Merits of Complex-Valued Neural Networks for PolSAR image segmentation.

J. Agustin Barrachina[1, 2], Chengfang Ren[1], Christèle Morisseau[2], Gilles Vieillard[2], Jean-Phillipe Ovarlez[1, 2]

[1]SONDRA, CentraleSupélec, Université Paris-Saclay, 91192 Gif-sur-Yvette, France

[2]DEMR, ONERA, Université Paris-Saclay, 91120 Palaiseau, France

jose-agustin.barrachina@centralesupelec.fr, chengfang.ren@centralesupelec.fr,
christele.morisseau@onera.fr, gilles.vieillard@onera.fr, jean-philippe.ovarlez@onera.fr,

Résumé – Dans cet article, nous mettons en oeuvre deux réseaux de neurones, l'un à valeurs complexes et l'autre à valeurs réelles, de dimensions équivalentes, pour la segmentation d'images Polarimetric Synthetic Aperture Radar (PolSAR). Ces réseaux sont basés sur des architectures convolutives de type *U-Net*. Une comparaison statistique exhaustive entre ces réseaux est présentée pour l'image PolSAR de Flevoland, en prenant en entrée la matrice de cohérence. Les résultats montrent une meilleure classification par le réseau de neurones à valeurs complexes.

Abstract – In this paper, we implement two capacity equivalent neural networks, one complex-valued and the other one real-valued for Polarimetric Synthetic Aperture Radar (PolSAR) image segmentation. An exhaustive statistical comparison between these two networks are done over the Flevoland PolSAR dataset using the coherency matrix as input. Results show a better generatlization for the complex-valued architecture.

1 Introduction

Deep learning algorithms are becoming increasingly popular and have been applied to the classification of Polarimetric Synthetic Aperture Radar (PolSAR) images [1, 2]. Because radar data is typically complex-valued due to In Phase and Quadrature (I-Q) channels, many publications employ Complex-Valued Neural Network (CVNN) as an alternative to traditional Real-Valued Neural Network (RVNN) for radar applications [3, 4]. [5] uses a Complex-Valued MultiLayer Perceptron (CV-MLP) to implement one of the earliest experiments on PolSAR image categorization using deep learning. These findings are also confirmed by [6]. Following that, Complex-Valued Convolutional Neural Network (CV-CNN) are presented for performing PolSAR classification [7,8]. In particular, [9] achieves 97.66% validation accuracy (test accuracy not mentioned) whereas [10] obtains 96.2% performance for a CV-CNN model architecture on the same dataset used for this work. Also, [11] implements a 3D-CV-CNN and obtained 93.74% accuracy.

Because CNN models demand a constant input size to categorize each object, pixel-wise classification of Synthetic Aperture Radar (SAR) images, also known as semantic segmentation, has got a lot of attention in the previous decade as a way to get rid of the input size constraint in the classification process. Complex-Valued Fully Convolutional Neural Network (CV-FCNN), based on U-net architecture [12], is specifically intended to segment SAR pictures in the latest advancement of neural networks. Indeed, such networks achieve *state-of-the-art* performance [13, 14].

In this paper we undertake an extensive comparison for the Flevoland PolSAR dataset, between a pair of CVNN and RVNN *state-of-the-art* inspired architectures. For each model, we run multiple iterations in order to deduce statistical metrics that allow us to claim a performance difference.

The dataset and the pre-processing approaches are presented in section 2. The model architectures utilized for the experiments are detailed in section 3. Finally, in section 4, the neural networks peformances are compared.

2 PolSAR Dataset

PolSAR images are acquired from single look complex data measured in the horizontal (H) and vertical (V) transmit/receive polarimetric channels known as the Sinclair scattering matrix :

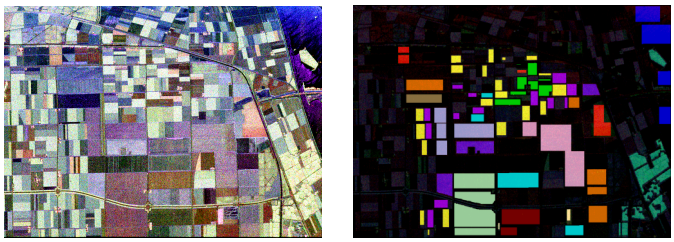
$$S = \begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix}. \tag{1}$$

For each pixel of the Synthetic Aperture Radar (SAR) image, the 4 components are usually expressed in Pauli basis as one complex vector $\boldsymbol{k} \in \mathbb{C}^3$ [15], so that

$$\boldsymbol{k} = 2^{-1/2}\left[S_{HH} + S_{VV}, S_{HH} - S_{VV}, 2\,S_{HV}\right]^T. \tag{2}$$

The Hermitian coherency matrix can then be built as $\boldsymbol{T} = \frac{1}{n}\sum_j^n \boldsymbol{k}_j\,\boldsymbol{k}_j^H$ where the operator ^{H} stands for complex conjugate operation and where n is the number of neighboring pixels chosen in a boxcar around the considered one. The coherency matrix is then used as input to the networks. For the real-valued network, both real and imaginary parts are injected separately to the network.

Here we perform our networks comparison on NASA/Jet Propulsion Laboratory (JPL) AIRSAR dataset over Flevoland (figure 1a), an agriculture area on the Netherlands. The ground truth is obtained from [16]. The dataset presents fifteen classes as it can be seen on figure 1b.



(a) Flevoland image (b) Ground truth

FIGURE 1 – Flevoland image and ground truth. **A** Steambeans ; **B** Peas ; **C** Forest ; **D** Lucerne ; **E** Wheat ; **F** Beet ; **G** Potatoes ; **H** Bare Soil ; **I** Grass ; **J** Rapeseed ; **K** Barley ; **L** Wheat 2 ; **M** Wheat 3 ; **N** Water ; **O** Buildings

We operate a sliding window on the image [17] to generate the dataset with the parameters used in [14] for stride and window size. With this method, we obtain smaller images patches of size 128×128. 80% of the generated patches are used as training and 10% for each validation and test sets.

3 Network Architectures

Difficulties in deploying CVNN models in practice have hampered the field's growth to this point [18]. For this study, an open-source and well-documented tool has been built that permits and facilitates the creation of a wide range of CVNN architectures for the community to further utilize [19]. This tool also allows to instantiate a real-equivalent model in terms of trainable parameters from a complex-valued network model while maintaining the same architecture with a consistent aspect ratio as defined in [6].

Our FCNN architecture utilizes the above described toolbox and is influenced by the models in reference [14] for both real- and complex-valued models, as these are the higher claimed accuracy for this application. The model architecture is composed of a downsampling or feature extraction part and the upsampling part. The downsampling part presents two sub-modules which are represented on figure 2 in green and red colours whereas the upsampling part, in term, has a combination of other two sub-modules, the second one being the same green sub-module present on the downsampling section. The first sub-module (yellow) is a max-unpooling module as explained on [20]. The green sub-module is a combination of a convolution layer, a BatchNormalization (BN) (complex adaptations explained on [21] sections 3.2 and 3.5) and Rectified Linear Unit (ReLU).

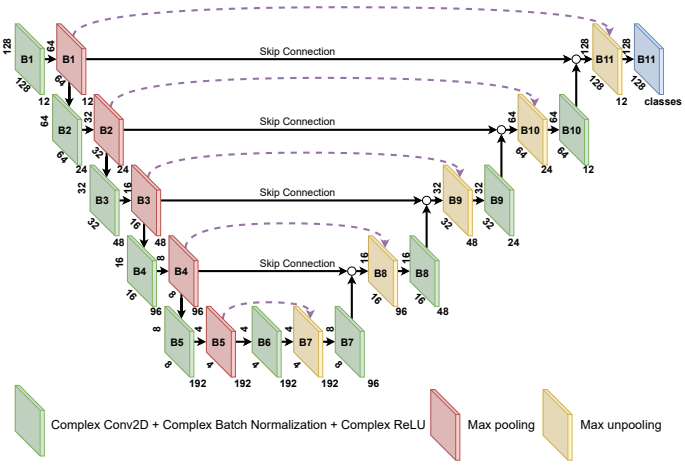


FIGURE 2 – Chosen Complex-Valued Fully Convolutional Neural Network architecture.

The number convolutional filters for each layer is shown in figure 2, being all 3×3 in size. For the real case, the number of kernels is multiplied by $\sqrt{2}$ and rounded to the nearest integer number to produce a real model, as reported on [6], where it was demonstrated this method generates a real-valued capacity equivalent model in terms of real-valued trainable parameters. As a result, we can do an accurate comparison between CVNN and RVNN.

The red sub-module is a max pooling layer, whose main objective is to shrink the image into smaller ones by keeping only the maximum value within a small window, in our case, of size 2×2. For the complex case, the absolute value of the complex number is used for comparison as proposed in [10]. This layer complements with the max-unpooling sub-module (yellow) which receives the locations where the maximum value was found. The max-unpooling layer enlarges the input image by placing their image pixels according to the locations received from the corresponding max-pooling layer represented by the dashed arrow in figure 2 [20].

The complex-valued activation function $\mathbb{C}$ReLU(.) consists in applying the well known real-valued function ReLU(.) to both the real and imaginary part separately. Softmax activation function is used for the output layer also applied to both the real and imaginary part separately for the complex case.

Categorical cross-entropy is implemented as the loss function which, for the complex network is computed twice, using first the real part and then the imaginary part as the prediction result. An average of the two error values is then calculated to be optimized using Adam optimizer with a learning rate of 0.01. It is worth noticing that pixels without labels (black areas on Figure 1b) are not taken into account either for loss computation or for the accuracy metric.

4 Experiment results

		CV-FCNN	RV-FCNN
OA	median	**99.80** $\pm$ 0.02	99.67 $\pm$ 0.03
	mean	99.79 $\pm$ 0.01	99.66 $\pm$ 0.02
	IQR	99.74 – 99.84	99.58 – 99.74
	full range	99.58 – 99.91	99.38 – 99.88
AA	median	**98.55** $\pm$ 0.38	98.25 $\pm$ 0.44
	mean	98.35 $\pm$ 0.19	97.87 $\pm$ 0.23
	IQR	97.84 – 99.52	97.08 – 99.10
	full range	94.20 – 99.87	93.07 – 99.75

TABLE 1 – FCNN test accuracy results (%).

Simulations on both CV-FCNN and RV-FCNN architectures, are performed 50 times each in order to infer statistical analysis over the results. On each iteration, the train, validation and test sets are randomly sampled so no two simulations have the same dataset split. Each training includes 1000 epochs. The median error is computed as in [22] which claims that if median intervals do not overlap, there is a 95% confidence that their values differ. The confidence interval of the mean is calculated for a confidence level of 99%.

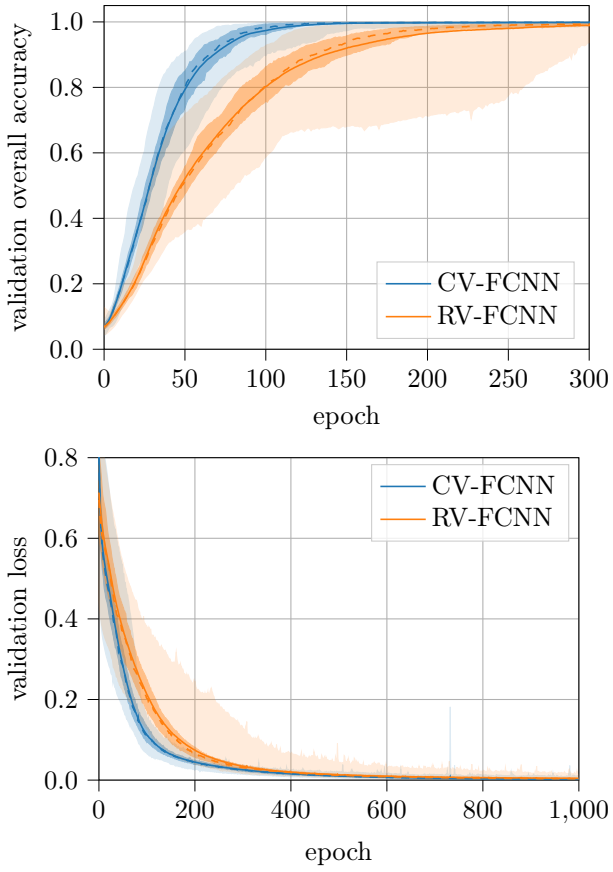
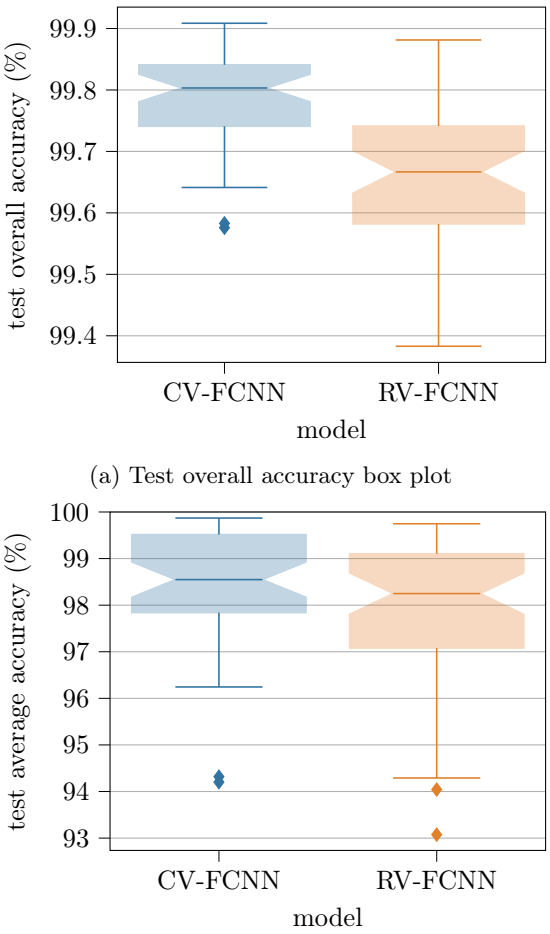


FIGURE 3 – Validation evolution per epoch

Statistical indicators of both the Overall Accuracy (OA), which is the ratio of the number of correctly predicted pixels with respect to the total pixels, and the Average Accuracy (AA), which is an average of the accuracy for each class independently, are summarized in Table 1 for the test set. Although high accuracy makes performance difference between the models seem small, confidence intervals remains far apart which allows to conclude that CV-FCNN generalizes better than RV-FCNN. 75% of CV-FCNN simulations achieve more than 99.74% OA whereas only 25% of RV-FCNN iterations achieve it.

The validation accuracy progression over epochs is shown in Figure 3, for the accuracy, only the first 300 epochs are depicted as the graph does not change much after that. It can be appreciated that complex-valued models converge quicker than the real-valued ones.

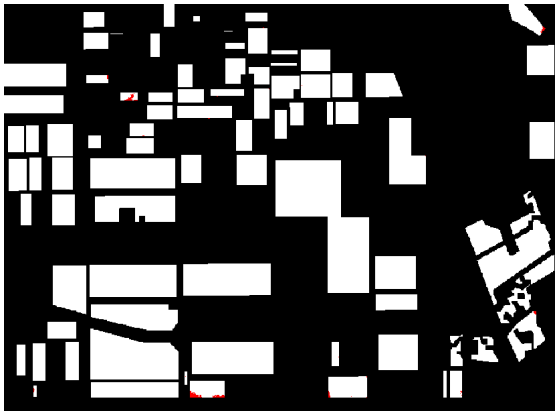


(a) Test overall accuracy box plot

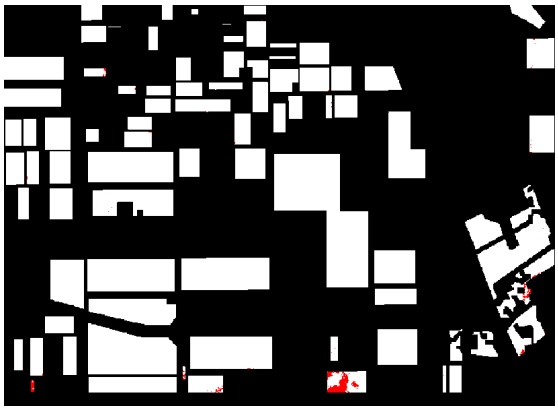
(b) Test average accuracy box plot

FIGURE 4 – Test accuracy box plot

Figure 4 shows the box plot for both OA and AA. This plots allow to better appreciate the outliers which is not possible to appreciate on table 1. Finally, 5 shows the median OA of both models correctly predicted pixels in white and the wrong predictions in red.



(a) CV-FCNN



(b) RV-FCNN

FIGURE 5 – Median overall accuracy models predictions

5 Conclusions

In this work, we propose two complex- and real-valued FCNN architectures equivalent in terms of training parameters, implemented using our open-source toolbox [19]. We illustrate the semantic segmentation performance of these models on well-known open-source Flevoland PolSAR database. The experimentation results highlight a better performance of CV-FCNN compared to its equivalent-RV-FCNN.

Références

[1] H. Parikh, S. Patel, and V. Patel, "Classification of SAR and PolSAR images using deep learning : A review," *International Journal of Image and Data Fusion*, vol. 11, no. 1, pp. 1–32, 2020.

[2] B. Konishi, A. Hirose, and R. Natsuaki, "Complex-valued reservoir computing for interferometric SAR applications with low computational cost and high resolution," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 7981–7993, 2021.

[3] A. Hirose, *Complex-valued neural networks : Advances and applications*, vol. 18. John Wiley & Sons, 2013.

[4] J. Bassey, L. Qian, and X. Li, "A survey of complex-valued neural networks," *arXiv preprint arXiv :2101.12249*, 2021.

[5] R. Hänsch and O. Hellwich, "Classification of polarimetric SAR data by complex valued neural networks," in *ISPRS workshop high-resolution earth imaging for geospatial information*, vol. 38, pp. 4–7, 2009.

[6] J. A. Barrachina, C. Ren, G. Vieillard, C. Morisseau, and J.-P. Ovarlez, "About the equivalence between complex-valued and real-valued fully connected neural networks - application to PolInSAR images," in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2021.

[7] R. Hänsch and O. Hellwich, "Complex-valued convolutional neural networks for object detection in PolSAR data," in *8th European Conference on Synthetic Aperture Radar*, pp. 1–4, VDE, 2010.

[8] J. Zhao, M. Datcu, Z. Zhang, H. Xiong, and W. Yu, "Contrastive-regulated CNN in the complex domain : A method to learn physical scattering signatures from flexible PolSAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 12, pp. 10116–10135, 2019.

[9] Y. Zhou, H. Wang, F. Xu, and Y.-Q. Jin, "Polarimetric SAR image classification using deep convolutional neural networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 12, pp. 1935–1939, 2016.

[10] Z. Zhang, H. Wang, F. Xu, and Y.-Q. Jin, "Complex-valued convolutional neural network and its application in polarimetric SAR image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 12, pp. 7177–7188, 2017.

[11] X. Tan, M. Li, P. Zhang, Y. Wu, and W. Song, "Complex-valued 3-d convolutional neural network for polsar image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 6, pp. 1022–1026, 2019.

[12] O. Ronneberger, P. Fischer, and T. Brox, "U-net : Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015.

[13] Y. Li, Y. Chen, G. Liu, and L. Jiao, "A novel deep fully convolutional network for PolSAR image classification," *Remote Sensing*, vol. 10, no. 12, p. 1984, 2018.

[14] Y. Cao, Y. Wu, P. Zhang, W. Liang, and M. Li, "Pixel-wise PolSAR image classification via a novel complex-valued deep fully convolutional network," *Remote Sensing*, vol. 11, no. 22, p. 2653, 2019.

[15] J. S. Lee and E. Pottier, *Polarimetric radar imaging : from basics to applications*. CRC press, 2017.

[16] X. Liu, L. Jiao, and F. Liu, "PolSF : PolSAR image dataset on San Francisco," *arXiv preprint arXiv :1912.07259*, 2019.

[17] Y. Li, Y. Chen, G. Liu, and L. Jiao, "A novel deep fully convolutional network for PolSAR image classification," *Remote Sensing*, vol. 10, no. 12, 2018.

[18] N. Mönning and S. Manandhar, "Evaluation of complex-valued neural networks on real-valued classification tasks," *arXiv preprint arXiv :1811.12351*, 2018.

[19] J. A. Barrachina, "Complex valued neural networks (cvnn)," Oct. 2021. https ://doi.org/10.5281/zenodo.4452131.

[20] I. Zafar, G. Tzanidou, R. Burton, N. Patel, and L. Araujo, *Hands-on convolutional neural networks with TensorFlow : Solve computer vision problems with modeling in TensorFlow and Python*. Packt Publishing Ltd, 2018.

[21] C. Trabelsi, A. Bilaniuk, Y. Zhang, D. Serdyuk, S. Subramanian, J. F. Santos, S. Mehri, N. Rostamzadeh, Y. Bengi, and J. Pal, C., "Deep complex networks," 2018.

[22] J. M. Chambers, *Graphical methods for data analysis*. CRC Press, 2018.