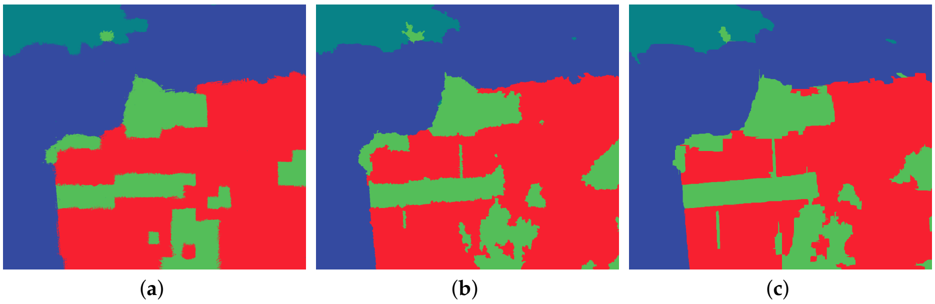


Figure 20. The visual classification results on San Francisco data set. (**a**) The classification map by QSLIC. (**b**) The classification map by QSEEDs. (**c**) The classification map by QPAN.

Table 1. The accuracy, precision and recall for Flevoland data set.

Class	**QSLIC**			**QSEEDs**			**The Proposed Method**		
	Accuracy	**Precision**	**Recall**	**Accuracy**	**Precision**	**Recall**	**Accuracy**	**Precision**	**Recall**
Stem Beans	0.9561	0.8996	0.9848	0.9843	0.9618	0.9956	0.9969	0.9515	0.9969
Peas	0.8499	0.7894	0.9015	0.9777	0.9705	0.9904	0.9661	1.0000	0.9661
Forest	0.9770	0.9978	0.9997	0.9757	0.9976	0.9998	0.9803	1.0000	0.9997
Lucerne	0.9024	0.8694	0.8077	0.9935	0.9995	0.9842	0.9407	0.9998	0.9465
Wheat	0.9123	0.9334	0.8874	0.9997	0.9998	0.9785	0.9897	0.9577	0.9897
Beet	0.8072	0.7648	0.8682	0.8943	0.8590	0.9788	0.9552	0.9902	0.9552
Potatoes	0.9442	0.9606	0.9228	0.9782	0.9867	0.9372	0.9784	0.9955	0.9784
Bare Soil	1.00	1.0000	0.8461	1.00	1.0000	0.8187	1.00	1.0000	1.0000
Grass	0.7996	0.6626	0.9784	0.8813	0.6886	0.9993	0.8695	0.9143	0.8695
Rapeseed	0.9631	0.9479	0.7133	0.9870	0.9832	0.9401	0.9426	0.9420	0.9426
Barley	0.9908	0.9972	0.9081	0.9874	0.9929	0.8506	0.9927	0.9989	1.0000
Wheat2	0.8290	0.8322	0.9516	0.9893	0.9900	0.9997	1.00	1.0000	1.0000
Wheat3	0.9411	0.9781	0.7936	0.9915	0.9990	0.8961	0.9934	0.9573	0.9996
Water	0.7206	0.9483	0.9981	0.6725	0.9303	1.0000	0.8925	1.0000	1.0000
Buildings	0.6450	0.0562	1.0000	0.8298	0.2442	1.0000	0.831	1.0000	0.8319
accuracy		0.8983			0.9398			0.9656	

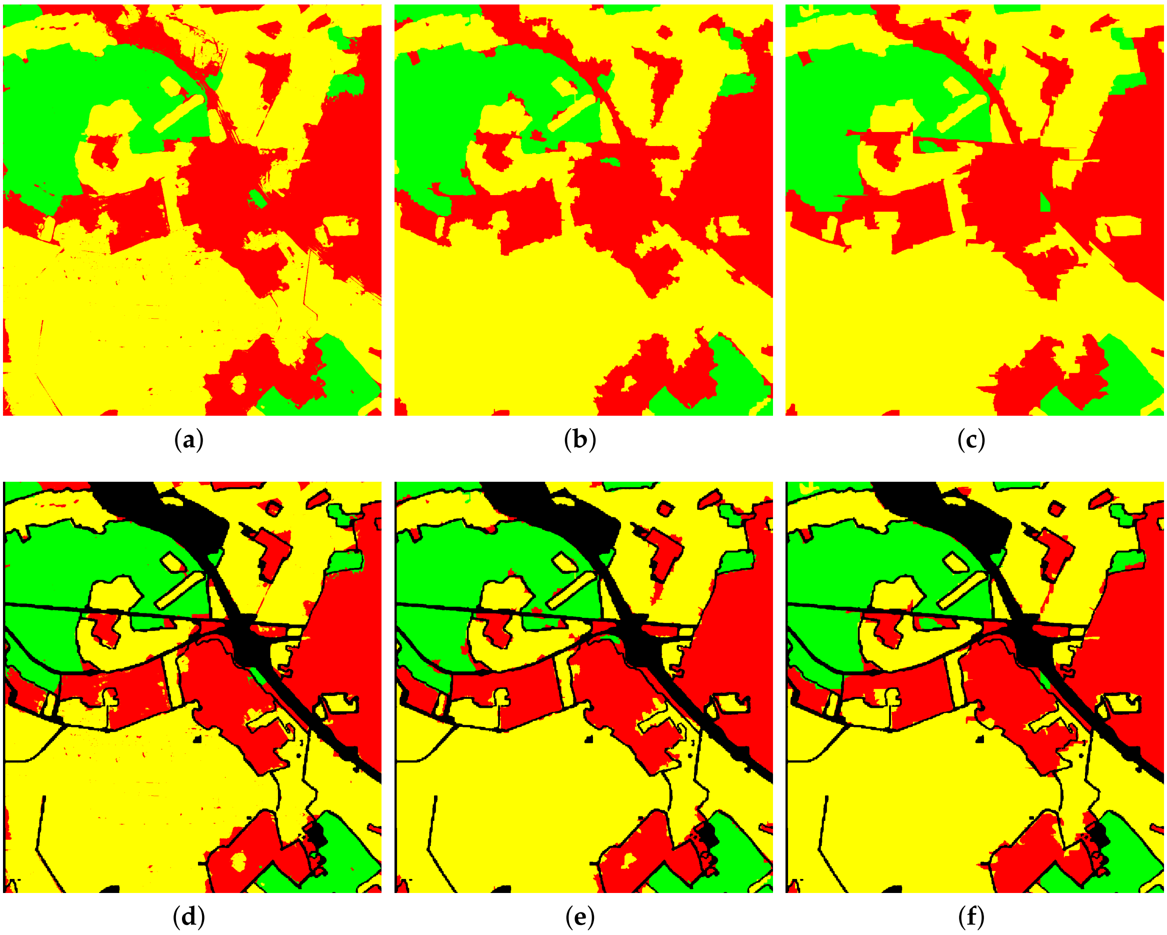


Figure 21. he visual classification results on Oberpfaffenhofen data set. (**a**) The classification map by QSLIC. (**b**) The classification map by QSEEDs. (**c**) The classification map by QPAN. (**d**) The classification map (**a**) overlaid with the ground truth map. (**e**) The classification map (**b**) overlaid with the ground truth map. (**f**) The classification map (**c**) overlaid with the ground truth map.

From classification results of San Francisco data set in Table 2, QSLIC still performs worst with lowest and class-specific accuracy. The OA score, 91.35% of QSEEDs is slightly worse than that of QPAN, 92.99%. However, QPAN can classify most of the classes correctly with acceptable classification accuracies, while other methods cannot. Comparing the precision and recall, the proposed method performs better.

Table 2. The accuracy, precision and recall for San Francisco data set.

Class	**QSLIC**			**QSEEDs**			**The Proposed Method**		
	Accuracy	**Precision**	**Recall**	**Accuracy**	**Precision**	**Recall**	**Accuracy**	**Precision**	**Recall**
Sea	0.9701	0.9925	0.9330	0.9705	0.9923	0.9559	0.9768	0.9967	0.9761
Mountains	0.9346	0.7557	0.9903	0.9333	0.7821	0.9922	0.9702	0.9072	0.9922
Grass	0.7898	0.5903	0.8939	0.8664	0.7302	0.8913	0.9060	0.8183	0.9025
Buildings	0.8876	0.9659	0.8361	0.8963	0.9721	0.8959	0.886	0.9680	0.9370
accuracy	0.9092			0.9135			0.9299		

Table 3 gives the classification results of Oberpfaffenhofen data set. It shows that acceptable OA scores can be obtained by both QSLIC and QSEEDs. However, the best OA score, 95.59% is still achieved by the proposed method, which is 2.13% and 1.24% higher than those acquired by QSLIC and SEEDs, respectively. Besides, the optimal classification accuracies of each class are still received by

QPAN. Due to simple terrain, the precision and recall obtained by our algorithm is not significantly higher than the other two methods.

Table 3. The accuracy, precision and recall for Oberpfaffenhofen data set.

Class	QSLIC			QSEEDs			The Proposed Method		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Buildings	0.9158	0.9241	0.9075	0.9312	0.9290	0.9318	0.9400	0.447	0.9335
Wood Lands	0.9235	0.9109	0.9940	0.9352	0.9139	0.9963	0.9339	0.9287	0.9969
Open Areas	0.9592	0.9858	0.9658	0.9561	0.9888	0.609	0.970	0.9910	0.9721
accuracy	0.9346			0.9435			0.9559		

Except for the above experiments, we also compare the proposed method with several state-of-the-art methods. Not all the compared methods used every data set utilized in our work, thus, the compared methods are not exactly the same for each data set. The compared approaches include classical algorithms, such as SVM [15], Wishart [16] and Mean shift [54], pixel-wised algorithms [21,51] and region-based algorithms [37,55,56]. Method in [21] applied two cascaded convolutional layers to learn hierarchical polarimetric spatial features for PolSAR image classification. Method in [37] combined SRCs with superpixels to classify PolSAR images. Method proposed in [51] adopted the nearest neighbor and SVM classifiers to classify PolSAR images based on the features extracted by tensor local discriminant embedding method. Method in [55] utilized Fuzzy superpixels to assist PolSAR image classification. Method employed in [56] classified PolSAR images with a deep neural network restrained by superpixels. The comparison results are shown in {Table 4. No matter which method is compared with, our method can always acquire the best OA. Therefore, the proposed method is a robust and competitive approach for PolSAR image classification.

Table 4. The comparison of the proposed method with several state-of-the-art methods.

Method	Flevoland	San Francisco	Oberpfaffenhofen
SVM [15]	0.8587	0.5608	0.8729
Wishart [16]	0.6177	0.6743	0.8050
Method in [21]	0.9246	0.9056	—
Method in [37]	—	—	0.9311
Method in [51]	—	0.8582	—
Method in [55]	0.8935	—	0.7725
Method in [56]	0.9267	—	0.9138
Method in [54]	—	0.8998	—
the proposed method	0.9656	0.9299	0.9559

4. Discussion

According to the forementioned experimental results, the proposed approach shows great advantages in PolSAR pseudo-color image classification. This mainly owes to two aspects. On the one hand, pre-trained PAN is adopted to extract the spatial relationship among pixels in color space. On the other hand, the interaction among color channels are fully considered by QCNN, which provides more representative color features. The following points in the experimental results are worth discussing.

- **The comparison of fine-tuned PAN and Pre-trained PAN.** Figure 9 show that the fine-tuned PAN with PolSAR segmentation data sets can not generate satisfactory superpixels. There are two main reasons for this phenomenon. One is that available PolSAR segmentation data sets are relatively inadequate to acquire a powerful fine-tuned network. So it is easy to over fit during fine tuning. The other is that the pre-trained PAN has been in a superior state to generate superpixels for color images. Therefore, the deep features extracted by pre-trained PAN in color space can be directly utilized to generate superpixels for PolSAR pseudo-color images.

- **The superiority of Segmentation-aware deep features.** The frequently-used superpixel generation techniques for PolSAR images use the statistical characteristics. However, the method adopted in this work utilize deep network to learn segmentation-aware deep features. Figures 10 and 11 indicate that the superpixels generated by our technique are better than those by SLIC and SEEDs. Especially for some narrow and small objects. That is to say, the segmentation-aware deep features extracted by pre-trained PAN is more effective for superpixel generation of PolSAR pseudo-color images.
- **The validity of interaction among color channels.** At the beginning of classification, the color channels of the PolSAR pseudo-color image are converted into a quaternion matrix, which will be fed into QCNN. Figures 12–17 show that QCNN can make full use of the interaction among color channels to find more expressive color features, which are more helpful for PolSAR pseudo-color image classification compared with the color features extracted by traditional CNN. On computation cost, the quaternion matrix operations are calculated during training QCNN, so it spends more time than CNN for one epoch. However, in our computer configuration environment, QCNN is trained with 30-40 epochs, which will be enough for each PolSAR data set. While CNN needs to be trained 8000-9000 epochs. Overall, QCNN takes less time than CNN. In conclusion, QCNN can extract more representative deep features considering interaction among color channels, which performs better in PolSAR classification than CNN.
- **Superpixels aided QCNN classification.** The experimental results indicate that the classification results by superpixels aided QCNN are better than those by QCNN. From Figure 18, OA increases faster when the number of superpixels is smaller than 6000, but as the number of superpixels is larger than 800, the OA increases slowly. There are some reasons for this phenomena. First, when the number of superpixels is small, there are a large number of pixels in each superpixel. In this condition, there is no guarantee that each superpixel completely belongs to one objective. Second, when the number of superpixels increases to 600–800, each superpixel must belong to only one objective. In other words, each target can be divided into several superpixels. Therefore, increasing the number of superpixels does not improve classification performance. For Flevoland dataset, there are 15 classes and each class occupies several small areas, so a relative large number of superpixels is needed to ensure the classification performance. Actually, the best OA is achieved at 1000 superpixels for Flevoland data set. For San Francisco and Oberpfaffenhofen data sets, there are only 4 or 3 classes and each class occupies a large area. Thus, a relative small number (600-800) superpixels can segment these two data sets greatly. Compared with other superpixel generation methods shown in Figures 19–21, the classification results based QPAN are the best. It is worth noting that, for San Francisco and Oberpfaffenhofen data sets, the OA based on the proposed method is around 2% higher than QSLIC and SEEDs. But for Flevoland, it is about 6% and 3% higher than QSLIC and QSEEDs, respectively. The reasons for such phenomenon are as follows. San Francisco and Oberpfaffenhofen include fewer classes with a lot of pixels. The OA has less fluctuation even if the number of misclassified pixels of each class is large. However, Flevoland includes 15 classes in total with around 3000 pixels of each class. When the number of incorrectly classified pixels increases, the OA has a large fluctuation. In this sense, the classification of Flevoland data set is more difficult, more complex, and also closer to reality. This shows the superiority of the proposed method in the complex classification task. Besides, it can be seen from the classification maps that the proposed method can correctly classify narrow and long terrains, while other methods can not. This attributes to the correct superpixel segmentation by PAN. In general, the method proposed in this paper has a greater advantage in PolSAR image classification in pixel level.

5. Conclusions

In this paper, a supervised classification method combining learned superpixels and QCNN is proposed for PolSAR pseudo-color images. Compared with several state-of-the-art methods, the proposed method is a very competitive method. Firstly, the deep texture and color features are extracted by PAN to generate reasonable superpixels, which well track the boundaries of object. Secondly, the interaction among the three color channels of the PolarSAR pseudo-color image are coded by QCNN, so the more representative color features are extracted which is helpful for PolSAR pseudo-color image classification. Finally, the final classification result is obtained by combining QCNN and superpixels. The proposed classification approach integrates both the color patterns and the spatial relationship of the pixels.

Author Contributions: X.Z. (Xinzheng Zhang) and J.X. conceived and designed the scheme. J.X. conducted experiments. X.Z. (Xinzheng Zhang) and J.X. analyzed and discussed the results. J.X. wrote the first draft. X.Z. (Xinzheng Zhang) and J.X. completed the revised paper. X.T., X.Z. (Xichuan Zhou) and T.W. gave some suggestions for the paper.

Funding: This research was funded by the National Science Foundation of China under Grants No. 61301224. This research was also funded by the Basic and Advanced Research Project in Chongqing under Grants No. cstc2017jcyjA1378 and No. cstc2016jcyjA0457..

Acknowledgments: Thanks to the reviewers and editors for their help with this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lee, J.-S.; Grunes, M.R.; Pottier, E. Quantitative comparison of classification capability: Fully polarimetric versus dual and single-polarization SAR. *IEEE Trans. Geosci. Remote Sens.* **2001**, *39*, 2343–2351.
2. Hara, Y.; Atkins, R.G.; Yueh, S.H.; Shin, R.T.; Kong, J.A. Application of neural networks to radar image classification. *IEEE Trans. Geosci. Remote Sens.* **1994**, *32*, 100–109. [CrossRef]
3. Chen, K.-S.; Huang, W.; Tsay, D.; Amar, F. Classification of multifrequency polarimetric SAR imagery using a dynamic learning neural network. *IEEE Trans. Geosci. Remote Sens.* **1996**, *34*, 814–820. [CrossRef]
4. Tzeng, Y.-C.; Chen, K.-S. A fuzzy neural network to SAR image classification. *IEEE Trans. Geosci. Remote Sens.* **1998**, *36*, 301–307. [CrossRef]
5. Del Frate, F.; Schiavon, G.; Solimini, D.; Borgeaud, M.; Hoekman, D.H.; Vissers, M.A. Crop classification using multiconfiguration C-band SAR data. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 1611–1619. [CrossRef]
6. Hoekman, D.H.; Vissers, M.A. A new polarimetric classification approach evaluated for agricultural crops. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 2881–2889. [CrossRef]
7. Chen, C.-T.; Chen, K.-S.; Lee, J.-S. The use of fully polarimetric information for the fuzzy neural classification of SAR images. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 2089–2100. [CrossRef]
8. Hänsch, R. Complex-Valued Multi-Layer Perceptrons—An Application to Polarimetric SAR Data. *Photogramm. Eng. Remote Sens.* **2010**, *76*, 1081–1088. [CrossRef]
9. Skriver, H.; Mattia, F.; Satalino, G.; Balenzano, A.; Pauwels, V.R.; Verhoest, N.E.; Davidson, M. Crop classification using short-revisit multitemporal SAR data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2011**, *4*, 423–431. [CrossRef]
10. Cloude, S.R.; Pottier, E. A review of target decomposition theorems in radar polarimetry. *IEEE Trans. Geosci. Remote Sens.* **1996**, *34*, 498–518. [CrossRef]
11. Xing, Y.; Zhang, Y.; Li, N.; Wang, R.; Hu, G. Improved superpixel-based polarimetric synthetic aperture radar image classification integrating color features. *J. Appl. Remote Sens.* **2016**, *10*, 026026. [CrossRef]
12. Uhlmann, S.; Kiranyaz, S. Integrating color features in polarimetric SAR image classification. *IEEE Trans. Geosci. Remote Sens.* **2013**, *52*, 2197–2216. [CrossRef]
13. Cheng, J.; Ji, Y.; Liu, H. Segmentation-based PolSAR image classification using visual features: RHLBP and color features. *Remote Sens.* **2015**, *7*, 6079–6106. [CrossRef]
14. Fukuda, S.; Hirosawa, H. Support vector machine classification of land cover: Application to polarimetric SAR data. In Proceedings of the IGARSS 2001 Scanning the Present and Resolving the Future, IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No. 01CH37217), Sydney, NSW, Australia, 9–13 July 2001; pp. 187–189.

15. Lardeux, C.; Frison, P.-L.; Tison, C.; Souyris, J.-C.; Stoll, B.; Fruneau, B.; Rudant, J.-P. Support vector machine for multifrequency SAR polarimetric data classification. *IEEE Trans. Geosci. Remote Sens.* **2009**, *47*, 4143–4152. [CrossRef]
16. Lee, J.-S.; Grunes, M.R.; Ainsworth, T.L.; Du, L.-J.; Schuler, D.L.; Cloude, S.R. Unsupervised classification using polarimetric decomposition and the complex Wishart classifier. *IEEE Trans. Geosci. Remote Sens.* **1999**, *37*, 2249–2258.
17. Loosvelt, L.; Peters, J.; Skriver, H.; De Baets, B.; Verhoest, N.E. Impact of reducing polarimetric SAR input on the uncertainty of crop classifications based on the random forests algorithm. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 4185–4200. [CrossRef]
18. Zhang, L.; Ma, W.; Zhang, D. Stacked sparse autoencoder in PolSAR data classification using local spatial information. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 1359–1363. [CrossRef]
19. Chen, Y.; Jiao, L.; Li, Y.; Zhao, J. Multilayer projective dictionary pair learning and sparse autoencoder for PolSAR image classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 6683–6694. [CrossRef]
20. Hou, B.; Guo, X.; Hou, W.; Wang, S.; Zhang, X.; Jiao, L. PolSAR Image Classification Based on DBN and Tensor Dimensionality Reduction. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Valencia, Spain, 22–27 July 2018; pp. 8448–8450.
21. Zhou, Y.; Wang, H.; Xu, F.; Jin, Y.-Q. Polarimetric SAR image classification using deep convolutional neural networks. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 1935–1939. [CrossRef]
22. Gao, F.; Huang, T.; Wang, J.; Sun, J.; Hussain, A.; Yang, E. Dual-branch deep convolution neural network for polarimetric SAR image classification. *Appl. Sci.* **2017**, *7*, 447. [CrossRef]
23. Zhang, Z.; Wang, H.; Xu, F.; Jin, Y.-Q. Complex-valued convolutional neural network and its application in polarimetric SAR image classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 7177–7188. [CrossRef]
24. Shang, R.; Wang, G.; A Okoth, M.; Jiao, L. Complex-Valued Convolutional Autoencoder and Spatial Pixel-Squares Refinement for Polarimetric SAR Image Classification. *Remote Sensi.* **2019**, *11*, 522. [CrossRef]
25. Zhu, X; Xu, Y; Xu, H. Quaternion Convolutional Neural Networks. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 631–647.
26. Yin, Q.; Wang, J.; Luo, X.; Zhai, J.; Jha, S.K.; Shi, Y.-Q. Quaternion Convolutional Neural Network for Color Image Classification and Forensics. *IEEE Access.* **2019**, *7*, 20293–20301. [CrossRef]
27. Shang, F.; Hirose, A. Quaternion neural-network-based PolSAR land classification in Poincare-sphere-parameter space. *IEEE Trans. Geosci. Remote Sens.* **2013**, *52*, 5693–5703. [CrossRef]
28. Kinugawa, K.; Shang, F.; Usami, N.; Hirose, A. Isotropization of Quaternion-Neural-Network-Based PolSAR Adaptive Land Classification in Poincare-Sphere Parameter Space. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 1234–1238. [CrossRef]
29. Kim, H.; Hirose, A. Unsupervised fine land classification using quaternion autoencoder-based polarization feature extraction and self-organizing mapping. *IEEE Trans. Geosci. Remote Sens.* **2017**, *56*, 1839–1851. [CrossRef]
30. Lee, J.-S.; Grunes, M.R.; Mango, S.A. Speckle reduction in multipolarization, multifrequency SAR imagery. *IEEE Trans. Geosci. Remote Sens.* **1991**, *29*, 535–544. [CrossRef]
31. Lee, J.-S.; Grunes, M.R.; De Grandi, G. Polarimetric SAR speckle filtering and its implication for classification. *IEEE Trans. Geosci. Remote Sens.* **1999**, *37*, 2363–2373.
32. Shi, J.; Malik, J. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 888–905.
33. Comaniciu, D.; Meer, P. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *5*, 603–619. [CrossRef]
34. Levinshtein, A.; Stere, A.; Kutulakos, K.N.; Fleet, D.J.; Dickinson, S.J.; Siddiqi, K. Turbopixels: Fast superpixels using geometric flows. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 2290–2297. [CrossRef] [PubMed]
35. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Süsstrunk, S. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2274–2282. [CrossRef] [PubMed]
36. Feng, J.; Cao, Z.; Pi, Y. Polarimetric contextual classification of PolSAR images using sparse representation and superpixels. *Remote Sens.* **2014**, *6*, 7158–7181. [CrossRef]

37. Hou, B.; Kou, H.; Jiao, L. Classification of polarimetric SAR images using multilayer autoencoders and superpixels. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, *9*, 3072–3081. [CrossRef]
38. Wang, H.; Han, J.; Deng, Y. PolSAR image classification based on Laplacian Eigenmaps and superpixels. *EURASIP J. Wirel. Commun. Netw.* **2017**, *2017*, 198. [CrossRef]
39. Zhang, Y.; Zou, H.; Shao, N.; Zhou, S.; Ji, K. Unsupervised classification of polsar imagery based on consensus similarity network fusion. In Proceedings of IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 3266–3269.
40. Ge, S.; Lu, J.; Gu, H.; Yuan, Z.; Su, W. Polarimetrie SAR image classification based on deep belief network and superpixel segmentation. In Proceedings of the 2017 3rd International Conference on Frontiers of Signal Processing (ICFSP), Paris, France, 6–8 September 2017; pp. 114–119.
41. Gu, F.; Zhang, H.; Wang, C. A Classification Method for Polsar Images using SLIC Superpixel Segmentation and Deep Convolution Neural Network. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Valencia, Spain, 22–27 July 2018; pp. 6671–6674.
42. Tu, W.-C.; Liu, M.-Y.; Jampani, V.; Sun, D.; Chien, S.-Y.; Yang, M.-H.; Kautz, J. Learning superpixels with segmentation-aware affinity loss. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 568–576.
43. Lee, J.-S.; Pottier, E. *Polarimetric Radar Imaging: From Basics to Applications*; CRC Press: Rochester, NY, USA, 2009.
44. Cao, F.; Hong, W.; Wu, Y.; Pottier, E. An Unsupervised Segmentation With an Adaptive Number of Clusters Using the $SPAN/H/\alpha/A$ Space and the Complex Wishart Clustering for Fully Polarimetric SAR Data Analysis. *IEEE Trans. Geosci. Remote Sens.* **2007**, *45*, 3454–3467. [CrossRef]
45. Wu, Y.; Ji, K.; Yu, W.; Su, Y. Region-based classification of polarimetric SAR images using Wishart MRF. *IEEE Geosci. Remote Sens. Lett.* **2008**, *5*, 668–672. [CrossRef]
46. Freitas, C.C.; Frery, A.C.; Correia, A.H. The polarimetric $\mathcal{G}$ distribution for SAR data analysis. *Environmetrics* **2005**, *16*, 13–31. [CrossRef]
47. Doulgeris, A.P.; Anfinsen, S.N.; Eltoft, T. Classification with a non-Gaussian model for PolSAR data. *IEEE Trans. Geosci. Remote Sens.* **2008**, *46*, 2999–3009. [CrossRef]
48. Doulgeris, A.P. An Automatic $\mathcal{U}$-Distribution and Markov Random Field Segmentation Algorithm for PolSAR Images. *IEEE Trans. Geosci. Remote Sens.* **2014**, *53*, 1819–1827. [CrossRef]
49. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
50. Kahaki, S.M.M.; Nordin, M.J.; Ashtari, A.H. Contour-Based Corner Detection and Classification by Using Mean Projection Transform. *Sensors* **2014**, *14*, 4126–4143. [CrossRef] [PubMed]
51. Huang, X.; Qiao, H.; Zhang, B.; Nie, X. Supervised polarimetric SAR image classification using tensor local discriminant embedding. *IEEE Trans. Image Process.* **2018**, *27*, 2966–2979. [CrossRef] [PubMed]
52. Shen, J.; Du, Y.; Wang, W.; Li, X. Lazy random walks for superpixel segmentation. *IEEE Trans. Image Process.* **2014**, *23*, 1451–1462. [CrossRef] [PubMed]
53. Kahaki, S.M.M.; Nordin, M.J.; Ashtari, A.H.; Zahra, S.J. Invariant Feature Matching for Image Registration Application Based on New Dissimilarity of Spatial Features. *PLoS ONE* **2016**, *11*, e0149710.
54. Beaulieu, J.-M.; Touzi, R. Mean-shift and hierarchical clustering for textured polarimetric SAR image segmentation/classification. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Honolulu, HI, USA, 25–30 July 2010; pp. 2519–2522.
55. Guo, Y.; Jiao, L.; Wang, S.; Wang, S.; Liu, F.; Hua, W. Fuzzy superpixels for polarimetric SAR images classification. *IEEE Trans. Fuzzy Syst.* **2018**, *26*, 2846–2860. [CrossRef]
56. Geng, J.; Ma, X.; Fan, J.; Wang, H. Semisupervised classification of polarimetric SAR image via superpixel restrained deep neural network. *IEEE Geosci. Remote Sens. Lett.* **2017**, *15*, 122–126. [CrossRef]