

FULLY COMPLEX-VALUED DEEP LEARNING MODEL FOR VISUAL PERCEPTION

Aniruddh Sikdar^{1†}, Sumanth Udupa^{2†}, Suresh Sundaram²

¹Robert Bosch Centre for Cyber Physical Systems, Indian Institute of Science, Bengaluru, India

²Department of Aerospace Engineering, Indian Institute of Science, Bengaluru, India

{aniruddhss, sumanthudupa, vssuresh} @iisc.ac.in

ABSTRACT

Deep learning models operating in the complex domain are used due to their rich representation capacity. However, most of these models are either restricted to the first quadrant of the complex plane or project the complex-valued data into the real domain, causing a loss of information. This paper proposes that operating entirely in the complex domain increases the overall performance of complex-valued models. A novel, fully complex-valued learning scheme is proposed to train a Fully Complex-valued Convolutional Neural Network (FC-CNN) using a newly proposed complex-valued loss function and training strategy. Benchmarked on CIFAR-10, SVHN, and CIFAR-100, FC-CNN has a 4-10% gain compared to its real-valued counterpart, with the same number of parameters. It achieves comparable performance to state-of-the-art complex-valued models on CIFAR-10 and SVHN with fewer parameters. For the CIFAR-100 dataset, it achieves state-of-the-art performance with 25% fewer parameters. FC-CNN shows better training efficiency and much faster convergence than all the other models.

Index Terms— Deep learning, Image classification, complex-valued neural network

1. INTRODUCTION

Deep learning is used for various tasks such as speech processing [1], robotics, signal processing, image processing, and sonar [2], which contain complex numbers, but the majority of the models rarely use complex numbers. Complex-valued data can exist in: (1) natural form, involving applications using Synthetic Aperture Radar (SAR) [3], Magnetic Resonance Imaging (MRI), and telecommunication, (2) complex-valued representation of real-valued data, such as Fourier transform [4] and spectrum-based methods [5], and (3) physics application such as optical physics [6], where both amplitude and phase information is essential. Complex-valued models (CV-models) have certain advantages over their real-valued counterparts, such as richer representation ability [7], easier optimization [8], efficient multi-task learning [9], superior MRI reconstruction [10], and robust memory

retrieval mechanism [11].

CV models can be used for standard computer vision tasks, such as image classification, because of their rich representation capacity in the complex domain. DCN [12] extends real-valued deep learning models to the complex domain by proposing complex-valued batch norm, weight initialization, and CReLU activation function. It operates only in the first quadrant of the complex plane. SurReal [13] tackles complex numbers from a manifold deep learning perspective but does not exploit the complex-valued representation. CDS models [14] introduce equivariant and invariant counterparts of common neural network layers to address complex scaling.

All the models mentioned above do not operate entirely in the complex domain and project data to the real domain to make predictions and compare with ground truth labels. This causes a loss of phase information, hence, the complex domain's representation capacity is not used optimally. It has been shown in [6], that more information is stored in the phase component compared to the amplitude component in the parameter subspace, proving the heterogeneous nature of neural networks. Operating completely in the complex domain to train the model may improve the overall generalization ability and performance of CV models for visual perception.

In this paper, a Fully Complex-valued Convolutional Neural Network (FC-CNN) is proposed for image classification with the same number of parameters as its real-valued counterpart. A novel complex-valued learning scheme is proposed to train FC-CNN entirely in the complex domain, taking phase information into account when updating network parameters. The key contributions can be summarized as follows: (1) A novel fully complex-valued learning scheme is proposed using orthogonal decision boundary theory [15] to directly convert shallow real-valued networks to fully complex-valued networks, which improves the performance while maintaining the same model complexity. (2) A new regularized complex-valued loss function and a two-step training strategy for FC-CNN are proposed to address overfitting, leading to faster learning and better convergence. (3) FC-CNN outperforms its real-valued counterpart on RGB and other encodings. It achieves state-of-the-art performance on CIFAR-10, CIFAR-100, and SVHN datasets, with less parameters.

[†]Equal contribution of authors.

2. FULLY COMPLEX-VALUED CONVOLUTIONAL NEURAL NETWORK

In this section, the Fully Complex-valued Convolutional Neural Network (FC-CNN) is explained in detail. Complex-valued prerequisites are briefly discussed first, followed by network architecture and the learning scheme.

2.1. Complex-valued Prerequisites

Complex-valued convolution operation Complex-valued convolution operator is defined for complex-valued weight W and complex-valued input I as follows,

$$\mathbf{W} * \mathbf{I} = (W_R + iW_I) * (I_R + iI_I) \quad (1)$$

where W_R , W_I denote the real and the imaginary parts of the complex-valued weight, and I_R , I_I denote the real and imaginary parts of the complex-valued input. Gauss's complex multiplication trick is used to increase the speed of the convolution operation by 25%.

Complex-valued activation function Complex differentiable functions are called analytic or holomorphic if the real and imaginary parts of the complex function follow Cauchy-Riemann differential equations. A sufficient condition for complex-valued activation functions to be compatible with complex backpropagation [8] is that they should be differentiable with respect to each complex-valued parameter and its complex conjugate [16]. Although there are numerous complex-valued activation functions, mainly CReLU [12] and complex cardioid [17] activation functions are considered as both are extensions of the ReLU function in the complex plane. CReLU can be defined as applying the ReLU function on real and imaginary parts separately, as shown below,

$$CRelu(z) = ReLU(R(z)) + i \cdot ReLU(Im(z)), \quad (2)$$

where z is a complex number, $R(z)$ and $Im(z)$ represent the real and imaginary components of the complex number respectively. While using CReLU, the complex plain is restricted to only the first quadrant. The complex cardioid is defined as,

$$f(z) = \frac{1}{2}(1 + \cos(\angle z)) \cdot z \quad (3)$$

where z is a complex number, $\angle z$ denotes the phase of the complex number. The cardioid function operates in the whole complex-plain while preserving phase and scaling the magnitude.

2.2. Fully complex-valued learning scheme

Network architecture FC-CNN model has 3 convolutional layers (stride 2) and 1 fully connected layer, as shown in Table 1. The group setting for the second and third convolutional layers is 2 and 4, respectively. Depthwise-separable convolution is used as a learnable pooling layer to reduce the

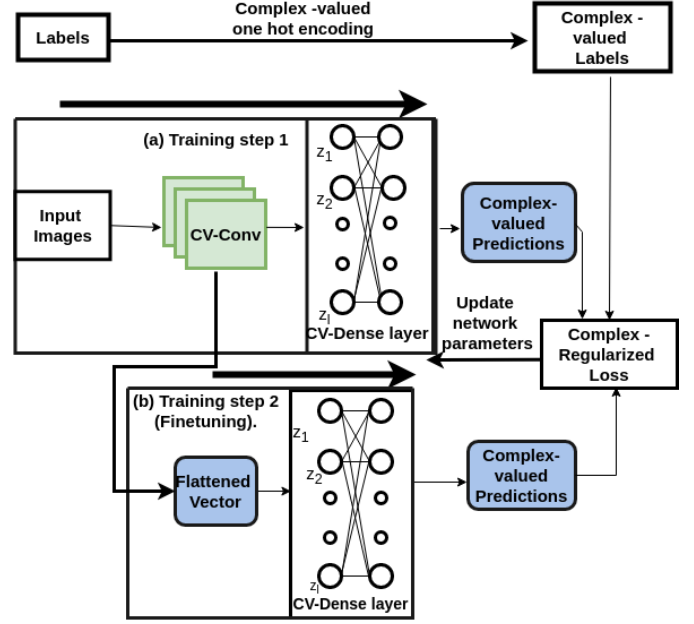


Fig. 1. Fully complex-valued learning scheme with a two-stage training strategy.

feature map size, with a group setting of 64. The real and imaginary components of the complex-valued weights are initialized using a uniform distribution. The cardioid activation function is used to operate in the whole complex plane and not be restricted to any particular quadrant. The output of the complex-valued fully connected layer is a complex-valued entity [18][19]. The last layer is modified according to the number of classes for a given dataset.

Complex-valued one hot encoding Real-valued labels are one hot encoded to complex-valued labels to compare with the complex-valued predictions made by the fully complex-valued network. Let $\{(x_1, c_1), (x_2, c_2) \dots (x_t, c_t) \dots (x_n, c_n)\}$ be the input data, where x_t represents the t -th input image and c_t represents its label [20][21]. One-hot encoding $y_k^t = \{y_1^t, y_2^t, \dots, y_k^t, \dots, y_n^t\}$ for its corresponding label c_t , is given by,

$$y_k^t = \begin{cases} 1 + i, & \text{if } c_t = k \\ -1 - i, & \text{otherwise.} \end{cases}$$

Orthogonal decision boundary theory A fully complex-valued network consists of two hypersurfaces as its decision boundary - real and imaginary [15]. These real and imaginary hypersurfaces are orthogonal to each other [22]. Complex-valued one hot encoding of the ground truth labels is projected in the output feature space. Either the real or imaginary hypersurface can be used to classify the on and off values of the labels. Using orthogonal decision boundary theory, the real components of the complex-valued output can be directly compared with the real components of the complex-valued one-hot encoded labels to compute the accuracy.

Layer Type	Input shape	Kernel	Stride	Output shape
CV - Conv1	[3,32,32]	3x3	2	[32,16,16]
Cardioid	[32,16,16]	-	-	[32,16,16]
CV - Conv2	[32,16,16]	3x3	2	[64,8,8]
Cardioid	[64,8,8]	-	-	[64,8,8]
CV - Conv3	[64,8,8]	3x3	2	[64,4,4]
Cardioid	[64,4,4]	-	-	[64,4,4]
CV-Depthwise Conv	[64,4,4]	4x4	2	[128,1,1]
Flatten Layer	[128,1,1]	-	-	[128]
CV - Linear layer	[128]	-	-	[10]

Table 1. FC-CNN model architecture for 10 output classes.

Regularized complex-valued hinge loss A new loss function is proposed to compute the error between complex-valued predictions and complex-valued labels. This helps the phase information to be taken into account when updating the network parameters. The complex-valued hinge loss e [21] is given by ,

$$e = \begin{cases} 0 & \text{if } \text{Re}(y_i^t) \cdot \text{Re}(\hat{y}_i^t) > 1 \\ (y_i^t) - (\hat{y}_i^t) & \text{otherwise.} \end{cases}$$

where (y_i^t) is the complex-valued one hot encoding and (\hat{y}_i^t) is the complex-valued predictions. FC-CNN has an over-fitting issue, which is why the loss function is modified to have a regularizing effect for the correctly predicted samples. Two terms, error threshold e_{thr} and maximum threshold e_M are defined, where error threshold e_{thr} is updated after every epoch and is given by,

$$e_{thr} = e^{-(0.05) \cdot (epoch)} \quad (4)$$

The term e_M represents the maximum value of error from all the samples in the batch, and is defined as,

$$e_M = \max(|(e(y_i^t, \hat{y}_i^t))|) \quad (5)$$

where $|\cdot|$ represents the magnitude of complex numbers and maximum operator returns the maximum value from all the samples in the batch. The following condition is used to update e ,

$$e = \begin{cases} 0 & \text{if } e_M < e_{thr} \\ e & \text{if } e_M > e_{thr} \end{cases}$$

For misclassified samples, the complex-valued hinge loss e remains the same. Real-valued loss function E is defined as the product of complex-valued hinge loss e and its complex conjugate[18], and is given by

$$E = \frac{1}{2n} (e^H)(e) \quad (6)$$

where n is the number of data samples.

Training strategy The fully complex-valued network (FC-CNN) is trained end-to-end using the proposed complex loss function and complex-valued labels, as shown in Fig.1(a) training step 1. Once the whole network is trained, the output of the final complex-valued convolutional layer of the last epoch is flattened and stored. The model is then finetuned in the second training step (Fig.1(b)), where only the complex-valued linear layer is re-trained. During this step, the stored flattened vector is fed to the complex-valued linear layer and trained for the same number of epochs with the same loss function. During inference time, the model consists of complex-valued convolutional layers and the finetuned complex-valued linear layer.

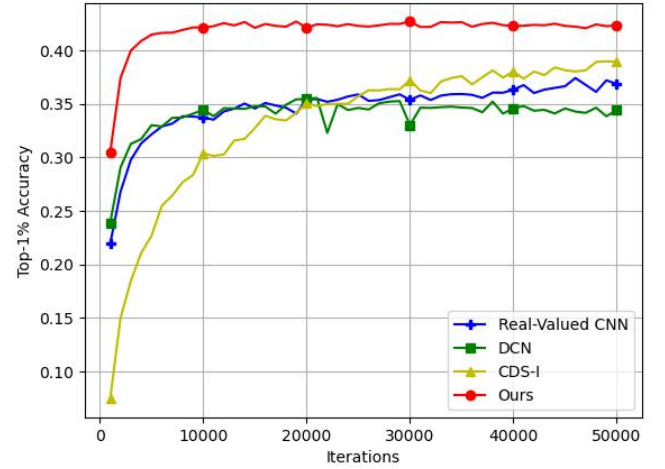


Fig. 2. Performance curves of test accuracy of real-valued CNN, CV-models and FC-CNN on CIFAR-100 (RGB) dataset.

3. EXPERIMENTAL RESULTS

The experiments are conducted on three benchmark datasets, i.e., CIFAR-10 [23], CIFAR-100, and SVHN [24]. CIFAR-10 and CIFAR-100 contain 10 and 100 classes, with 50000 training images and 10000 testing images, respectively. SVHN contains ten classes, with 73,257 training images and 26,032 test images. For a fair comparison, real-valued CNN and DCN both have the same network architecture as FC-CNN and are trained using cross-entropy loss. Relu activation function is used for the real model, and CReLU activation is used for the DCN model. Sur-real [13], CDS Type - I, and E models [14] have the same network architecture as mentioned in [14]. Note that as Sur-real's code is not publicly available, the results mentioned in Table 2 were borrowed from the re-

Method	# Param		CIFAR-10			SVHN			CIFAR-100	
		RGB	LAB	Sliding	RGB	LAB	Sliding	RGB	LAB	Sliding
Real-valued CNN	22,634/ 34,244	67.27	62.52	65.0	87.3	86.58	87.11	37.81	33.49	36.48
DCN	23,914/ 47,044	65.64	63.72	66.94	86.80	85.77	86.56	35.88	35.25	34.85
Sur-Real	35,274/-	50.68	53.02	54.61	80.51	53.48	80.79	23.57	25.97	26.66
CDS Type - I	24,241/ 47281	69.18	67.89	70.28	89.40	89.27	89.81	40.2	37.94	39.27
CDS Type - E	23,697/ 46737	69.98	67.7	69.5	88.75	74.38	90.50	40.11	38.91	40.97
FC-CNN	22,634/ 34,244	69.82	69.87	69.99	89.65	89.13	89.12	41.71	42.26	41.46

Table 2. Comparison of test accuracy between real-valued CNN, state of the art complex-valued and fully complex-valued model (FC-CNN) on CIFAR-10, CIFAR-100 and SVHN.

Method	CIFAR-100 (RGB)	FLOPs (MACS)
Real -valued CNN	37.81	1M
DCN (With CReLU)	35.88	1.545M
DCN (With Cardioid)	34.29	1.499M
FC-CNN (W/o Finetuning)	39.06	1.480M
FC-CNN (With Finetuning)	41.71	1.480M

Table 3. Comparison of test accuracy and FLOPs (forward pass) of real-valued, DCN and fully complex-valued (FC-CNN) models. FLOPs are reported during testing.

sults cited in [14].¹ All the models are trained with AdamW optimizer using momentum (0.99,0.999), with a learning rate of 0.001, batch size 256, and weight decay of 0.1.

Test accuracy of real-valued and complex-valued models are shown in Table 2 for real-valued RGB images and their corresponding complex-valued sliding [14] and LAB encoding [14] for all three datasets. The number of parameters for all the models drastically increases when changing the output categories from 10 to 100. The real-valued CNN and DCN models have comparable performance to each other, but for RGB data, real-valued CNN outperforms the DCN model. Comparing these networks to fully complex-valued network (FC-CNN), a significant improvement in the performance can be seen. For CIFAR-10 and SVHN datasets, FC-CNN achieves similar performance to the best-performing CDS model, with fewer parameters. For the CIFAR-100 dataset, FC-CNN outperforms its real-valued counterpart on RGB and all the encodings with the same number of parameters. It achieves state-of-the-art performance compared to other CV models with considerably less number of parameters. Fig.2 shows the test accuracy of real-valued, complex-valued, and FC-CNN models on CIFAR-100 (RGB). FC-CNN outperforms all the other models with much more efficient learning.

¹The results for CDS-I and E are reproduced from the github code : <https://github.com/sutkarsh/cds>. The results for FC-CNN are reproduced using the same code and same seed.

It converges within 15k iterations, while all the other models take considerably more training time. Compared to its real-valued counterpart, with the same number of parameters, the overall test accuracy and convergence are much better. Thus, it generalizes better on larger datasets than real-valued and CV models.

Table 3 shows the test accuracy and model complexity of real-valued, DCN, and FC-CNN networks on the CIFAR-100 dataset. FLOPs for the DCN and FC-CNN networks remain almost the same. FC-CNN performs better than both networks without the finetuning step. Using either CReLU or cardioid for the DCN model does not improve its performance compared to the real-valued network. Hence, we infer that operating in all quadrants of the complex plane is not enough, and the model must also train entirely in the complex domain and not be projected back to the real domain. The empirical results of FC-CNN justify this intuition. The two-stage training strategy, i.e., finetuning the CV-linear layer, shows a considerable gain in performance.

4. CONCLUSION

This paper proposes a Fully Complex-valued network for image classification of real-valued images and their complex-valued encodings. It is an extension of a shallow real-valued CNN to the complex domain with an equal number of parameters. A novel, fully complex-valued learning scheme is proposed for the FC-CNN to operate entirely in the complex domain and operate in all four quadrants of the complex domain, using a novel regularized loss function and training strategy. Experimental results on CIFAR-10, SVHN, and CIFAR-100 show that FC-CNN has a gain in accuracy of 4-10% compared to its real-valued counterpart on RGB, Sliding, and LAB encodings. Compared to the shallow DCN model, accuracy increases by 4-6% with 5% lesser computational complexity. Compared to other complex-valued models, it achieves comparable performance to the state-of-the-art network on CIFAR-10, SVHN. For CIFAR-100, it achieves state-of-the-art performance with almost 25% fewer parameters.

5. REFERENCES

- [1] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, “Wavenet: A generative model for raw audio,” *arXiv preprint arXiv:1609.03499*, 2016.
- [2] Joshua Bassey, Lijun Qian, and Xianfang Li, “A survey of complex-valued neural networks,” *arXiv preprint arXiv:2101.12249*, 2021.
- [3] Jiankun Chen, Xiaolan Qiu, Chibiao Ding, and Yirong Wu, “Cvcmff net: Complex-valued convolutional and multifeature fusion network for building semantic segmentation of insar images,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2021.
- [4] Risi Kondor and Shubhendu Trivedi, “On the generalization of equivariance and convolution in neural networks to the action of compact groups,” in *International Conference on Machine Learning*. PMLR, 2018, pp. 2747–2755.
- [5] Michael Maire, Takuya Narihira, and Stella X Yu, “Affinity cnn: Learning pixel-centric pairwise relations for figure/ground embedding,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 174–182.
- [6] Yuqi Nie and Hui Yuan, “Neural network is heterogeneous: Phase matters more,” *arXiv preprint arXiv:2111.02014*, 2021.
- [7] Tohru Nitta, “The computational power of complex-valued neuron,” in *Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003*, pp. 993–1000. Springer, 2003.
- [8] George M Georgiou and Cris Koutsougeras, “Complex domain backpropagation,” *IEEE transactions on Circuits and systems II: analog and digital signal processing*, vol. 39, no. 5, pp. 330–334, 1992.
- [9] Brian Cheung, Alexander Terekhov, Yubei Chen, Pulkit Agrawal, and Bruno Olshausen, “Superposition of many models into one,” *Advances in neural information processing systems*, vol. 32, 2019.
- [10] Neelam Sinha, Manojkumar Saranathan, Kalpathi R Ramakrishnan, and S Suresh, “Parallel magnetic resonance imaging using neural networks,” in *2007 IEEE International Conference on Image Processing*. IEEE, 2007, vol. 3, pp. III–149.
- [11] Moritz Wolter and Angela Yao, “Complex gated recurrent neural networks,” *Advances in neural information processing systems*, vol. 31, 2018.
- [12] Chiheb Trabelsi, Olexa Bilaniuk, Dmitriy Serdyuk, Sandeep Subramanian, João Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J. Pal, “Deep complex networks,” *CoRR*, vol. abs/1705.09792, 2017.
- [13] Rudrasis Chakraborty, Yifei Xing, and X Yu Stella, “Surreal: complex-valued learning as principled transformations on a scaling and rotation manifold,” *IEEE Transactions on Neural Networks and Learning Systems*, 2020.
- [14] Utkarsh Singhal, Yifei Xing, and Stella X Yu, “Co-domain symmetry for complex-valued deep learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 681–690.
- [15] Tohru Nitta, “Orthogonality of decision boundaries in complex-valued neural networks,” *Neural computation*, vol. 16, no. 1, pp. 73–97, 2004.
- [16] Ken Kreutz-Delgado, “The complex gradient operator and the cr-calculus,” *arXiv preprint arXiv:0906.4835*, 2009.
- [17] Patrick M Virtue, *Complex-valued deep learning with applications to magnetic resonance image synthesis*, University of California, Berkeley, 2019.
- [18] Sundaram Suresh, Narasimhan Sundararajan, and Ramasamy Savitha, *Fully Complex-valued Multi Layer Perceptron Networks*, pp. 31–47, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [19] Akira Hirose, *Complex-valued neural networks*, vol. 400, Springer Science & Business Media, 2012.
- [20] Sundaram Suresh, Narasimhan Sundararajan, and Ramasamy Savitha, *Circular Complex-valued Extreme Learning Machine Classifier*, pp. 109–123, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [21] Sundaram Suresh, Narasimhan Sundararajan, and Ramasamy Savitha, *Supervised learning with complex-valued neural networks*, Springer, 2013.
- [22] Tohru Nitta, “Solving the xor problem and the detection of symmetry using a single complex-valued neuron,” *Neural Networks*, vol. 16, no. 8, pp. 1101–1105, 2003.
- [23] Alex Krizhevsky, Geoffrey Hinton, et al., “Learning multiple layers of features from tiny images,” 2009.
- [24] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bisacco, Bo Wu, and Andrew Y Ng, “Reading digits in natural images with unsupervised feature learning,” 2011.