



## Article

# PolSAR Image Classification with Active Complex-Valued Convolutional-Wavelet Neural Network and Markov Random Fields

Lu Liu <sup>1,\*</sup> and Yongxiang Li <sup>2</sup>

<sup>1</sup> Qian Xuesen Laboratory of Space Technology, China Academy of Space Technology, No. 104 Youyi Road, Beijing 100094, China

<sup>2</sup> Aerospace Information Research Institute, Chinese Academy of Sciences, No. 9 Dengzhuang South Road, Beijing 100094, China; liyx@aircas.ac.cn

\* Correspondence: lluiullui@126.com

**Abstract:** PolSAR image classification has attracted extensive significant research in recent decades. Aiming at improving PolSAR classification performance with speckle noise, this paper proposes an active complex-valued convolutional-wavelet neural network by incorporating dual-tree complex wavelet transform (DT-CWT) and Markov random field (MRF). In this approach, DT-CWT is introduced into the complex-valued convolutional neural network to suppress the speckle noise of PolSAR images and maintain the structures of learned feature maps. In addition, by applying active learning (AL), we iteratively select the most informative unlabeled training samples of PolSAR datasets. Moreover, MRF is utilized to obtain spatial local correlation information, which has been proven to be effective in improving classification performance. The experimental results on three benchmark PolSAR datasets demonstrate that the proposed method can achieve a significant classification performance gain in terms of its effectiveness and robustness beyond some state-of-the-art deep learning methods.

**Keywords:** PolSAR classification; complex-valued convolutional-wavelet neural network (CV-CWNN); dual-tree complex wavelet transform (DT-CWT); active learning (AL); Markov random field (MRF)



**Citation:** Liu, L.; Li, Y. PolSAR Image Classification with Active Complex-Valued Convolutional-Wavelet Neural Network and Markov Random Fields. *Remote Sens.* **2024**, *16*, 1094. <https://doi.org/10.3390/rs16061094>

Academic Editor: Massimiliano Pieraccini

Received: 23 January 2024

Revised: 12 March 2024

Accepted: 12 March 2024

Published: 20 March 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Recently, polarimetric synthetic aperture radar (PolSAR) [1,2] has been able to capture the polarimetric characteristics of ground targets in addition to their scattering echo amplitude, phase, and frequency characteristics; it is one of the most important remote sensing types and more sophisticated than the traditional SAR system. PolSAR obtains the target polarization scattering matrix by sending and receiving electromagnetic waves with different polarization modes in order to measure the ground target's polarization scattering properties [3]. Terrain classification using PolSAR image data has become a very challenging interpretation task due to limited labeled samples, unexploited spatial information, and speckle noise. PolSAR terrain classification, which focuses on assigning a meaningful category to each pixel vector according to PolSAR image contents, has wide application prospects, such as sea monitoring, geological resource exploration, agriculture status assessment, navigation safety, topographic mapping, etc. [4–7].

In the last few decades, PolSAR classification has been carried out with a wide variety of machine learning methods when limited numbers of labeled samples are obtainable. Kong et al. [8] developed a Bayesian classification approach to terrain cover identification using polarimetric data. Tao et al. [9] incorporated spatial and feature information into the k-nearest neighbor (KNN) classifier for PolSAR classification. Lardeux et al. [10] successfully applied SVM for Multifrequency PolSAR classification. Ersahin et al. [11] proposed a spectral graph clustering technique to improve performance in the classification

of PolSAR images. Bi et al. [12] employed discriminative clustering for unsupervised PolSAR image classification. Antropov et al. [13] adopted a probabilistic neural network (PNN) for land cover mapping using spaceborne PolSAR image data. Compared with the polarimetric decomposition technique, machine learning PolSAR classification can usually make classification performance more robust and accurate.

Recently, a group of deep learning methods for PolSAR classification have been discussed in the literature and are demonstrated to be powerful for acquiring deep representations and discriminative features from PolSAR data. Zhou et al. [14] used a deep convolutional neural network for supervised multilooked POLSAR classification. Liu et al. [15] suggested a Wishart deep belief network (W-DBN) for PolSAR classification by making full use of local spatial information. Jiao and Liu [16] designed a Wishart deep stacking network (W-DSN) for fast PolSAR classification. Zhang et al. [17] presented a complex-valued CNN (CV-CNN) for PolSAR classification by employing both amplitude and phase information. Bi et al. [18] proposed a graph-based semisupervised deep neural network for PolSAR image classification. However, the success of these deep learning methods for PolSAR classification depends on massive amounts of labeled samples, as we know that PolSAR-annotated data is difficult to obtain in actual cases since labeling samples is time-consuming and labor-intensive. Additionally, speckle noise can easily affect the classification performance of PolSAR image data. Therefore, we will essentially develop a robust PolSAR classification technique that overcomes limited labeled samples and speckle noise by integrating phase and spatial information.

To tackle the problem of limited labeled samples, active learning (AL) is a powerful strategy for PolSAR classification since it can choose the most valuable samples of unlabeled data for annotation. AL has been widely applied in PolSAR and hyperspectral remote sensing images. Tuia et al. [19] proposed two AL methods for the semiautomatic definition of training samples for remote sensing image classification and applied them and SVM to the classification of VHR optical imagery and hyperspectral data. Tuia et al. [20] introduced a survey of AL for solving supervised remote sensing image classification tasks. In the literature, AL approaches concentrate on establishing new training sets by iteratively enhancing classification performance through sampling. Samat et al. [21] presented an AL ELM algorithm with fast operation and strong generalization for quad-polarimetric SAR classification. Liu et al. [22] proposed an active object-based classification algorithm with random forest for PolSAR imagery to enhance classification performance. Bi et al. [23] integrated AL and a fine-tuned CNN to improve the performance of supervised PolSAR image classification. Additionally, Li et al. [24] applied AL and loopy belief propagation to exploit the spectral and the spatial information of hyperspectral image data. Cao et al. [25] adopted deep learning and AL methods in a unified framework for hyperspectral image classification. Thus, AL is an effective method to decrease the cost of obtaining large labeled training samples for PolSAR image classification using deep learning methods.

As for speckle noise, CNN-based and CV-CNN-based methods for PolSAR image classification cannot effectively balance classification efficiency and classification performance. The wavelet transform- and complex wavelet transform-constrained pooling layers in CNN and CV-CNN can remove the speckle noise and have a better performance in PolSAR image classification. Additionally, wavelet transform domain-based CNN and CV-CNN can preserve PolSAR image edges, sharpness, and some other structures, while CNN and CV-CNN with conventional pooling have not yet taken into consideration the structure of the previous layer. Consequently, a rising number of wavelet transform domain-based deep learning algorithms have been developed to address various computer vision issues, including object categorization, picture super-resolution, etc. De et al. [26] developed two wavelet-based edge feature enhancements in the input layer to enhance the classification performance of CNN. Liu et al. [27] adapted a multi-level wavelet CNN algorithm and applied it to image restoration, including single-image super-resolution, image denoising, and object classification. Sahito et al. [28] applied scale-invariant deep learning based on stationary wavelet transform to single-image super-resolution and captured more informa-

tion on the images. Therefore, using wavelet transform domain-based CNN and CV-CNN for PolSAR image classification is probably an effective way to preserve PolSAR image edges and sharpness while suppressing speckle noise.

Motivated by the discussions above, in this paper, we investigate the application of an active complex-valued convolutional-wavelet neural network and Markov random fields (ACV-CWNN-MRFs) in PolSAR image classification. Firstly, we incorporate dual-tree complex wavelet transform into CV-CNN (CV-CWNN) for PolSAR image classification to decrease the level of speckle noise. The CV-CWNN can preserve texture structures details and effectively enhance the classification performance of PolSAR images. Secondly, to address the issue of limited labeled samples, we take advantage of the potential of CV-CWNN in an active learning scheme (ACV-CWNN). Finally, we refine the PolSAR classification maps of ACV-CWNN via Markov random fields (MRFs), which utilize spatial contextual information to characterize the local correlation and implement the smooth alignment of PolSAR image edges.

The primary contributions of this research to the PolSAR classification literature are threefold:

1. We apply a dual-tree complex wavelet transform-constrained pooling layer to CV-CNN classification from PolSAR images to decrease the level of speckle noise and preserve some structure features (e.g., edges, sharpness) at the pooling layer.
2. We integrate a combination of AL and an MRF model-based framework into CV-CWNN for the PolSAR image classification task, which not only obtains the most informative unlabeled training samples based on the output of CV-CWNN, but also achieves spatial local correlation information to improve the classification performance.
3. The experimental results on several PolSAR benchmark datasets demonstrate that our proposed ACV-CWNN-MRF PolSAR classification algorithm outperforms other state-of-the-art CNN-based and CV-CNN-based PolSAR algorithms with fewer labeled samples, especially with speckle noise backgrounds.

The rest of this paper is organized as follows. In Section 2, we describe in detail the development of ACV-CWNN-MRF for PolSAR classification. In Section 3, we present a six-dimensional complex matrix representation of PolSAR data. In Section 4, we report and analyze the PolSAR image classification experiments and results on three benchmark datasets. A comparison of the results for some state-of-the-art deep learning methods is discussed in Section 5. The conclusions are provided in Section 6.

## 2. Methods

In this section, we first establish the basic theory of CV-CWNN by integrating dual-tree complex wavelet transform (DT-CWT) for the PolSAR image classification task. Finally, we introduce a significant development in incorporating CV-CWNN into the active learning (AL) and Markov random field (MRF) frameworks named ACV-CWNN-MRF. The configuration of the ACV-CWNN-MRF architecture for PolSAR image classification is as depicted in Figure 1.

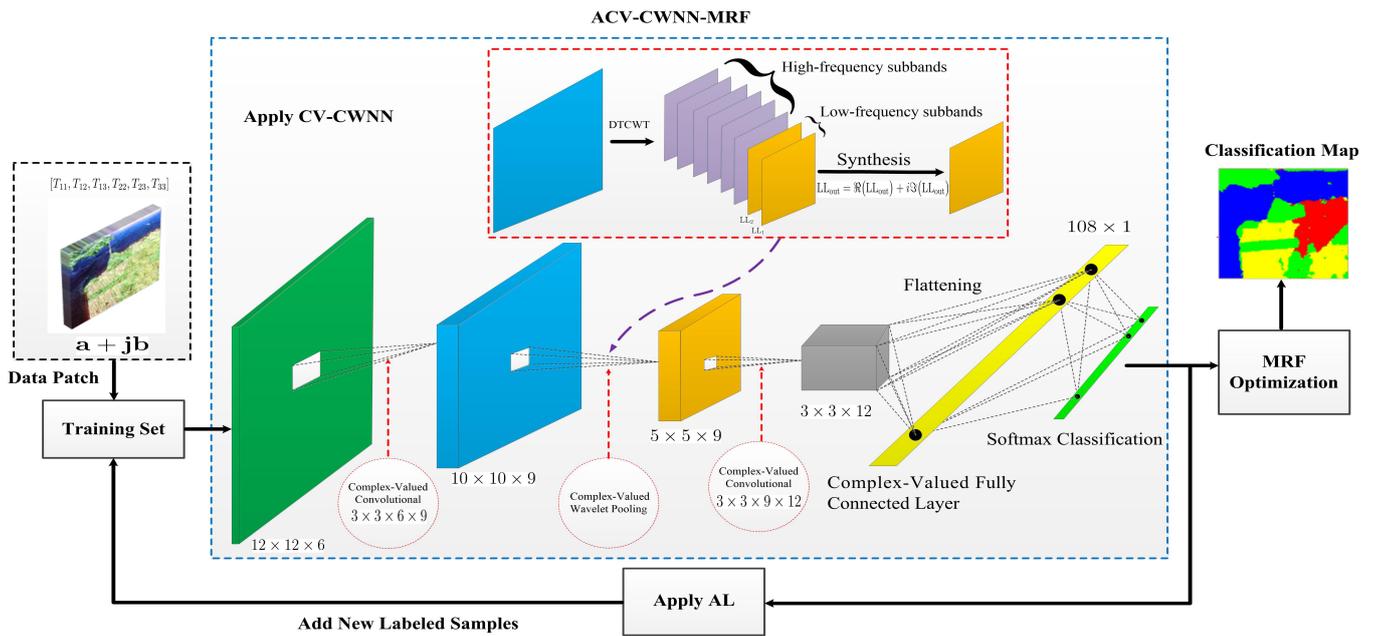


Figure 1. The configuration of ACV-CWNN-MRF architecture for PolSAR image classification.

### 2.1. The Architecture of CV-CWNN

Complex-valued convolutional-wavelet neural network (CV-CWNN) is a variant of the real-valued convolutional neural network (RV-CNN) that uses signals or image amplitude and phase information. The framework of CV-CWNN consists of five main processes: the complex-valued convolutional layer, complex-valued activation layer, complex-valued wavelet pooling layer, complex-valued fully connected layer, and complex-valued output layer. CV-CWNN is designed to deal with high-dimensional complex-valued data, and has superiority in preserving the structure of the previous layer and suppressing speckle noise at the same time over traditional CV-CNN [29,30] and RV-CNN.

Assume that  $\mathcal{X} \in \mathbb{C}^{P \times Q \times R}$  and  $\mathcal{Y} \in \mathbb{R}^{P \times Q}$  denote the PolSAR image dataset and corresponding ground-truth mask, respectively, where  $\mathbb{C}$  and  $\mathbb{R}$  are the CV and RV domains, respectively,  $P$  and  $Q$  are the range and azimuth dimensions of the PolSAR image, and  $R$  is the number of band channels. As depicted in Figure 1, the input of CV-CWNN or ACV-CWNN-MRF is the CV data patches  $\mathcal{I} = \{X[1], X[2], \dots, X[N]\}$  obtained by clipping the PolSAR dataset  $\mathcal{X}$ , where  $X[n] \in \mathbb{C}^{H \times W \times R}$  the  $n$ th input CV patch and  $N = PQ$  is the number of CV patches. The patch size, which affects the risk of overfitting and computational efficiency, plays an important role. We select the input patch dataset with a size of  $12 \times 12$ . The label corresponding to the  $n$ th input CV patch  $X[n]$  is  $Z[n] \in \mathbb{C}^{C \times N}$ , where  $C$  is the number of PolSAR target classes.

(1) *Complex-Valued Convolutional Layer*: In the CV convolutional layer, the  $l$ th ( $2 \leq l \leq L$ ) CV-CWNN convolutional layer's output feature maps  $V_i^{(l)} \in \mathbb{C}^{H_l \times W_l \times I}$  ( $1 \leq i \leq I$ ) calculates by the CV convolutional operation on the  $(l - 1)$ th layer's output feature maps  $O_j^{(l-1)} \in \mathbb{C}^{H_{l-1} \times W_{l-1} \times J}$  ( $1 \leq j \leq J$ ), where  $i$  and  $j$  are the number of feature maps in  $l$ th and  $(l - 1)$ th layers, respectively. The CV convolution with a bank of CV filters  $w_{ij}^{(l)} \in \mathbb{C}^{K_1 \times K_2 \times J \times I}$  and a CV bias  $b_i^{(l)} \in \mathbb{C}^J$  is expressed as

$$\begin{aligned}
& V_i^{(l)} \\
&= \sum_{j=1}^J w_{ij}^{(l)} * O_j^{(l-1)} + b_i^{(l)} \\
&= \sum_{j=1}^J \left( \Re(w_{ij}^{(l)}) \cdot \Re(O_j^{(l-1)}) - \Im(w_{ij}^{(l)}) \cdot \Im(O_j^{(l-1)}) \right) \\
&+ j \sum_{j=1}^J \left( \Re(w_{ij}^{(l)}) \cdot \Im(O_j^{(l-1)}) + \Im(w_{ij}^{(l)}) \cdot \Re(O_j^{(l-1)}) \right) + b_i^{(l)},
\end{aligned}$$

where  $\Re(\cdot)$  and  $\Im(\cdot)$  are, respectively, the real and imaginary parts of a CV number  $\cdot$  and  $*$  denotes the CV convolution operation. The parameters  $w_{ij}^{(l)}$  and  $b_i^{(l)}$  of CV-CWNN in the CV convolutional layer will be trained. For a CV convolutional operation with a stride size of  $S$  and a zero-padding size of  $P$ , the sizes of the CV output feature maps of  $l$ th CV-CWNN convolutional layer are represented as  $H_l = (H_{l-1} - K_1 + 2P)/S + 1$  and  $W_l = (W_{l-1} - K_2 + 2P)/S + 1$ .

(2) *Complex-Valued Activation Layer*: The linear CV features can be extracted by each CV convolutional layer. To achieve improved performance and robustness of CV-CWNN or ACV-CWNN-MRF, we must apply a nonlinear transformation to these linear CV features. In CV deep learning, the most popular CV activation functions are the CV sigmoid function (CSig) and the CV rectified linear unit (CRLU) [31,32]. The output  $O_i^{(l)} \in \mathbb{C}^{H_l \times W_l \times I}$  ( $1 \leq i \leq I$ ) of the CV activation layer can be calculated as

$$\begin{aligned}
O_i^{(l)} &= \text{CSig}(V_i^{(l)}) = \text{Sig}(\Re(V_i^{(l)})) + j\text{Sig}(\Im(V_i^{(l)})) \\
&= \frac{1}{1 + \exp(-\Re(V_i^{(l)}))} + j \frac{1}{1 + \exp(-\Im(V_i^{(l)}))},
\end{aligned}$$

or

$$\begin{aligned}
O_i^{(l)} &= \text{CRLU}(V_i^{(l)}) = \text{RLU}(\Re(V_i^{(l)})) + j\text{RLU}(\Im(V_i^{(l)})) \\
&= \begin{cases} \Re(V_i^{(l)}) + j\Im(V_i^{(l)}), & \Re(V_i^{(l)}) > 0, \Im(V_i^{(l)}) > 0 \\ \Re(V_i^{(l)}), & \Re(V_i^{(l)}) > 0, \Im(V_i^{(l)}) \leq 0 \\ j\Im(V_i^{(l)}), & \Re(V_i^{(l)}) \leq 0, \Im(V_i^{(l)}) > 0 \\ 0, & \text{otherwise.} \end{cases}
\end{aligned}$$

Note that the CV activation functions (e.g., CRLU and CSig) are, respectively, applied to the real and imaginary parts of the input layer, which is critical to greatly simplifying the complex-valued back-propagation of CV-CWNN, defined as follows.

(3) *Complex-Valued Wavelet Pooling Layer*: A CV pooling operator in CV-CNN is applied to reduce the width and height dimensions of the input features, which does not change the number of band channels. Nowadays, determining the CV average and CV maximum of a rectangular CV neighborhood by down-sampling the real and imaginary parts have become two recommended pooling functions [33]. In the CV convolutional and activation layers, we can obtain some pivotal structures (e.g., edges, sharpness). However, the conventional CV average and CV maximum pooling in CV-CNN do not take the pivotal structures of the previous layer into consideration. Additionally, the intrinsic speckle noise of the PolSAR images will have an effect on the CV-CNN's classification performance. Thus, it becomes completely necessary to preserve more pivotal structures at the CV pooling layer and suppress the speckle noise at the same time.

To address the above-mentioned issue, we utilize a dual-tree complex wavelet transform (DTCWT) [34,35]-constrained pooling operator to replace the CV average and CV maximum pooling operators. By means of the CV wavelet pooling operator based on

DTCWT, the pivotal structures can be preserved and the speckle noise can be suppressed at the same time. The DTCWT-constrained pooling operator in CV-CWNN is performed with the real part of the output  $O_i^{(l)}$  of the CV activation layers. From a multi-resolution analysis perspective,  $\Re(O_i^{(l)})$  can be perfectly reconstructed from a linear combination of a CV scaling function  $\phi_{j_0,k}$  with scale  $j_0$  and six CV wavelet functions  $\psi_{j,l}^\theta$  oriented at angle  $\theta \in \Theta = \{\pm 15^\circ, \pm 45^\circ, \pm 75^\circ\}$ , which is expressed as

$$\Re(O_i^{(l)})(x, y) = \sum_{k \in \mathbb{Z}^2} s_{j_0,k} \phi_{j_0,k}(x, y) + \sum_{\theta \in \Theta} \sum_{j \leq j_0} \sum_{k \in \mathbb{Z}^2} c_{j,k}^\theta \psi_{j,k}^\theta(x, y),$$

where  $(x, y)$  denotes the pixel coordinate in the real part of feature map  $O_i^{(l)}$ , and  $s_{j_0,k}$  and  $c_{j,k}^\theta$  denote the scaling coefficients and wavelet coefficients, respectively. Moreover, DTCWT is designed to have five desired properties in perfect reconstruction, approximate shift invariance, good directional selectivity, efficient order-N computation, and limited redundancy [36]. DTCWT with tree A (odd-length filters) and tree B (even-length filters) for pooling layers in CV-CWNN is illustrated in Figure 2. The level 1 filter pairs,  $h_0(n)$  and  $h_1(n)$ ,  $g_0(n)$ , and  $g_1(n)$ , in Figure 2 are low-pass/high-pass and conjugate orthogonality filter pairs. Thus, for the real part of each input feature map  $O_i^{(l)}$ , DTCWT can produce eight complex bandpass subimages:

$$\text{DTCWT}(\Re(O_i^{(l)})(x, y)) = \{LL_1, LH_1, HL_1, HH_1, LL_2, LH_2, HL_2, HH_2\},$$

where  $LL_1$  and  $LL_2$  are two complex low-frequency bandpasses subimages, and  $LH_1, HL_1, HH_1, LH_2, HL_2,$  and  $HH_2$  are six complex high-frequency bandpass subimages. Then, the average of the real and imaginary parts of the low-frequency bandpass subimages  $LL_1$  and  $LL_2$  is calculated as the real part of the output feature of the CV wavelet pooling layer in CV-CWNN, which is expressed as

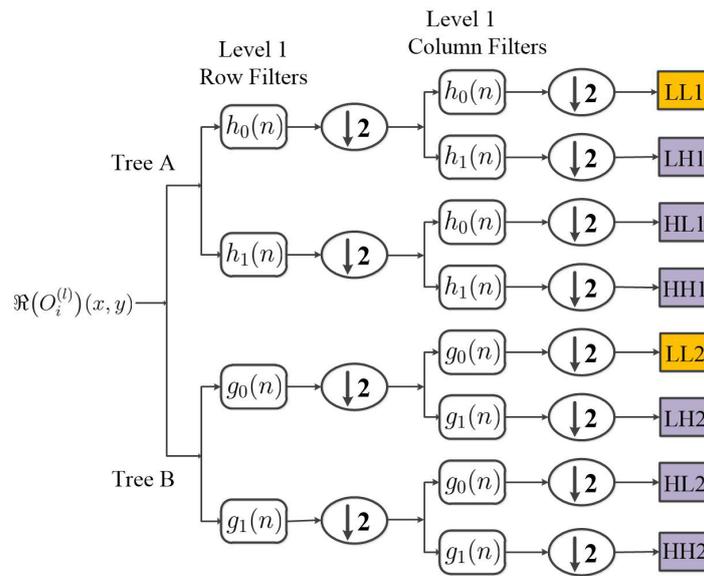
$$\Re(LL_{\text{out}}) = \frac{1}{4}(\Re(LL_1) + \Im(LL_1) + \Re(LL_2) + \Im(LL_2)),$$

where  $\Re(LL_{\text{out}})$  is the real part of the output feature of the CV wavelet pooling layer. For the same operation of  $\Im(O_i^{(l)})$ , we can obtain  $\Im(LL_{\text{out}})$ , which is the imaginary part of the output feature of the CV wavelet pooling layer. In fact, the low-frequency bandpass subimages in CV-CWNN are retained in order to preserve some pivotal structures of the input, and the high-frequency bandpass subimages are discarded in order to suppress the speckle noise.

(4) *Complex-Valued Fully Connected Layer*: The CV fully connected layer converts the CV data stream into 1-dimensional data to perform category determination. On top of several combinations of CV convolutional layers, CV activation layers, and CV wavelet pooling layers, one or two CV fully connected layers are employed in ACV-CWNN, which is considered as a special case of a CV convolutional layer. Each output neuron in the CV fully connected layer is in connection with the input neurons in the previous layer. Moreover, the number of output neurons is not fixed. The output CV feature vector  $O_i^{(L)}$  of the CV fully connected layer is calculated with the output feature map  $O_i^{(L)}$  of the previous layer, which is expressed as

$$O_i^{(L)} = f(V_i^{(L)}), \quad V_i^{(L)} = \sum_{j=1}^C w_{ij}^{(L)} \cdot O_j^{(L-1)} + b_i^{(L)}, \quad (1)$$

where  $f$  is the CSig or CReLU function, and  $K$  is the number of input neurons in the  $L$ th CV fully connected layer.



**Figure 2.** Flowchart of 2D DTCWT for level  $j_0 = 1$ .

(5) *Complex-Valued Output Layer:* A CV softmax operation is applied to the CV output layer of CV-CWNN for PolSAR images classification. The output of the CV softmax operation is a  $C$ -dimensional CV one-hot vector, while  $C$  is the number of PolSAR target classes. Each element of the CV one-hot vector varies in the range  $[0, 1] + j \cdot [0, 1]$ , and all the elements sum to  $1 + 1 \cdot j$ , which satisfies the probability-like property. The input of the CV softmax operation is the  $(L)$ th CV fully connected layer's output feature map  $O_i^{(L)}$ . Then, the CV softmax operation for the  $i$ th element of the CV one-hot vector can be represented as

$$\begin{aligned}
 O_i^{\text{out}} &= \text{softmax}\left(O_i^{(L)}\right) \\
 &= \frac{\exp\left(\Re\left(O_i^{(L)}\right)\right)}{\sum_{i=1}^C \exp\left(\Re\left(O_i^{(L)}\right)\right)} + j \frac{\exp\left(\Im\left(O_i^{(L)}\right)\right)}{\sum_{i=1}^C \exp\left(\Im\left(O_i^{(L)}\right)\right)}, \tag{2}
 \end{aligned}$$

where  $i \in \{1, 2, \dots, C\}$  and the output  $O_i^{\text{out}}$  denotes the CV probability belonging to class  $i$  for one CV training sample.

### 2.2. The Complex-Valued Back-Propagation of CV-CWNN

In this paper, we utilize the  $\mathbb{C}$ Sig function in the CV activation layer and CV fully connected layer. The error  $E_n^{(L)}$  between the  $L$ th CV output layer's output  $O^{(L)}[n]$  and the label  $Y[n]$  of the  $n$ th single CV patch can be directly measured as

$$E_n^{(L)} = Y[n] - O^{(L)}[n].$$

The MSE loss function is widely used in RV-CNN and CV-CNN algorithms. However, CV data patches may be contaminated by non-Gaussian noise or the outliers. For PolSAR classification with non-Gaussian noise and outliers, the cross-entropy loss function might perform better than the MSE loss function since the MSE loss function is just optimal for samples with Gaussian noise and introduces the problem of the gradient disappearing.

Then, the CV cross-entropy loss function is chosen to train the CV-CWNN network, which is given by

$$\text{Loss} = -\frac{1}{N} \sum_{n=1}^N \sum_{i=1}^C \left\{ [(1 - \Re(Z_i[n])) \ln(1 - \Re(O_i[n])) + \Re(Z_i[n]) \ln(\Re(O_i[n]))] \right. \\ \left. + [(1 - \Im(Z_i[n])) \ln(1 - \Im(O_i[n])) + \Im(Z_i[n]) \ln(\Im(O_i[n]))] \right\}. \quad (3)$$

In the process of the complex-valued back-propagation [37,38] of CV-CWNN, the CV intermediate quantity  $\delta^{(l)}$  with the output  $V^{(l)}$  of the previous layer, also called the CV backward error term, is defined as follows

$$\delta^{(l)} = \frac{\partial \text{Loss}}{\partial \Re(V^{(l)})} + j \frac{\partial \text{Loss}}{\partial \Im(V^{(l)})}, \quad \delta^{(l)} \in \mathbb{C}^I$$

Thus, the parameters  $w_{ij}^{(l)}$  and  $b_i^{(l)}$  in CV-CWNN can be adjusted iteratively by  $J$  and the learning rate  $\eta$  in terms of

$$w_{ij}^{(l)}(t+1) = \begin{cases} w_{ij}^{(l)}(t) - \eta \delta_i^{(l)} \cdot \overline{O_i^{(l-1)}}, & l = L \\ w_{ij}^{(l)}(t) - \eta \delta_i^{(l)} * \overline{O_i^{(l-1)}}, & l < L \end{cases}, \quad (4)$$

and

$$b_{ij}^{(l)}(t+1) = b_{ij}^{(l)}(t) - \eta \delta_i^{(l)}, \quad (5)$$

where  $t$  is the training time, and  $\overline{(\cdot)}$  is the complex conjugate. The parameters are adjusted until the value of loss function no longer decreases. Mathematically,  $\delta^{(l)}$  can be calculated for each layer in the following subsection.

(1) *CV Output Layer:* There are no parameters to be learned in the CV output layer. Combining (2) and (3), the CV backward error term  $\delta_i^{(\text{out})}$  of the CV output layer can be calculated as

$$\delta_i^{(\text{out})} = \frac{\partial \text{Loss}}{\partial \Re(V_i^{(L)})} + j \frac{\partial \text{Loss}}{\partial \Im(V_i^{(L)})} \\ = \left( \frac{\exp(\Re(O_i^{(L)}))}{\sum_{i=1}^C \exp(\Re(O_i^{(L)}))} - \Re(Z_i) \right) + j \left( \frac{\exp(\Im(O_i^{(L)}))}{\sum_{i=1}^C \exp(\Im(O_i^{(L)}))} - \Im(Z_i) \right) \\ = \left( \Re(O^{\text{out}}) - \Re(Z_i) \right) + j \left( \Im(O^{\text{out}}) - \Im(Z_i) \right).$$

(2) *CV Fully Connected Layer:* Combining (1) and (3), the CV backward error term  $\delta_i^{(l+1)}$  of the  $(l+1)$ th CV fully connected layer can be expressed as

$$\delta_i^{(l+1)} \\ = \left( \frac{\partial \text{Loss}}{\partial \Re(O_i^{(l+1)})} \frac{\partial \Re(O_i^{(l+1)})}{\partial \Re(V_i^{(l+1)})} + \frac{\partial \text{Loss}}{\partial \Im(O_i^{(l+1)})} \frac{\partial \Im(O_i^{(l+1)})}{\partial \Re(V_i^{(l+1)})} \right) \\ + j \left( \frac{\partial \text{Loss}}{\partial \Re(O_i^{(l+1)})} \frac{\partial \Re(O_i^{(l+1)})}{\partial \Im(V_i^{(l+1)})} + \frac{\partial \text{Loss}}{\partial \Im(O_i^{(l+1)})} \frac{\partial \Im(O_i^{(l+1)})}{\partial \Im(V_i^{(l+1)})} \right) \\ = \left( \Re(O_i^{(l+1)}) - \Re(Z_i) \right) + j \left( \Im(O_i^{(l+1)}) - \Im(Z_i) \right). \quad (6)$$

In the previous layer,  $O_i^{(l)}$  is affected by the backward error term  $\delta_i^{(l+1)}$  through all the units in  $O_i^{(l+1)}$ . Considering (6), the CV recurrence relation from  $(l+1)$ th layer to  $l$ th layers can be formulated as

$$\begin{aligned} \delta_i^{(l)} &= \Re(O_i^{(l)}) \left(1 - \Re(O_i^{(l)})\right) \times \left( \frac{\partial \text{Loss}}{\partial \Re(V_i^{(l+1)})} \frac{\partial \Re(V_i^{(l+1)})}{\partial \Re(O_i^{(l)})} + \frac{\partial \text{Loss}}{\partial \Im(V_i^{(l+1)})} \frac{\partial \Im(V_i^{(l+1)})}{\partial \Re(O_i^{(l)})} \right) \\ &+ j \Im(O_i^{(l)}) \left(1 - \Im(O_i^{(l)})\right) \times \left( \frac{\partial \text{Loss}}{\partial \Re(V_i^{(l+1)})} \frac{\partial \Re(V_i^{(l+1)})}{\partial \Im(O_i^{(l)})} + \frac{\partial \text{Loss}}{\partial \Im(V_i^{(l+1)})} \frac{\partial \Im(V_i^{(l+1)})}{\partial \Im(O_i^{(l)})} \right) \\ &= \Re(O_i^{(l)}) \left(1 - \Re(O_i^{(l)})\right) \sum_{j=1}^J \left( \Re(w_{ij}^{(l+1)}) \Re(\delta_j^{(l+1)}) + \Im(w_{ij}^{(l+1)}) \Im(\delta_j^{(l+1)}) \right) \\ &+ j \Im(O_i^{(l)}) \left(1 - \Im(O_i^{(l)})\right) \times \sum_{j=1}^J \left( \Re(w_{ij}^{(l+1)}) \Im(\delta_j^{(l+1)}) - \Im(w_{ij}^{(l+1)}) \Re(\delta_j^{(l+1)}) \right) \end{aligned} \quad (7)$$

(3) *CV Convolutional Layer and Activation Layer*: The CV backward error term  $\delta_i^{(l)}$  of  $l$ th CV convolutional layer is related to the CV backward error term  $\delta_i^{(l+1)}$  of the CV wavelet pooling layer and the CV wavelet pooling weight  $\alpha_i^{(l+1)}$ . The weight  $\alpha_i^{(l+1)}$  in a pooling layer map is calculated as

$$\alpha_i^{(l+1)} = \frac{1}{\phi^h(x, y)} + j \frac{1}{\phi^s(x, y)} = \frac{1}{\phi^h(x)} \frac{1}{\phi^h(y)} + j \frac{1}{\phi^s(x)} \frac{1}{\phi^s(y)},$$

where  $\phi^h(x, y)$  and  $\phi^s(x, y)$  are the CV scaling functions of tree A and tree B respectively.

To keep the same size as the CV convolutional layer's map, we upsample  $\delta_j^{(l+1)}$  by repeating each pixel 2 times in horizontal and vertical directions, denoted by  $\text{up}(\delta_j^{(l+1)})$ . Similar to the derivation (7) of the hidden layer in the CV fully connected layer, the CV backward error term  $\delta_i^{(l)}$  of the CV convolutional layer and activation layer can be expressed as

$$\begin{aligned} \delta_i^{(l)} &= \Re(O_i^{(l)}) \left(1 - \Re(O_i^{(l)})\right) \sum_{j=1}^J \Re(\alpha_i^{(l+1)}) \Re(\text{up}(\delta_j^{(l+1)})) \\ &+ j \Im(O_i^{(l)}) \left(1 - \Im(O_i^{(l)})\right) \sum_{j=1}^J \Im(\alpha_i^{(l+1)}) \Im(\text{up}(\delta_j^{(l+1)})). \end{aligned} \quad (8)$$

(4) *CV Wavelet Pooling Layer*: There are no parameters to be learned in the CV wavelet pooling layer. According to (7), the CV backward error term  $\delta_i^{(l)}$  of the CV wavelet pooling layer can be calculated as

$$\begin{aligned} \delta_i^{(l)} &= \sum_{j=1}^J \left( \Re(w_{ij}^{(l+1)}) \Re(\delta_j^{(l+1)}) + \Im(w_{ij}^{(l+1)}) \Im(\delta_j^{(l+1)}) \right) \\ &+ j \sum_{j=1}^J \left( \Re(w_{ij}^{(l+1)}) \Im(\delta_j^{(l+1)}) - \Im(w_{ij}^{(l+1)}) \Re(\delta_j^{(l+1)}) \right). \end{aligned} \quad (9)$$

After calculating the CV backward error term of each layer, the weight and bias values are updated iteratively using Equations (4) and (5) until the precision between the network's output and the target satisfies the requirement.

### 2.3. AL and MRF-Based CV-CWNN PolSAR Classification

In this subsection, we briefly introduce the core theory of AL and MRF for PolSAR classification using CV-CWNN.

(1) *BvSB-Based AL*: The basic idea of AL is enlarging the training dataset iteratively by selecting the most informative unlabeled samples. Many effective AL criteria are proposed to measure the informativeness of candidate sets to query for annotations, such as random selection (RS), mutual information (MI)-based criteria [39], the breaking ties (BT) algorithm [40], the entropy measure (EP) [41] and the best-versus-second-best (BvSB) measure [42]. The EP and BvSB measures are implemented by the probability estimates of candidate pixels. Due to our designed CV-CWNN model, which can supply exact pixel-wise class probabilities and perform the multiclass PolSAR classification task, it is beneficial to adopt the BvSB measure as the AL strategy in our CV-CWNN (i.e., ACV-CWNN). Specifically, the BvSB measure can be defined as

$$X_n^{\text{BvSB}} = \arg \min_{X[n], n \in \mathcal{U}} \left[ \left( \Re(O^{\text{out}}(y_{\text{Best}}|X[n])) + \Im(O^{\text{out}}(y_{\text{Best}}|X[n])) \right) - \left( \Re(O^{\text{out}}(y_{\text{Second-Best}}|X[n])) + \Im(O^{\text{out}}(y_{\text{Second-Best}}|X[n])) \right) \right], \quad (10)$$

where  $y_{\text{Best}}, y_{\text{Second-Best}} \in \{1, 2, \dots, C\}$ ,  $p(y_{\text{Best}}|X[n])$  and  $p(y_{\text{Second-Best}}|X[n])$  denote the best and second best class membership probability outputs from the CV-CWNN of the CV data patch  $X[n]$ , respectively, and  $\mathcal{U}$  is the candidate pool.

This demonstrates that adopting the BvSB-Based AL strategy in CV-CWNN is very essential since the PolSAR classification task lacks enough unlabeled training samples. Additionally, it should be emphasized that (10) assumes that only one sample is annotated in each iteration. However, the reality is that more than one candidate is annotated.

(2) *MRF-Based Optimization*: In order to improve the classification performance of ACV-CWNN, we utilize MRF [12,43] to model the spatial correlation of PolSAR neighboring pixels. The MAP-MRF framework makes the interpixel class dependence assumption that spatially neighboring pixels are likely to belong to the same class. According to the Hammersley–Clifford theorem, the final MRF classification model can thus be formulated as

$$\mathbf{y} = \arg \max_{\mathbf{y} \in \mathcal{K}^N} \left\{ \sum_{n=1}^N \log \left( \Re(O^{\text{out}}(y_i|X[n])) + \Im(O^{\text{out}}(y_i|X[n])) \right) + \beta \sum_{n=1}^N \sum_{m \in \mathcal{C}_n} \delta(y_n, y_m) \right\}, \quad (11)$$

where  $\mathcal{K} = \{1, 2, \dots, C\}$  is a label set for  $C$  classes,  $\mathbf{y} = \text{reshape}(\mathcal{Y}, N, 1)$  is a class label set for CV data patches  $\mathcal{I}$ ,  $\delta(\cdot, \cdot)$  is the Kronecker delta function (i.e.,  $\delta(y_n, y_m) = 1$  if  $y_n = y_m$ , and  $\delta(y_n, y_m) = 0$  otherwise) to count the support from the neighboring pixels if the current pixel  $X[n]$  is labeled as  $y_n$ ,  $\mathcal{C}_n$  is the neighbor of pixel  $X[n]$ , and  $\beta$  ( $\beta > 0$ ) is a constant that controls the importance of the neighboring pixels.

Solving the MRF model (11) is a combinatorial optimization problem which is NP-hard. Recently, some approximating algorithms, such as graph cut [44], message passing [45], and loopy belief propagation [46], have been proposed to achieve the optimal solution. In this paper, we adopt loopy belief propagation since it generally converges fast within less than ten iterations.

### 3. PolSAR Data Processing

For PolSAR images, the complex scattering matrix  $\mathcal{S}$  is usually used to capture the scattering characteristics of a single target through HH, HV, VH, and VV polarization channels, which can be expressed as

$$\mathcal{S} = \begin{bmatrix} \mathcal{S}_{\text{HH}} & \mathcal{S}_{\text{HV}} \\ \mathcal{S}_{\text{VH}} & \mathcal{S}_{\text{VV}} \end{bmatrix},$$

where “H” and “V” denote the orthogonal horizontal/vertical polarization directions, respectively,  $\mathcal{S}_{\text{HV}}$  is the complex scattering coefficient of horizontal transmitting and vertical receiving polarization, and the other coefficients are similarly defined. In the case of recip-

rocal backscattering,  $\mathcal{S}_{HV} = \mathcal{S}_{VH}$ , and  $\mathcal{S}$  based on the Pauli basis [47] can be represented by a complex scattering vector  $k_{3P}$ ,

$$k_{3P} = \frac{1}{\sqrt{2}}[\mathcal{S}_{HH} + \mathcal{S}_{VV}, \mathcal{S}_{HH} - \mathcal{S}_{VV}, 2\mathcal{S}_{HV}]^T,$$

where subscript  $T$  denotes the matrix transpose.

Thus, the coherency matrix  $\mathbf{T}$  of PolSAR data can be obtained as follows:

$$\mathbf{T} = k_{3P}k_{3P}^H = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{12}^* & T_{22} & T_{23} \\ T_{13}^* & T_{23}^* & T_{33} \end{bmatrix},$$

where subscript  $H$  and  $*$  denote the conjugate transpose and conjugate operation, respectively. Therefore, the upper triangular elements ( $T_{11}, T_{12}, T_{13}, T_{22}, T_{23}, T_{33}$ ) of the coherency matrix  $\mathbf{T}$  are used as the input dataset of ACV-CWNN-MRF, and  $R = 6$ .

After the preprocessing of the input datasets, such as expansion operation, data augmentation (DA), and batch normalization (BN), we can obtain a higher classification accuracy for PolSAR images.

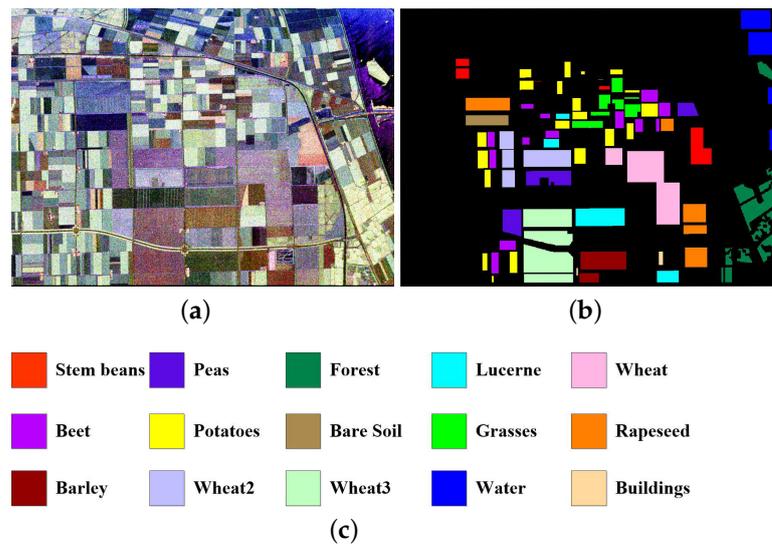
#### 4. Results

In order to verify the effectiveness of the proposed ACV-CWNN-MRF approach, a series of experiments on three real PolSAR datasets are analyzed and compared with some state-of-the-art deep learning-based methods in the following experiments. The overall accuracy (OA), average accuracy (AA), Kappa coefficient ( $\kappa$ ), and the confusion matrix are used to evaluate the performance on the proposed ACV-CWNN-MRF method. Details about these three benchmark PolSAR datasets are described as follows. We randomly select some samples as the training dataset and the remaining samples as the testing dataset.

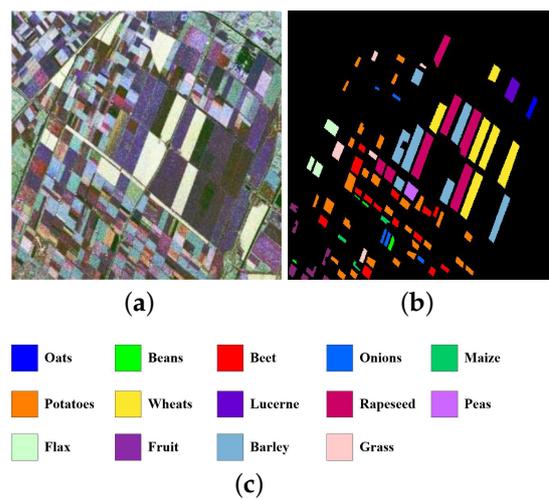
(1) *Flevoland Dataset*: This dataset was acquired by the NASA/JPL AIRSAR platform over an agriculture area of the Netherlands on August 1989, and is a subset of an L-band, full PolSAR image with a range resolution of 6.6 m and an azimuth resolution of 12.1 m. There are, in total, 15 identified categories, including stem beans, beet, barley, peas, potatoes, three types of wheat, forest, bare soil, lucerne, grasses, water, rapeseed, and buildings. The PauliRGB image of the Flevoland dataset is shown in Figure 3a, whose size is  $750 \times 1024$  pixels. A ground-truth map of the 15 categories and a corresponding color legend of the ground-truth map are shown in Figure 3b,c, respectively.

(2) *Benchmark Dataset*: A Benchmark PolSAR dataset of another L-band full PolSAR image was acquired over the Flevoland area in 1991, and is widely used for PolSAR image classification research. The size of the PolSAR image is  $1020 \times 1024$  pixels. There are, in total, 14 identified categories, including oats, potatoes, flax, beans, wheats, fruit, beet, lucerne, barley, onions, rapeseed, grass, maize, and peas. The PauliRGB image of the Flevoland dataset, a ground-truth map of the 14 categories, and a corresponding color legend of the ground-truth map are shown in Figure 4a–c, respectively.

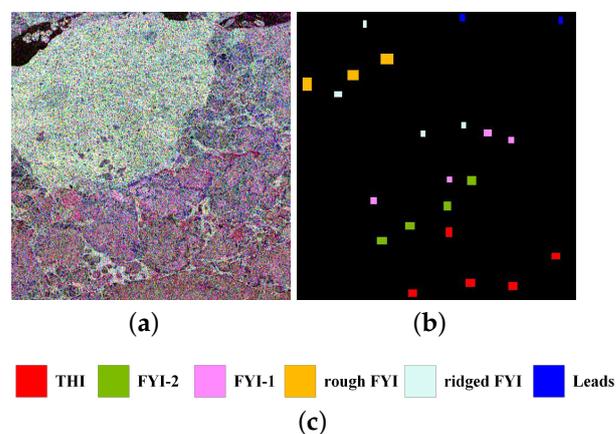
(3) *CONVAIR Ice Dataset*: The third dataset comprises multilook CONVAIR C-band full PolSAR data obtained over an ice area in Canada in 2001. The original PolSAR image has dimensions of  $544 \times 5238$  pixels with a spatial resolution of  $4 \text{ m} \times 0.43 \text{ m}$ . This Pauli RGB image with  $544 \times 523$  pixels in Figure 5a was produced by using 10x Azimuth MultiLooking during the CONVAIR data extraction and conversion process. There are, in total, five identified categories, including leads, ThI (thin ice, new forming ice), rough FYI (rough First-Year Ice), ridged FYI (ridged First-Year Ice), FYI-1 (floes, far range), and FYI-2 (floes, near range). A ground-truth map of the five categories and a corresponding color legend of the ground-truth map are shown in Figure 5b,c, respectively.



**Figure 3.** Flevoland dataset. (a) Pauli RGB image of the target scene. (b) Ground-truth map of the 15 categories. (c) Color legend of the ground-truth map.



**Figure 4.** Benchmark dataset. (a) Pauli RGB image of the target scene. (b) Ground-truth map of the 14 categories. (c) Color legend of the ground-truth map.



**Figure 5.** CONVAIR Ice dataset. (a) Pauli RGB image of the target scene. (b) Ground-truth map of the 5 categories. (c) Color legend of the ground-truth map.

#### 4.1. Parameter Setting

In this subsection, we conduct an experiment to evaluate the effectiveness of our proposed ACV-CWNN-MRF method for PolSAR image classification incorporating four key strategies: (1) batch normalization (BN); (2) data augmentation (DA); (3) active learning (AL); (4) MRF optimization. In this experiment, the baseline method is CV-CWNN. To discuss the effectiveness of these strategies, we establish four comparison methods which gradually add each of the other techniques.

- (1) *CV-CWNN*: The method that is introduced in Section 2.1.
- (2) *CV-CWNN + BN (CV-CWNN-B)*: A method that combines the CV-CWNN method and BN strategy.
- (3) *CV-CWNN + BN + DA (CV-CWNN-BD)*: A method that incorporates the CV-CWNN-B method with the DA strategy.
- (4) *CV-CWNN + BN + DA + AL (ACV-CWNN-BD)*: A method that integrates the AL strategy with the CV-CWNN-BD method.
- (5) *CV-CWNN + BN + DA + AL +MRF (ACV-CWNN-MRF)*: A method that integrates the MRF optimization strategy with the ACV-CWNN-BD method.

We take the Flevoland dataset to construct the experiment, and the relevant experimental parameters for the CV-CWNN-based methods are set as follows. The learning rate  $\eta$  is set at 0.6. The batchsize is empirically set as 128. For all the methods, we conduct two rounds for training in the experiments and randomly select different sample rate of available labeled samples for training. For non-AL based methods, including CV-CWNN, CV-CWNN-B, and CV-CWNN-BD, each round contains 40 training epochs until the training model converges. For AL based methods, including ACV-CWNN-BD and ACV-CWNN-MRF, the first round contains 15 training epochs, while the second round contains 20 training epochs. Specifically, for the first round, we randomly choose 500 labeled samples as the training dataset and the remaining samples as the testing dataset. For the second round, we randomly add 50 samples into the training set for the non-AL based methods and actively select 50 samples to enlarge the training set for the AL based methods. We compute The average OA and the standard derivation of CV-CWNN based methods by averaging 5 independent runs. The experimental results are recorded in Table 1.

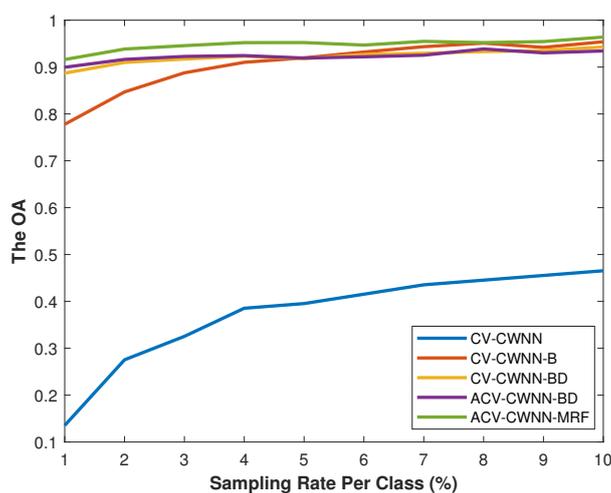
**Table 1.** The average OA and the standard derivation (%) of CV-CWNN based methods with different strategies for PolSAR classification.

Method	CV-CWNN	CV-CWNN-B	CV-CWNN-BD	ACV-CWNN-BD	ACV-CWNN-MRF
1st round, 500 samples	11.52 (2.86)	17.16 (1.51)	46.29 (2.12)	60.74 (2.47)	61.66 (1.51)
2st round, 550 samples	12.26 (1.80)	28.13 (2.91)	68.85 (1.92)	70.14 (2.92)	72.33 (2.14)

From Table 1, we can observe by column that each method has better performance as the round carries out, which is due to add some samples into the training dataset after the first round. From the results by row, we observe that the different strategy can improve classification and recognition performance. Specifically, CV-CWNN-B can first obtain a classification OA of up to 10% improvement compared with CV-CWNN. Second, DA strategy can achieve an increase of approximately 35% OA by comparing CV-CWNN-BD and CV-CWNN-B. Next, compared with CV-CWNN-B, ACV-CWNN-BD with AL strategy increase OA by about 8%. Finally, MRF strategy can further obtain about 1.5% improvement of classification OA. Therefore, it demonstrates that each different strategy used in the CV-CWNN based methods can improve the overall classification performance.

Second, in order to evaluate the performance of the CV-CWNN based methods and investigate an optimal sampling rate, we carry out the contrasting experiments on the overall accuracy (OA) versus different sampling rate per class. As can be seen from Figure 6,

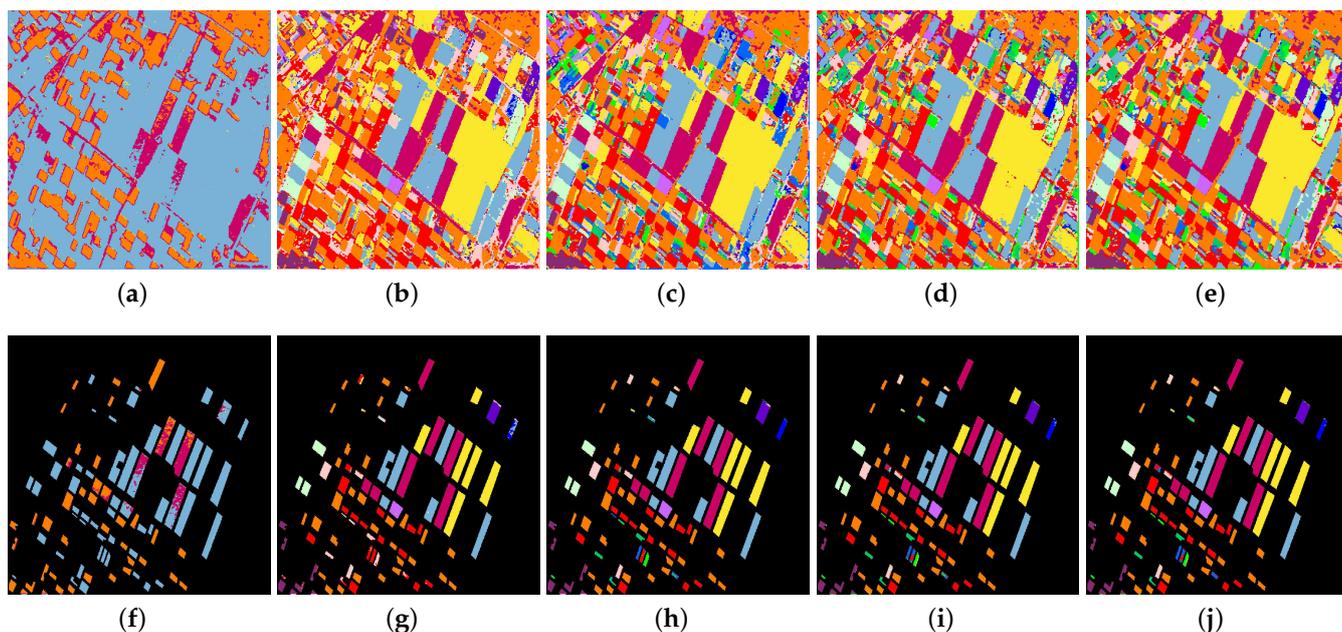
when the sampling rate is 1%, the OA of ACV-CWNN-MRF is approximately 91.61%, and as the sampling rate approaches 4%, the accuracy quickly rises to 95%. It follows that for the Flevoland data set, a 5% sample rate is sufficient for ACV-CWNN-MRF. Indeed, it is observable from Figure 6 that as the number of training samples increases, the accuracy of all CV-CWNN based classification methods improves. Additionally, it is evident that each technique is capable of enhancing the classification performance of the CV-CWNN based model to a certain extent.



**Figure 6.** The overall accuracy (OA) obtained by all CV-CWNN based methods with different numbers of training samples on the Flevoland dataset.

#### 4.2. Effectiveness Verification

We present our proposed method's performance and analysis using the Benchmark Dataset. According to the discussions aforementioned, for training, we randomly select 5% of the available labeled samples each class. For AL based methods, the first round contains 15 training epochs, while the second round contains 20 training epochs. For the second round, we randomly add 50 samples into the training set for the non-AL based methods and actively select 50 samples to enlarge the training set for the AL based methods. The size of a local patch is 12. The classification maps achieved by the CV-CWNN based methods are shown in Figure 7, and the classification accuracies are reported in Table 2. It is evident from Table 2 that, when compared to all other approaches, the proposed ACV-CWNN-MRF method performs the best in terms of AA and OA metrics. Comparing the CAs of all the CV-CWNN based methods, it can be seen that the suggested method has the best classification accuracy in the majority of the classes. Furthermore, Figure 7 shows that the proposed ACV-CWNN-MRF method can produce smoother and more accurate classification maps.



**Figure 7.** Classification maps for the Benchmark dataset with 15 training epochs the first round and 20 training epochs the second round by different methods. (a,f) CV-CWNN classified map. (b,g) CV-CWNN-B classified map. (c,h) CV-CWNN-BD classified map. (d,i) ACV-CWNN-BD classified map. (e,j) ACV-CWNN-MRF classified map.

**Table 2.** Classification accuracies (%) for the Benchmark dataset with 15 training epochs the first round and 20 training epochs the second round by different methods.

Class	CV-CWNN	CV-CWNN-B	CV-CWNN-BD	ACV-CWNN-BD	ACV-CWNN-MRF
1	0	99.78	99.54	99.63	99.80
2	0	90.83	91.27	91.54	95.39
3	0	50.34	93.74	92.88	96.67
4	0	99.09	98.58	96.94	97.70
5	99.94	99.65	99.76	99.94	99.95
6	0	0.15	42.34	46.47	57.50
7	0	99.82	99.71	99.67	99.78
8	0	0	49.71	80.90	88.60
9	0	92.73	99.07	96.87	98.73
10	0	0	94.05	95.40	98.19
11	0	96.21	95.37	95.42	97.28
12	99.39	99.84	99.63	99.78	99.70
13	0	71.49	88.03	82.24	87.32
14	0	93.69	93.33	95.97	96.40
AA	14.24	70.97	88.87	90.89	93.79
OA	38.85	94.37	97.25	97.36	98.14
$\kappa$	24.19	93.35	96.75	96.89	97.80

## 5. Discussion

In this section, we compare the ACV-CWNN-MRF method with other some state-of-the-art PolSAR classification methods on CONVAIR Ice dataset with the speckle noise. These compared methods include real-valued CNN (RV-CNN) [14], real-valued CNN-MRF (RV-CNN-MRF) [48], complex-valued CNN (CV-CNN) [17], and real-valued CNN-AL-MRF (RV-CNN-AL-MRF) [25]. The experimental settings for the CONVAIR Ice dataset are as

follows. For all methods, there are 238 training samples altogether since we choose 5% of the CONVAIR Ice dataset at random from each class to use as the training set. The learning rate  $\eta$  is set as 0.8. The batch size is empirically set at 100. For RV-CNN, RV-CNN-MRF, and CV-CNN, each round contains 50 epochs, which do not employ the AL strategy, while for RV-CNN-AL-MRF and ACV-CWNN-MRF, the first round contains 25 epochs and the second round contains 50 epochs. For the second round, we actively select 50 samples to enlarge the training set for the AL-based methods.

The speckle noise in the PolSAR image of the HH, HV, VH, and VV polarization channels is modeled as a multiplicative form, which can be expressed as  $\tilde{f} = f + \gamma \times f$ , where  $f = S_{HH}, S_{HV}, S_{VH}$ , and  $S_{VV}$ , and  $\gamma$  is white Gaussian noise. The original multiplicative noise of the PolSAR speckling model can be converted to an additive case by the logarithmic operation  $\log(\tilde{f}) = \log(f) + \log(\gamma + 1)$ . Therefore, the log-transformed speckle model of PolSAR images can also be regarded as an additive white Gaussian noise model with a mean of 0 and variance  $\sigma$  for mathematical tractability [49]. We consider the range of the level  $\gamma$  of speckle noise as  $\gamma \in [0.02, 0.18]$ , and the local patch size as  $12 \times 12$ , so as to train the compared model and the proposed model for PolSAR classification.

Finally, the overall accuracies (OAs), the average accuracies (AAs), and the Kappa coefficients ( $\kappa$ ) for all considered methods with different noise levels of speckles are reported in Figure 8. Visually, Figure 8 shows that as the level  $\gamma$  of speckle noise increases, the PolSAR image classification performance for all considered methods will decrease. Moreover, when the noise level exceeds 0.08, the classification performance drops sharply. However, it can be also easily observed that in the presence of PolSAR speckle noise, our ACV-CWNN-MRF method achieves the best performance in OA, AA, and  $\kappa$  on the CONVAIR Ice dataset compared with the other four methods. The superior classification performance of our ACV-CWNN-MRF method can be explained by exploiting the CWNN and MRF strategy simultaneously, where the complex-valued wavelet pooling layer of CWNN can not only remove some PolSAR speckle noise, but it can also maintain the structures of the feature maps in accordance with the specified rules. Additionally, classification maps of the CONVAIR Ice dataset with noise levels of  $\gamma = 0.02$  and  $\gamma = 0.16$  using different methods are presented in Figures 9 and 10, respectively. It can be visually seen from Figures 9 and 10 that our ACV-CWNN-MRF method for the despeckled PolSAR classification achieves a smoother classification map than RV-CNN, RV-CNN-MRF, RF-CNN-AL-MRF, and CV-CNN. According to the aforementioned discussions, our ACV-CWNN-MRF method can suppress the influence of speckle noise and obtain more accurate, smooth, and coherent classification performance.

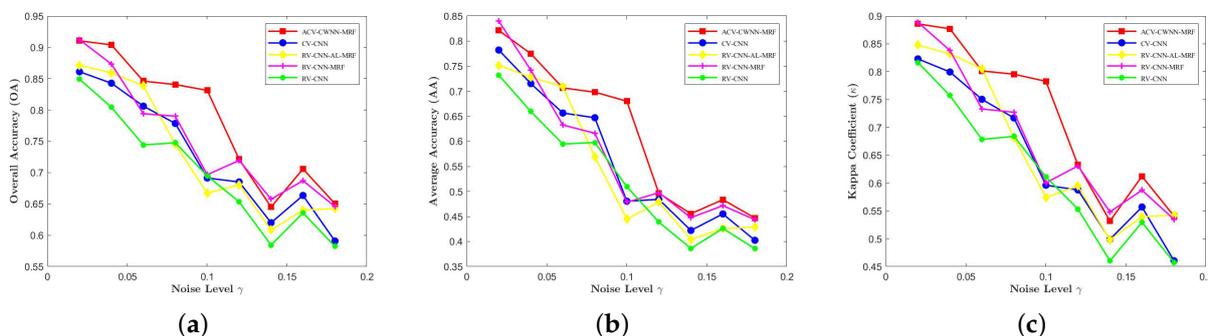
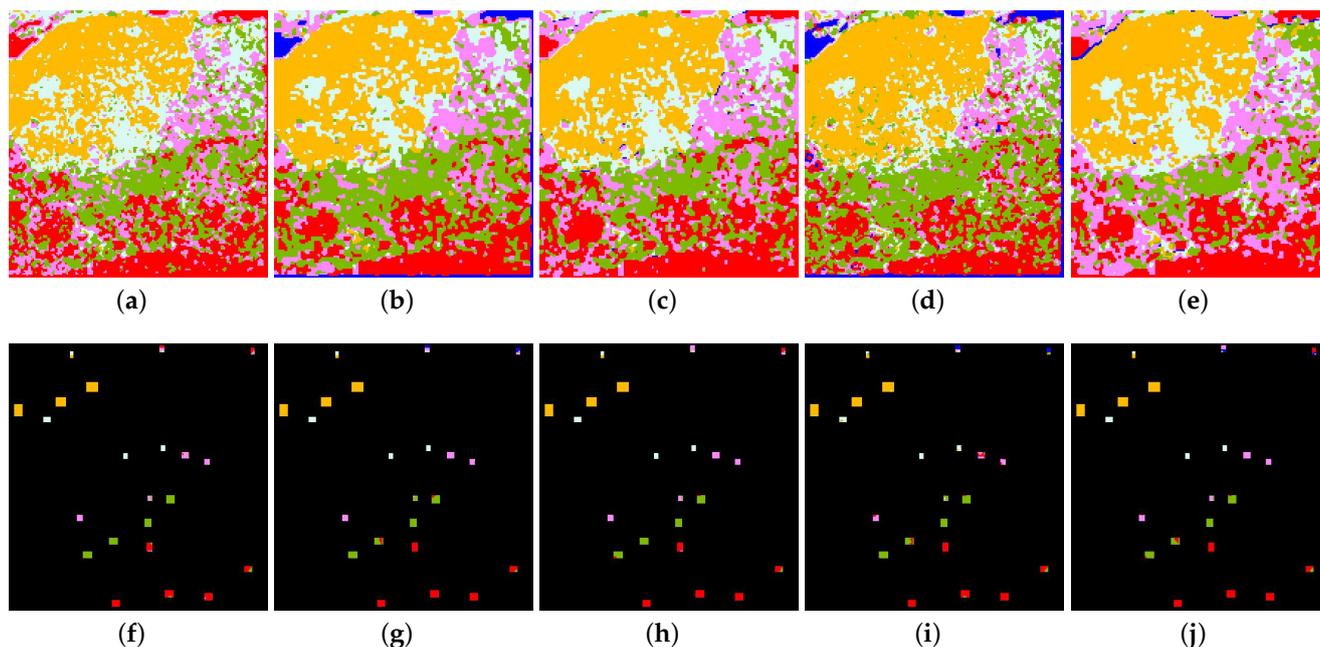
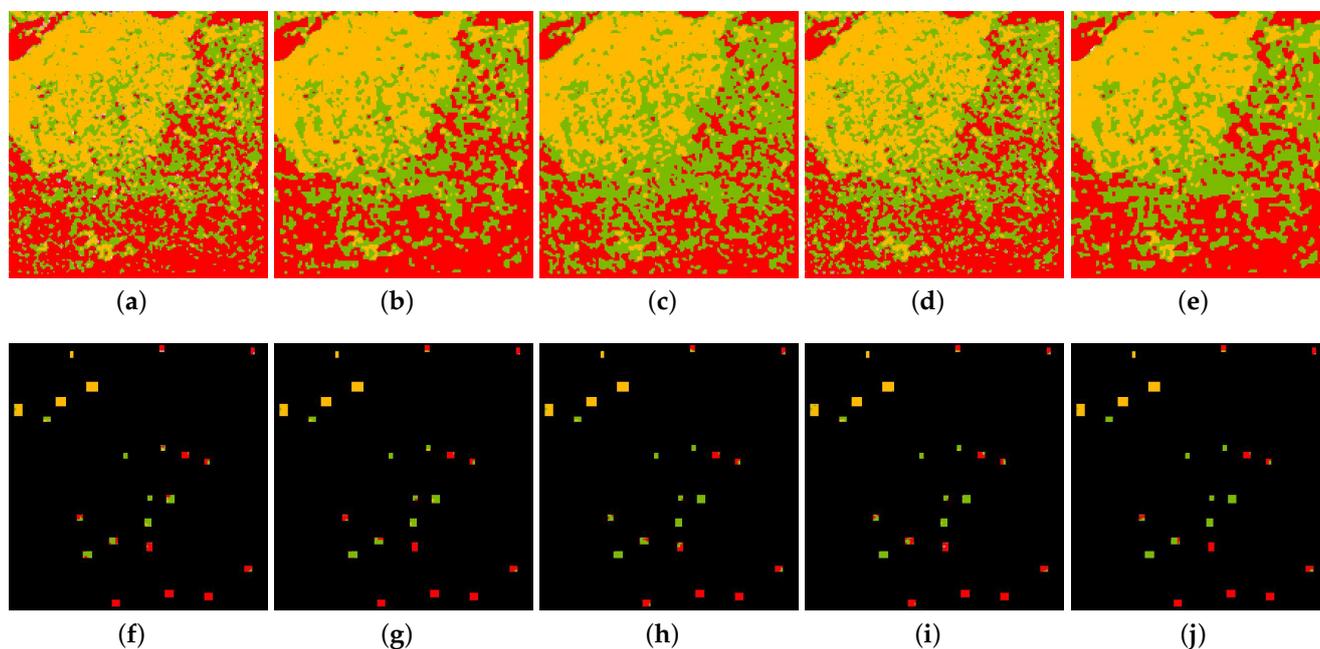


Figure 8. The OA, AA, and Kappa coefficient versus the level  $\gamma$  of speckle noise using different classification methods on CONVAIR Ice dataset. (a) OA vs.  $\gamma$  curves. (b) AA vs.  $\gamma$  curves. (c)  $\kappa$  vs.  $\gamma$  curves.



**Figure 9.** Classification maps of the CONVAIR Ice dataset with noise level  $\gamma = 0.02$  using different methods. (Top row) Results of the whole map classification. (Bottom row) Result overlaid with the ground-truth map. (a,f) RV-CNN-classified map. (b,g) RV-CNN-MRF-classified map. (c,h) RV-CNN-AL-MRF-classified map. (d,i) CV-CNN-classified map. (e,j) ACV-CWNN-MRF-classified map.



**Figure 10.** Classification maps of the CONVAIR Ice dataset with noise level  $\gamma = 0.16$  using different methods. (Top row) Results of the whole map classification. (Bottom row) Result overlaid with the ground-truth map. (a,f) RV-CNN-classified map. (b,g) RV-CNN-MRF-classified map. (c,h) RV-CNN-AL-MRF-classified map. (d,i) CV-CNN-classified map. (e,j) ACV-CWNN-MRF-classified map.

## 6. Conclusions

In this paper, we have proposed a novel PolSAR image classification method based on an active complex-valued convolutional-wavelet neural network framework by incorporating dual-tree complex wavelet transform and Markov random field. Our proposed

ACV-CWNN-MRF method has the following advantages. Firstly, we introduce a DT-CWT-constrained pooling layer into CV-CNN-based PolSAR classification to suppress speckle noise in a PolSAR image and maintain the SAR image structure information. Secondly, we combine the AL and MRF strategies into CV-CWNN to obtain the most informative unlabeled training samples and achieve the local correlation of neighboring pixels. Finally, experiments on three benchmark PolSAR datasets show that our proposed ACV-CWNN-MRF method can outperform some other state-of-the-art deep learning-based PolSAR image classification methods. We conclude that our fine-tuned ACV-CWNN-MRF method is an effective alternative method for PolSAR classification. We will continue to study how to better perform PolSAR classification on our own datasets in our future work.

**Author Contributions:** Conceptualization, L.L.; methodology, L.L.; software, L.L.; validation, L.L.; formal analysis, L.L.; investigation, L.L.; resources, L.L. and Y.L.; data curation, L.L.; writing—original draft preparation, L.L.; writing—review and editing, L.L. and Y.L.; visualization, L.L.; supervision, L.L.; project administration, L.L.; funding acquisition, L.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** The research was supported by National Natural Science Foundation of China under grant 42074225.

**Data Availability Statement:** Our experiments employ the open-source datasets introduced in [17].

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Cloude, S. *Polarisation: Applications in Remote Sensing*; OUP Oxford: Oxford, UK, 2009.
2. Lee, J.S.; Pottier, E. *Polarimetric Radar Imaging: From Basics to Applications*; CRC Press: Boca Raton, FL, USA, 2017.
3. Hänsch, R.; Hellwich, O. Skipping the real world: Classification of PolSAR images without explicit feature extraction. *ISPRS J. Photogramm. Remote Sens.* **2018**, *140*, 122–132. [[CrossRef](#)]
4. Durieux, L.; Lagabriele, E.; Nelson, A. A method for monitoring building construction in urban sprawl areas using object-based analysis of Spot 5 images and existing GIS data. *ISPRS J. Photogramm. Remote Sens.* **2008**, *63*, 399–408. [[CrossRef](#)]
5. Zhong, P.; Wang, R. A multiple conditional random fields ensemble model for urban area detection in remote sensing optical images. *IEEE Trans. Geosci. Remote Sens.* **2007**, *45*, 3978–3988. [[CrossRef](#)]
6. Besic, N.; Vasile, G.; Chanussot, J.; Stankovic, S. Polarimetric incoherent target decomposition by means of independent component analysis. *IEEE Trans. Geosci. Remote Sens.* **2014**, *53*, 1236–1247. [[CrossRef](#)]
7. 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](#)]
8. Kong, J.; Swartz, A.; Yueh, H.; Novak, L.; Shin, R. Identification of terrain cover using the optimum polarimetric classifier. *J. Electromagn. Waves Appl.* **1988**, *2*, 171–194.
9. Tao, M.; Zhou, F.; Liu, Y.; Zhang, Z. Tensorial independent component analysis-based feature extraction for polarimetric SAR data classification. *IEEE Trans. Geosci. Remote Sens.* **2014**, *53*, 2481–2495. [[CrossRef](#)]
10. 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](#)]
11. Ersahin, K.; Cumming, I.G.; Ward, R.K. Segmentation and classification of polarimetric SAR data using spectral graph partitioning. *IEEE Trans. Geosci. Remote Sens.* **2009**, *48*, 164–174. [[CrossRef](#)]
12. Bi, H.; Sun, J.; Xu, Z. Unsupervised PolSAR image classification using discriminative clustering. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3531–3544. [[CrossRef](#)]
13. Antropov, O.; Rauste, Y.; Astola, H.; Praks, J.; Häme, T.; Hallikainen, M.T. Land cover and soil type mapping from spaceborne PolSAR data at L-band with probabilistic neural network. *IEEE Trans. Geosci. Remote Sens.* **2013**, *52*, 5256–5270. [[CrossRef](#)]
14. 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](#)]
15. Liu, F.; Jiao, L.; Hou, B.; Yang, S. POL-SAR image classification based on Wishart DBN and local spatial information. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 3292–3308. [[CrossRef](#)]
16. Jiao, L.; Liu, F. Wishart deep stacking network for fast POLSAR image classification. *IEEE Trans. Image Process.* **2016**, *25*, 3273–3286. [[CrossRef](#)] [[PubMed](#)]
17. 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](#)]
18. Bi, H.; Sun, J.; Xu, Z. A graph-based semisupervised deep learning model for PolSAR image classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *57*, 2116–2132. [[CrossRef](#)]

19. Tuia, D.; Ratle, F.; Pacifici, F.; Kanevski, M.F.; Emery, W.J. Active learning methods for remote sensing image classification. *IEEE Trans. Geosci. Remote Sens.* **2009**, *47*, 2218–2232. [[CrossRef](#)]
20. Tuia, D.; Volpi, M.; Copa, L.; Kanevski, M.; Munoz-Mari, J. A survey of active learning algorithms for supervised remote sensing image classification. *IEEE J. Sel. Top. Signal Process.* **2011**, *5*, 606–617. [[CrossRef](#)]
21. Samat, A.; Gamba, P.; Du, P.; Luo, J. Active extreme learning machines for quad-polarimetric SAR imagery classification. *Int. J. Appl. Earth Obs. Geoinf.* **2015**, *35*, 305–319. [[CrossRef](#)]
22. Liu, W.; Yang, J.; Li, P.; Han, Y.; Zhao, J.; Shi, H. A novel object-based supervised classification method with active learning and random forest for PolSAR imagery. *Remote Sens.* **2018**, *10*, 1092. [[CrossRef](#)]
23. Bi, H.; Xu, F.; Wei, Z.; Xue, Y.; Xu, Z. An active deep learning approach for minimally supervised PolSAR image classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 9378–9395. [[CrossRef](#)]
24. Li, J.; Bioucas-Dias, J.M.; Plaza, A. Spectral–spatial classification of hyperspectral data using loopy belief propagation and active learning. *IEEE Trans. Geosci. Remote Sens.* **2012**, *51*, 844–856. [[CrossRef](#)]
25. Cao, X.; Yao, J.; Xu, Z.; Meng, D. Hyperspectral image classification with convolutional neural network and active learning. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 4604–4616. [[CrossRef](#)]
26. De Silva, D.; Fernando, S.; Piyatilake, I.; Karunaratne, A. Wavelet based edge feature enhancement for convolutional neural networks. In Proceedings of the Eleventh International Conference on Machine Vision (ICMV 2018), Munich, Germany, 1–3 November 2018; International Society for Optics and Photonics: Bellingham, WA, USA, 2019; Volume 11041, p. 110412R.
27. Liu, P.; Zhang, H.; Lian, W.; Zuo, W. Multi-level wavelet convolutional neural networks. *IEEE Access* **2019**, *7*, 74973–74985. [[CrossRef](#)]
28. Sahito, F.; Zhiwen, P.; Ahmed, J.; Memon, R.A. Wavelet-integrated deep networks for single image super-resolution. *Electronics* **2019**, *8*, 553. [[CrossRef](#)]
29. Guberman, N. On complex valued convolutional neural networks. *arXiv* **2016**, arXiv:1602.09046.
30. Tygert, M.; Bruna, J.; Chintala, S.; LeCun, Y.; Piantino, S.; Szlam, A. A mathematical motivation for complex-valued convolutional networks. *Neural Comput.* **2016**, *28*, 815–825. [[CrossRef](#)]
31. Trabelsi, C.; Bilaniuk, O.; Zhang, Y.; Serdyuk, D.; Subramanian, S.; Santos, J.; Mehri, S.; Rostamzadeh, N.; Bengio, Y.; Pal, C. Deep Complex Networks. *arXiv* **2018**, arXiv:1705.09792.
32. Benvenuto, N.; Piazza, F. On the complex backpropagation algorithm. *IEEE Trans. Signal Process.* **1992**, *40*, 967–969. [[CrossRef](#)]
33. Lau, M.M.; Lim, K.H.; Gopalai, A.A. Malaysia traffic sign recognition with convolutional neural network. In Proceedings of the 2015 IEEE International Conference on Digital Signal Processing (DSP), Singapore, 21–24 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1006–1010.
34. Kingsbury, N. Complex wavelets for shift invariant analysis and filtering of signals. *Appl. Comput. Harmon. Anal.* **2001**, *10*, 234–253. [[CrossRef](#)]
35. Baraniuk, R.; Kingsbury, N.; Selesnick, I. The dual-tree complex wavelet transform—a coherent framework for multiscale signal and image processing. *IEEE Signal Process. Mag.* **2005**, *22*, 123–151.
36. Kingsbury, N. Image processing with complex wavelets. *Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* **1999**, *357*, 2543–2560.
37. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536.
38. Bengio, Y. Practical recommendations for gradient-based training of deep architectures. In *Neural Networks: Tricks of the Trade*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 437–478.
39. MacKay, D.J. Information-based objective functions for active data selection. *Neural Comput.* **1992**, *4*, 590–604.
40. Luo, T.; Kramer, K.; Goldgof, D.B.; Hall, L.O.; Samson, S.; Rensen, A.; Hopkins, T.; Cohn, D. Active learning to recognize multiple types of plankton. *J. Mach. Learn. Res.* **2005**, *6*, 589–613.
41. Settles, B. Active learning. *Synth. Lect. Artif. Intell. Mach. Learn.* **2012**, *6*, 1–114.
42. Joshi, A.J.; Porikli, F.; Papanikolopoulos, N. Multi-class active learning for image classification. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 2372–2379.
43. Farag, A.A.; Mohamed, R.M.; El-Baz, A. A unified framework for map estimation in remote sensing image segmentation. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 1617–1634.
44. Boykov, Y.; Veksler, O.; Zabih, R. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.* **2001**, *23*, 1222–1239.
45. Kolmogorov, V. Convergent tree-reweighted message passing for energy minimization. In Proceedings of the International Workshop on Artificial Intelligence and Statistics, Bridgetown, Barbados, 6–8 January 2005; pp. 182–189.
46. Yedidia, J.S.; Freeman, W.T.; Weiss, Y. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Trans. Inf. Theory* **2005**, *51*, 2282–2312.
47. Cloude, S.R.; Pottier, E. A review of target decomposition theorems in radar polarimetry. *IEEE Trans. Geosci. Remote Sens.* **1996**, *34*, 498–518.

48. Cao, X.; Zhou, F.; Xu, L.; Meng, D.; Xu, Z.; Paisley, J. Hyperspectral image classification with Markov random fields and a convolutional neural network. *IEEE Trans. Image Process.* **2018**, *27*, 2354–2367. [[CrossRef](#)] [[PubMed](#)]
49. Xie, H.; Pierce, L.E.; Ulaby, F.T. Statistical properties of logarithmically transformed speckle. *IEEE Trans. Geosci. Remote Sens.* **2002**, *40*, 721–727. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.