Complex Valued U-Net for Segmentation of SAR Images

An internship report submitted
in partial fulfillment for the award of the degree of

Bachelor of Technology

in

Electronics and Communication Engineering

by

Saumya Vilas Roy

Department of Avionics
Indian Institute of Space Science and Technology
Thiruvananthapuram, India

October 2023

Certificate

This is to certify that the internship report titled ***Complex Valued U-Net for Segmentation of SAR Images*** submitted by **Saumya Vilas Roy**, to the Indian Institute of Space Science and Technology, Thiruvananthapuram, in partial fulfillment for the award of the degree of **Bachelor of Technology** in **Electronics and Communication Engineering** is a bona fide record of the original work carried out by him/her under my supervision. The contents of this internship report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Mrs. Haripriya S
Designation

Name of Department Head
Designation

Place: Thiruvananthapuram
Date: October 2023

Declaration

I declare that this internship report titled ***Complex Valued U-Net for Segmentation of SAR Images*** submitted in partial fulfillment for the award of the degree of **Bachelor of Technology** in **Electronics and Communication Engineering** is a record of the original work carried out by me under the supervision of **Mrs. Haripriya S**, and has not formed the basis for the award of any degree, diploma, associateship, fellowship, or other titles in this or any other Institution or University of higher learning. In keeping with the ethical practice in reporting scientific information, due acknowledgments have been made wherever the findings of others have been cited.

Place: Thiruvananthapuram
Date: October 2023

Saumya Vilas Roy
(SC20B121)

Acknowledgements

I would like to express my sincere gratitude to Mrs. Haripiya S., Scientist/Engineer 'SE', SARDPD/MDPG/DPA during this internship has been an enriching experience. I am truly grateful for the knowledge and insights I gained from her, and this opportunity to work and enrich myself.

I would also like to extend my thanks to Dr. Deepak Mishra, Professor, Department of Avionics, IIST for consistently motivating me to set higher goals and for providing valuable feedback, guiding me in the right direction.

Finally, I am indebted to all my family, friends and seniors who supported and assisted me throughout this internship. Their collective encouragement and guidance have been instrumental in shaping my learning experience.

Saumya Vilas Roy

Abstract

The increasing data analysis demands of the satellite images has caused the increase in use of Machine learning (ML) in pattern recognition, and clustering using conventional ML techniques but in recent years a sharp increase in using Deep Learning (DL) techniques for more wider application and use of it's niche ability to be fine tuned to particular tasks have sparked the interest, But most of these techniques are mostly based on real valued data but not applicable for Complex data which contains both magnitude and a direction as Polarimetric Synthetic Aperture Radar (PolSAR) images are all complex in nature further processing is required for it to be adapted into the real domain, The aim of this project is to propose a U-net Complex Valued Neural Network (CVNN) for segmentation of the raw PolSAR images using the Pauli representation, it is trained on Indian Space Research Organization (ISRO) EOS-04 satellite data.

Contents

List of Figures

List of Tables

Abbreviations

GNU	GNU's Not Unix
CVNN	Complex Valued Neural Networks
RVNN	Real Valued Neural Network
CV-FCNN	Complex-Valued Fully Convolutional Neural Network
PolSAR	Polarimetric Synthetic Aperture Radar
ML	Machine Learning
DL	Deep Learning
TF	Tensorflow
MLP	Multi Layer Perceptron
FCCCNN	Fully Connected Convolution Complex Neural Network

Chapter 1

Introduction

The advent of different PolSAR technologies allowing us to capture higher resolution of images the analysis of which can led to valuable insights the conventional methods may fail in such a huge dataset pool to counter that DL techniques have come up in the part decades and have made significant progress in analysis of such data, One such problem tackled is semantic segmentation of the images.

1.1 What is semantic segmentation ?

The process of assigning a class label to each pixel of the image to indicate what is represented by the particular pixel, This task is also called the dense prediction, since we are predicting for each pixel in the image. The result of this method is essentially a high-quality image with each pixel assigned to a certain class, often the same size as the original image. So, it's a pixel-level image classification. It is used extensively in Imagery, health care, automation, tracking etc.

1.2 How to achieve semantic segmentation ?

There are multiple methods to achieve semantic segmentation like U-Net [?], SegNet [?] ,Pyramid Scene Parsing Network (PSPNet) [?], DeepLab [?], ParseNet [?] all these are deep learning techniques to achieve the goal all these have their own specific uses advantages and disadvantages but for this project Unet was selected due to it's high versatility and vast literature is available.

1.3 Introduction to Unet

U-Net is a fully convolution network i.e. it doesn't posses any dense layers connections it was developed for biomedical image segmentation the main idea behind it is to supplement a network with a downsampeling part where blocks are pooled together to reach a smaller size and then acted by a upsampeling part where those polling layes are replaced by unpooling operation to map the pixels to their respective labels, modern U-Net also allow skip connection to propagate the information that might be dropped during the downsampling layers.

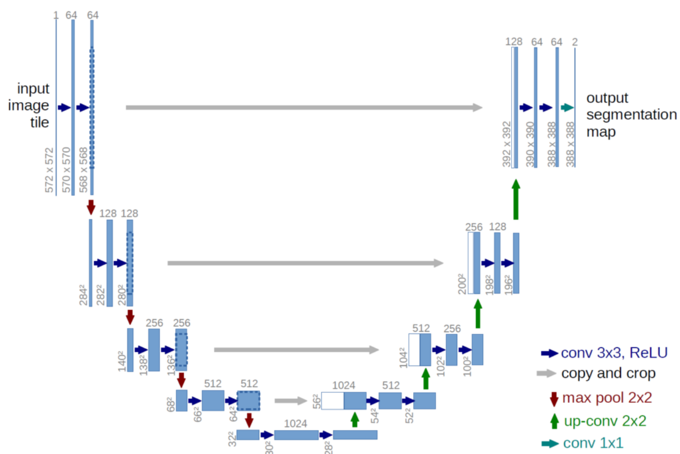


Figure 1.1: U-Net architecture

1.4 Why Complex -Valued neural network?

Many real word process cannot be modeled efficiently by real values belonging to R such as SAR images, Signals, biological process and so on require two parameters one for the magnitude and other for the direction i.e. they belong in the complex domain C, processing of such data is mostly done by casting in into real domain and then using RVNN on it but that causes some information to be lost and might inhibit us from finding meaningful relations so using the CVNN would help us to preserve such information and find a much concrete solution.

Chapter 2

Literature review

In A Survey of Complex-Valued Neural Networks [?] paper the authors go in dept about the history of CVNN and how it's inception was carried out only after gradient decent was derived for the Complex plane and the future use of Wirtinger's calculas to generalize it for the differentiation to be performed with the complex variable instead of the real counter part , further discussion on mathematics will be done in next chapter, the paper mentions how activation functions and loss functions were adapted for the complex domain, and several different approaches for optimization, training with gradient and non-gradient based methods they further discuss the scope of CVNN over different domains and applications image processing, computer vision and it's usage in wireless communication as complex values occur there naturally or by design, finally they end by mentioning the shortcomings and improvement scopes for the CVNN like lack of support by libraries like TF, pytorch etc. Or using CVNN increases the model complexity due to addition of multiple params and cause the training to be very computationally expensive.

The Theory and Implementation of Complex-Valued Neural Networks [?] paper provides extensive details into programming and implementations of CVNN here the authors introduce their open source library cvnn build for python over TF to implement the CVNN as standing it's the only library available with all these features, further more they provide code snippets to show their API, which by design is similar to TF for the user convenience after which they go in depth about the mathematics behind which I will discuss in the next chapter, further they try and classify real valued signals using RVNN and then after using Hilberts transform to convert into complex domain and use the CVNN to classify the same signal and produce higher classification accuracy with the CVNN.

In Complex-Valued Neural Networks for Polarimetric Sar Segmentation Using Pauli

Representation [?] the authors show that the use of Pauli vector representation is when used instead of coherency matrix produce better segmentation results as Pauli representations contain richer information and produced much higher accuracy i.e. better segmentation.

The Classification of Polarimetric SAR Data by Complex Valued Neural Networks [?] in this paper they use E-SAR sensor in L band and classify and segment the image into 3 classes they show that MLPs cannot work with original signal which has different features here they show CVNN with different error functions and show that Complex Cauchy Function and Complex Quadratic Function were able to achieve low error percentage.

2.1 Summary

CVNN is slowly gaining momentum in development because the analysis in complex domain allows us to preserve relationship in phase and magnitude which is lost to us if we map the data into a real domain, the further development of CVNN will allow us to have much more needed methods in fine tuning and developing better models, currently PolSAR analysis using CVNN is seeing a lot of popularity and attention in recent years as it's ability to adapt and use the specific mathematical insights. And it provides numerous advantages over the conventional RVNN in specific tasks which may allow us to automate much of the PolSAR image processing and analysis.

Chapter 3

Theory

3.1 Theorems in complex domain calculus

Theorem 3.1. *Liouville Theorem:*
In complex analysis, an entire function, also called an integral function, is a complex-valued function that is holomorphic at all finite points over the whole complex plane
Given a function f, the function is bounded if $M \in R^{+} : |f(x)| < M$.

This theorem shows if the objective function in CVNN is holomorophic then it would be finite in space.

Theorem 3.2. *Cauchy-Reimann equations:*
The Cauchy–Riemann equations on a pair of real-valued functions of two real variables u(x, y) and v(x, y) are the two equations:
$\frac{\partial u}{\partial v} = \frac{\partial v}{\partial y}$
$\frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}$

These equations help check holomorphicity of the complex functions

Theorem 3.3. *Wirtinger Calculus:*
Given a complex function $f(z)$ *of a complex variable* $z = x + iy \in C, x, y \in R$*.*
The partial derivatives with respect to z and z respectively are defined as:
$\frac{\partial f}{\partial z} \triangleq \frac{1}{2}(\frac{\partial f}{\partial x} - i\frac{\partial f}{\partial y})$
$\frac{\partial f}{\partial \bar{z}} \triangleq \frac{1}{2}(\frac{\partial f}{\partial x} + i\frac{\partial f}{\partial y})$

These derivatives are called R-derivative and conjugate R-derivative, respectively. As said before, the holomorphic case is only a special case where the function can be consid-

ered as $f(z, z), z = 0$. *Wirtinger calculus enables to work with non-holomorphic functions, providing an alternative method for computing the gradient that also improves the stability of during training.*

3.2 DL techniques in CVNN

As complex values exist in a domain all together, so conventional real valued techniques won't be applicable and useful so facilitate for complex domain they are forced undergo necessary changes:

3.2.1 Complex-Valued Backpropagation

In complex calculus standard complex derivatives exist only for holomorphic function but as Liouville's theorem states it is finite over the space it would not let the back propagation reach an optimal solution, to over come this Wirtinger's calculus is used to approximate the partial derivatives to add non holomophicity in the network

- By loss function minimization over complex variables.
- By non holomorphic activation functions.

Wit Only the partial differential changes in CVNN when compared to RVNN if we use the Wirtinger calculus definition we can make the holomorphic function a special case of itself and it enables us to do back propagation in CVNN.

3.2.2 Complex pooling

As the values in Complex domain are not ordered similar to real values the maximum value cannot be determined to allow such comparison norm of the magnitude can be used of the values can be used or the circular mean can be calculated based upon the angles and normalized values.

3.2.3 Complex upsampeling

In CVNN it's done using bilinear interpolation or Nearest Neighbor to upscale the layers.

3.2.4 Complex activation functions

The activation function in should be piece wise smooth to allow computation of the gradients, to create a complex domain activation function the real domain activation function is mostly extended into complex domain in either of these ways.

- Type-A: $\sigma_A(z) = \sigma_{Re}(Re(z) + i\sigma_{Im}(Im(z)),$
- Type-A: $\sigma_B(z) = \sigma_r(|z|)\exp(i\sigma_\phi(arg(z)),$

Here $\sigma_A, \sigma_B, \sigma_r, \sigma_\phi \in R$ and most popular activation functions like sigmoid, hyperbolic tangent (tanh) and Rectified Linear Unit (ReLU) and softmax is extendable this way.

3.2.5 Complex compatible loss function

The output of a loss function compatible with Complex value is will still have a real output for analysis we can create and expand domains of existing loss functions we can make a new average cross-entropy loss which is just the average of categorical cross-entropy loss on both real and imaginary values.

$L^{ACE} = \frac{1}{2}[L^{CCE}(Re(y), d) + L^{CCE}(Im(y), d)]$

Here L^{ACE} is just the complex average cross-entropy, L^{CCE} is just categorical cross-entropy, y is the prediction from the model whereas h is the ground truth.

With these mathematical tools and theorems we can now expand the Real valued DL into the Complex Domain and implement a CVNN.

Chapter 4

Experimental Setup

4.1 Dataset

The data which is used here was acquired from EOS-04 satellite launched by ISRO in 2022, it has a C-band synthetic-aperture radar (SAR) the images are of Ahemdabad region and are just RAW PolSAR data without any preprocessing and for labeling the data to generate the mask Pauli matrix values are used the assign labels the RGB value present in the matrix are averaged out and thresholded as follows for 4 class segmentation:

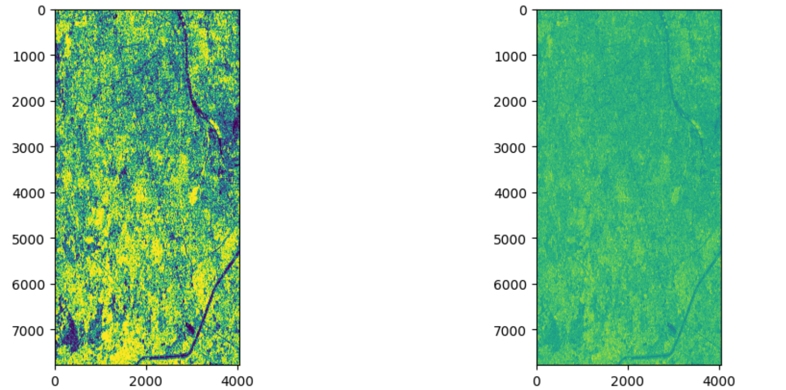
$$d = \begin{cases} 1 & \text{if } R \text{ is >greater than .5} \\ 2 & \text{if } G \text{ is >greater than .5} \\ 3 & \text{if } B \text{ is >greater than .5} \\ 0 & else \end{cases}$$

After thresholding this way 54.85% were assigned to label 3, which caused very big class imbalance and led to very poor segmentation results, to counter it the problem was converted from a 4 class problem to 2 class problem.

$$d = \begin{cases} 1 & \text{if } B \text{ is >greater than .5} \\ 0 & else \end{cases}$$

Here d denotes the real label which were are assigning.

Training was done with these masks given as the labels, there were further preprossesing done by using Median filer on these masks to smooth them out.



a. 2 Class Thresholding b. 4 Class Thresholding.

Figure 4.1: Thresholded Mask Images

4.2 Process flow

For this project a U-net architecture was implemented in Python using cvnn library [**?**] with TF the architecture of U-net is as follows

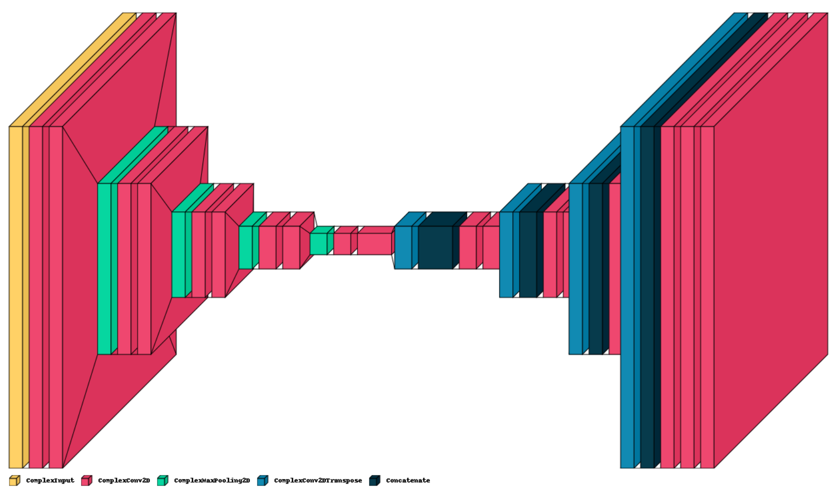


Figure 4.2: CVNN U-Net architecture

The U-net architecture here follows the conventional structure with encoder and decoder also it had a skip connection from every last conv2D layer in each step of the encoder to the relevant concatenation layer in the decoder to allow passage of information that might have been lost due to the kernels.

Here are the output filters in each layer or the encoder $[32, 64, 128, 256]$ decoder has the same output filters in just reverse order $[256, 128, 64, 32]$ As defined in the Nomenclature this U-net will be refereed to as ϕ CVNN

To find overfitting capabilities of CVNN an another encoder $[64, 128, 256, 512]$ and decoder $[512, 256, 128, 64]$ were used as ρ CVNN.

The original dataset shaped 7779 x 4038 was split into 1860 small patches of shape 128 x 128, which were further split into 70% for training, 15% for validation and rest 15% for testing set and it was done by using Patchify library python [**?**].

In Analysis on the Dropout Effect in Constitutional Neural Networks [**?**] the authors talk about how a very small dropout in Convolution layers assist in countering overfitting to incorporate that an another model with different rates of dropouts were implemented results will be discussed in the next chapter.

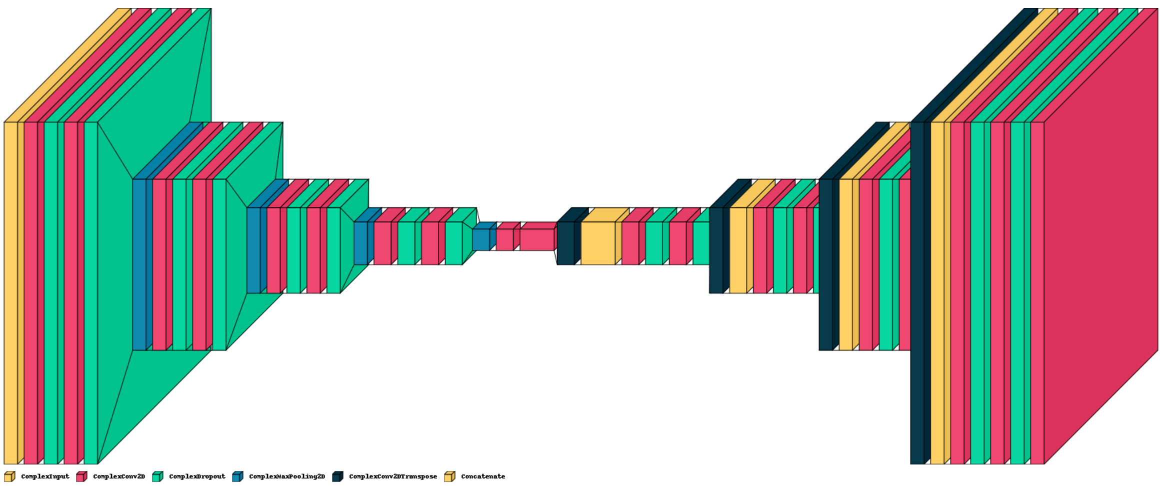


Figure 4.3: CVNN U-Net architecture with dropout layers

Here only a dropout layer is introduced after the convolution operations rest is identical to the above U-net architecture.

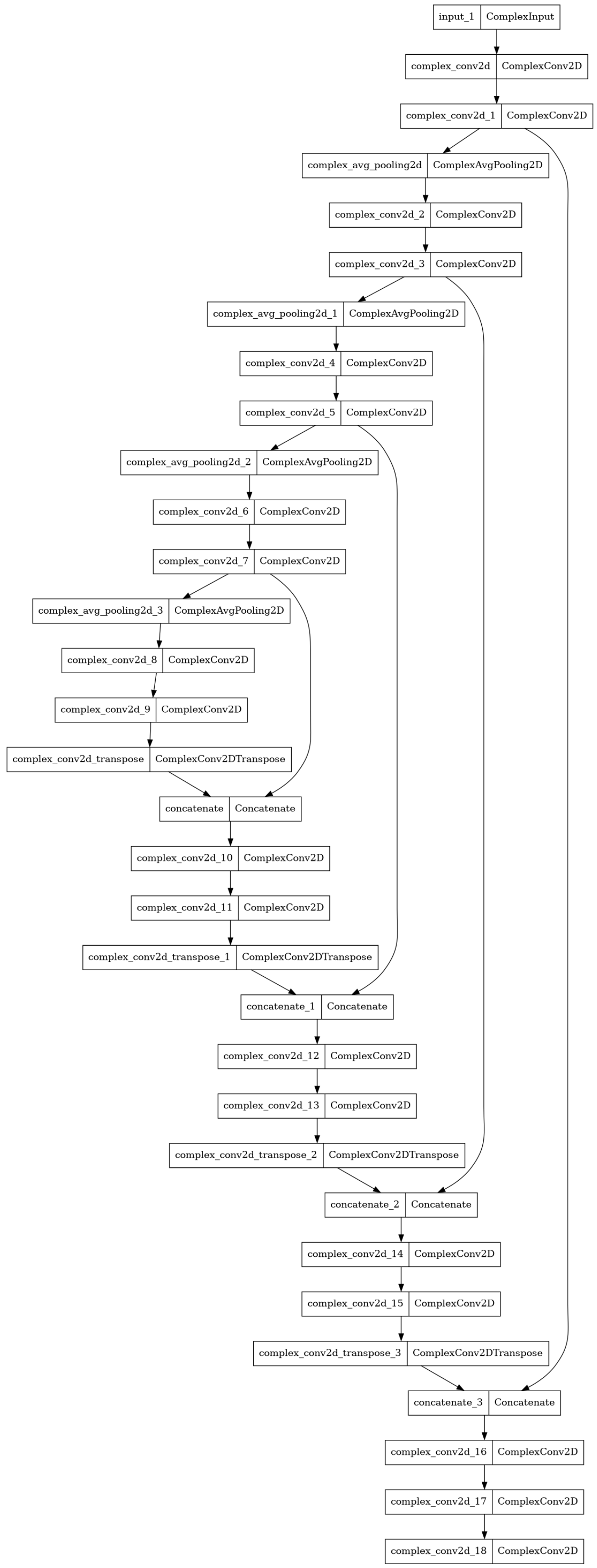


Figure 4.4: CVNN U-Net architecture flowchart

Chapter 5
Analysis

5.1 Methodology

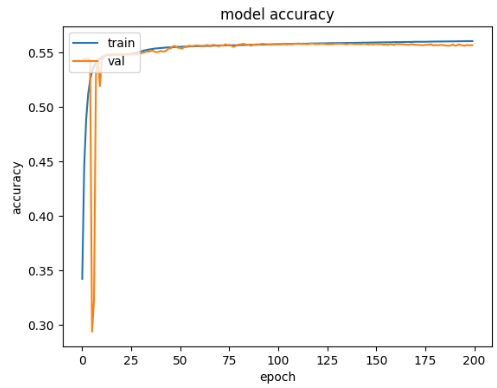
To check the effectiveness of CVNN in segmentation several different methods and optimization were used to find the optimal model parameters and theresholding technique.

The following methods were tried:

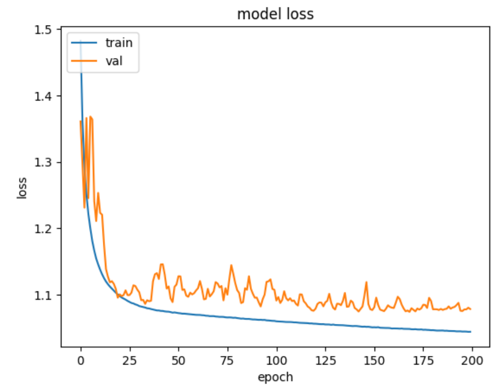
1. ϕ CVNN for segmentation on the 4 class mask.
2. ϕ CVNN for segmentation on the 2 class mask.
3. ϕ CVNN for segmentation with drop out rate 0.1 on the 2 class mask.
4. ϕ CVNN for segmentation with drop out rate 0.2 on the 2 class mask.
5. ϕ CVNN for segmentation with drop out rate 0.5 on the 2 class mask.
6. ρ CVNN for segmentation on the 2 class mask.
7. ρ CVNN for segmentation with drop out rate 0.1 on the 2 class mask.

Results of the training with following parameters and masks are plotted in the subsequent section.

5.2 Plots and Trends

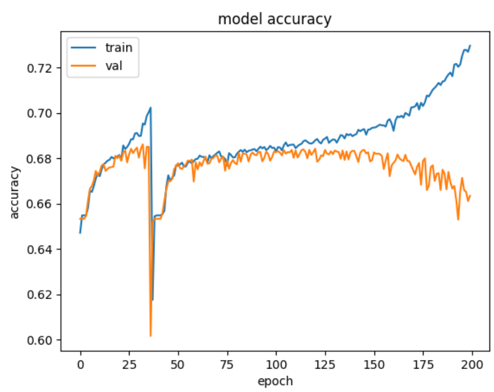


a. Model validation and training accuracy

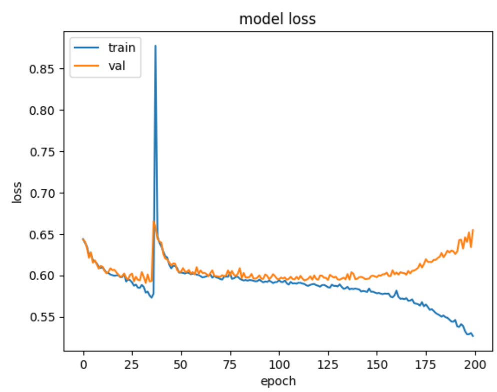


b. Model Loss and training accuracy

Figure 5.1: ϕ CVNN with no dropouts on 4 class segmentation

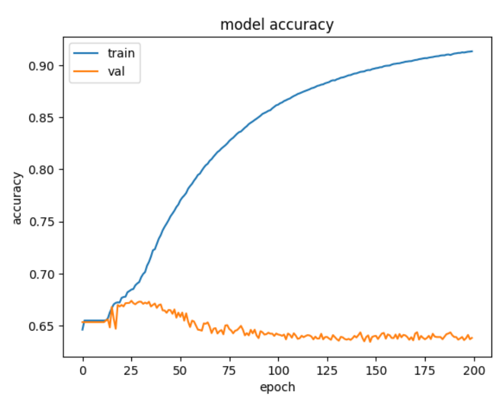


a. Model validation and training accuracy



b. Model Loss and training accuracy

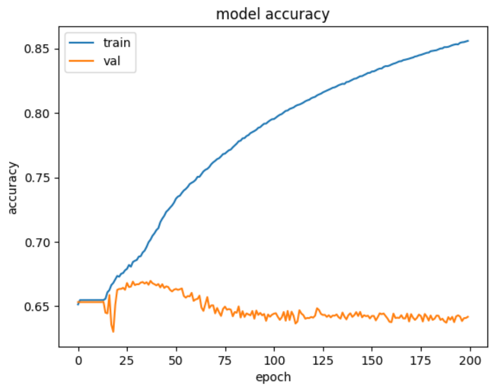
Figure 5.2: ϕ CVNN with no dropouts on 2 class segmentation



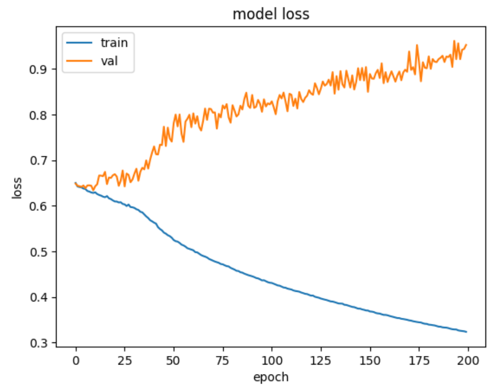
a. Model validation and training accuracy

b. Model Loss and training accuracy

Figure 5.3: ϕ CVNN with dropout rate 0.1 on 2 class segmentation

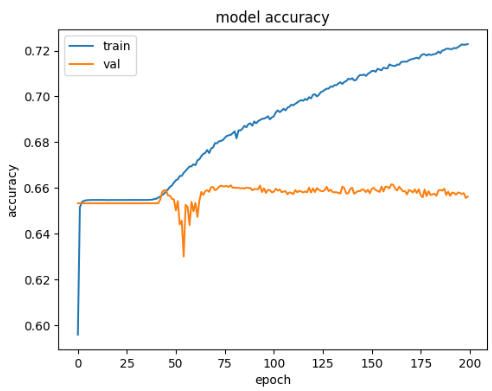


a. Model validation and training accuracy



b. Model Loss and training accuracy

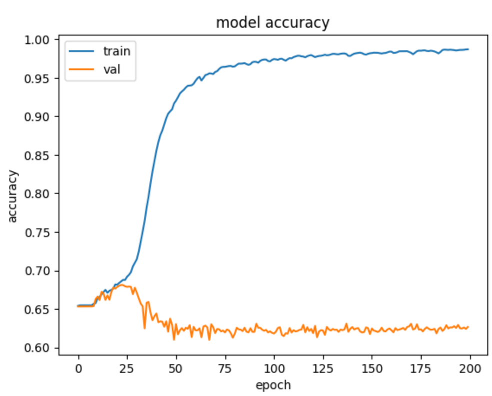
Figure 5.4: ϕ CVNN with dropout rate 0.2 on 2 class segmentation



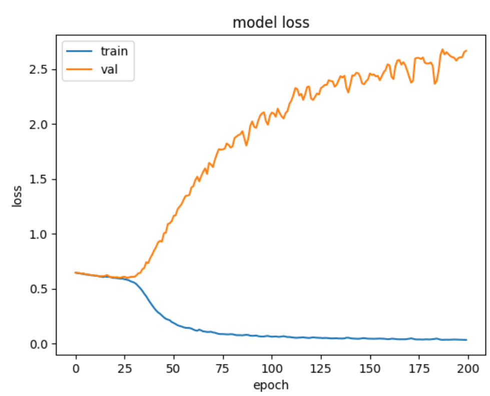
a. Model validation and training accuracy

b. Model Loss and training accuracy

Figure 5.5: ϕ CVNN with dropout rate 0.5 on 2 class segmentation

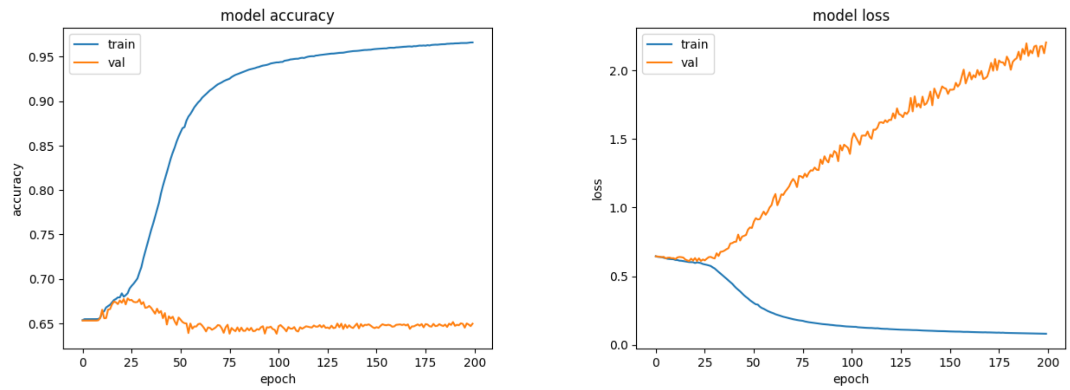


a. Model validation and training accuracy



b. Model Loss and training accuracy

Figure 5.6: ρ CVNN with no dropouts on 2 class segmentation



a. Model validation and training accuracy b. Model Loss and training accuracy

Figure 5.7: ρ CVNN with dropout rate 0.1 on 2 class segmentation

5.3 Observations

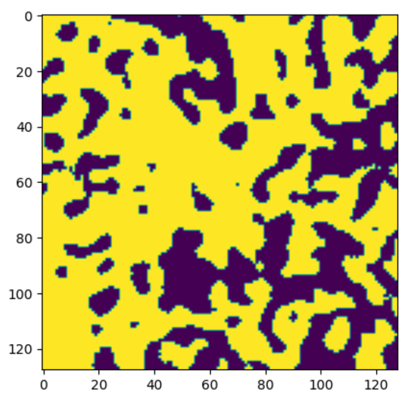
As we can observe in the Figure 5.1 due to high class variation due to unbalanced labeling the model is unable to reach a optimum solution as one label is very dominant,In Figure 5.2 with ϕ CVNN model goes to the most optimal solution after which it starts to overfit after 60 epochs, the same model ϕ CVNN was trained with varying dropout rates in Figures 5.3, 5.4 and 5.5 show that dropouts are mostly ineffective in FCCCNN, For Figure 5.6 ρ CVNN architecture was considered and were to test overfitting capabilities of highy complex CVNN models and as we can see from trends from the validation loss and accuracy against the training loss and accuracy we can see the model is not learning anything after 25 epochs, even in Figure 5.7 adding dropout didn't help much and model kept overfitting.

5.4 Results

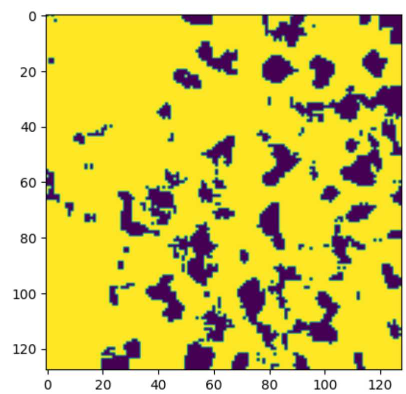
As we can see from the testing results model is overfitting if the output parameters in U-net are large to find an optimum we need to employ more regularization techniques as CVNN contain much more parameter as compared with RVNN and as we can see from he testing results and they overfit the data easily, but the model ϕCVNN produces the highest accuracy with no dropouts.

Table 5.1: The testing results of the models

S. no.	Model	Dropout Val.	Output Class	Testing Loss	Testing Accuracy	Testing Precision
1	ϕCVNN	0	4	1.1034	0.5507	0.6499
2	ϕCVNN	0	2	0.6458	0.6710	0.6710
3	ϕCVNN	0.1	2	1.1405	0.6468	0.6468
4	ϕCVNN	0.2	2	0.9265	0.6519	0.6519
5	ϕCVNN	0.5	2	0.8619	0.6605	0.6605
6	ρCVNN	0	2	2.5442	0.6390	0.6390
7	ρCVNN	0.1	2	2.1523	0.6590	0.6590



a. Target 2 class mask patch.



b. Predicted 2 class mask patch.

Figure 5.8: Target patch and Predicted patch from ϕ CVNN Model with no dropouts.

5.5 Conclusions

As we can concur from the trends and results, The performance of CVNN for segmentation is highly depended on the labeling techniques used and lack of regularizations in complex domain is challenging problem in itself for smooth training and to reach the optimal solution, but for data such as PolSAR or signals in complex domain it is very important avenue for us to explore as it allows us to keep the correlated information of phase and magnitude which is mostly lost if it's adapted into Real domain R more research on optimization techniques for CVNN may allow us to create much optimal solutions for Complex data.

5.6 Future scope

The following work can be improved upon future by using Semi-supervised DL or ML techniques to label the data as the current thresholding methodology of using the Pauli RBG matrx is heavily biased towards one class and the labeling is not smooth and continuous, can be improved upon further, lack for regularization techniques in complex domain as $Ridge(L2)$ and $LASSO(L1)$ regularizations won't transform well in complex domain as lack of bounds for complex values even if use the magnitude of the complex values the phase information is lost so using either of them are not viable, also CVNN are very sensitive to hyperparameter tuning and further multiple models with varying learning rate or using variable learning rate might help us to optimize the results further, addition of more data for training might also help the model to improve further.