Complex-valued Neural Networks in Computer Vision

THESIS

submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Computer Science and Engineering

by

Saurabh Yadav

PhD20023

Under the supervision of

Dr. Koteswar Rao Jerripothula



Indraprastha Institute of Information Technology, Delhi

New Delhi, 110020

October 2026

Declaration

◇ ◇ ◇

I declare that I am the author of the thesis entitled **Complex-valued Neural Networks in Computer Vision**. This thesis presents research carried out by me under the supervision of **Dr. Koteswar Rao Jerripothula** at the Indraprastha Institute of Information Technology, Delhi. To the best of my knowledge, this work is original and has not been submitted, in whole or in part, to any other institution or university for the award of a degree.



Saurabh Yadav
Doctoral Candidate,
Dept. of Computer Science and Engineering,
IIIT Delhi, 110020.

Place: New Delhi

Date: 15 October, 2026

Declaration Regarding Use of AI

◇ ◇ ◇

I acknowledge that I am fully responsible for the entire content of my thesis, including any sections assisted by online tools, including Artificial Intelligence-based tools. I accept full accountability for any violations of ethical standards in publications arising from the use of such tools.



Saurabh Yadav
Doctoral Candidate,
Dept. of Computer Science and Engineering,
IIIT Delhi, 110020.

Place: New Delhi

Date: 15 October, 2026

Thesis Certificate

◇ ◇ ◇

This is to certify that the thesis titled **Complex-valued Neural Networks in Computer Vision** being submitted by **Saurabh Yadav** to the Indraprastha Institute of Information Technology-Delhi, for the award of the degree of **Doctor of Philosophy**, is an original research work carried out by him under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.
The results contained in this thesis have not been submitted in part or in full to any other university or institute for the award of any degree/diploma.



Dr. Koteswar Rao Jerripothula
Adjunct Faculty,
Department of Computer Science and Engineering,
Indraprastha Institute of Information Technology Delhi,
Okhla Industrial Estate, Phase III,
New Delhi, Delhi 110020.

Date: 15 October, 2026

Acknowledgement

I would like to express my profound gratitude to my supervisor, Dr. Koteswar Rao Jerripothula, for his invaluable mentorship and constant support throughout my Ph.D. journey. His guidance was instrumental not only in the technical details of my research but also in my personal and professional development. I am deeply appreciative of his generosity with his time, his willingness to engage in deep technical discussions, and his unwavering support during the challenging conference review processes. He patiently helped me deconstruct and learn from difficult reviewer feedback, and I thank him for his continuous encouragement and for helping me learn and grow.

I extend my sincere thanks to my Ph.D. monitoring committee members, Dr. Saket Anand and Dr. A. V. Subramanyam, for their encouragement and insightful feedback throughout my doctoral studies. Their constructive criticism was essential in shaping the direction of my work. I am also grateful to Dr. Subrahmanyam Murala for his valuable feedback during my Ph.D. comprehensive exam. I also thank Prof. Prabhat Munshi for his assessment and feedback. I am also grateful to the anonymous reviewers of my papers for their constructive feedback, which helped improve the quality of my research.

This research was made possible by the financial support of the University Grants Commission (UGC), Government of India. I am sincerely grateful for their funding, which was essential for the completion of this work. I would like to thank CAI and CSE, IIITD, for providing an incredible research environment and facilities, which helped me conduct my research work.

On a personal note, I would like to express my deepest gratitude to my parents, Mrs. Lalita Yadav and Mr. Rajendra S. Yadav. It was their love, moral support, and emotional encouragement that nurtured me into the person I am today. My special thanks go to my beloved younger sister, Shivani, for being a constant source of joy and for taking wonderful care of our parents in my absence.

A Ph.D. journey is not undertaken in isolation. It is a path filled with challenges, and mine would have been far more difficult without the camaraderie of my colleagues. I am thankful for the spirited discussions and the intellectual debates; this journey would not have been the same without them.

I am also incredibly grateful to my friends at IIITD: Gunjan, Avi, Shyama, Mayank, Himanshu, and many others. They helped me navigate the inevitable struggles of research and fostered a collaborative environment where we could grow together as researchers. During this journey, I also met Purva who holds a special place, I remain thankful to God for her presence. Finally, I would like to extend a special thank you to

Divesh, Vipin and Nitin for their constant support and for bringing joy to my daily life throughout this process.

Looking back on these years, I have learned to be patient with uncertainty, to accept criticism, and to keep working when progress is slow. I am grateful to everyone who shared their time, ideas, and encouragement along the way. As I complete this thesis, I carry these lessons with me, along with a deep appreciation for the people who made this journey possible.

Abstract

Neural network-based approaches have been evolving largely due to their remarkable capacity for learning powerful representations from data. Despite this success, the exploration of complex-valued neural networks (iCNNs) in mainstream computer vision has been limited. iCNNs, by their inherent complex-valued nature, possess a richer representational capacity compared to their real-valued counterparts, enabling them to simultaneously encode magnitude and phase information, which can be crucial for various data types. This rich capacity, coupled with their potential for faster learning and improved generalization characteristics, makes them theoretically appealing. However, the primary hurdles to their widespread adoption in computer vision lie in two key areas: first, the lack of naturally occurring complex-valued visual data; and second, the complexities involved in processing and optimizing information within the complex domain. Consequently, their application has largely been confined to niche domains where complex-valued data is inherently available, such as audio speech enhancement, radar signal processing, and SAR data analysis.

While recent studies have begun to explore iCNNs in computer vision, they often report only comparable improvements, suggesting that the full potential of complex-valued networks in this field remains largely untapped. This study addresses these fundamental challenges, proposing novel methodologies to unlock the power of complex-valued representations and processing for a broader spectrum of computer vision tasks. Our overarching theme is to demonstrate that by carefully constructing complex-valued inputs, designing appropriate complex-valued architectures, and formulating differentiable complex loss functions, we can significantly advance applications in vision problems.

The first foundational challenge in applying iCNNs to standard computer vision tasks is the transformation of real-valued image data into a meaningful complex-valued representation. Without this, the inherent benefits of complex-valued processing cannot be fully realized. To this end, our first study introduces a novel complex-valued color model: iHSV. iHSV is designed to convert conventional RGB images into complex-valued formats, preserving and enriching color information in the phase and magnitude components. The iHSV model leverages the cylindrical nature of the HSV color space, mapping real values to a complex representation. Effectively handling the complex-valued information flow throughout the network and optimizing its complex-valued outputs are critical. Standard real-valued operations and loss functions are not directly applicable or optimal in the complex domain. To overcome this, we propose maintaining an end-to-end complex-valued information flow by utilizing complex operations for convolutions, activations, and pooling. We introduce a novel differentiable complex loss function specifically tailored to optimize complex-valued outputs for performing image

classification in the complex domain. Building upon these innovations, we propose FCCNs: Fully Complex-valued Convolutional Networks.

While our first study established the benefits of end-to-end complex-valued processing in the spatial domain, in our second study, we delve deeper into the interplay between spatial and frequency domain information within a complex-valued framework. We propose a novel complex-valued color R2C (Real-to-Complex) transformation, designed to encapsulate color and intensity information separately in complex representation for exploring binary segmentation problems. Central to this study is the introduction of a novel token-based complex-valued architecture, DCS^{2}Net (Deep Complex Spatio-spectral Network), which explores the concurrent processing of complex-valued information in both spatial and frequency domains. We rigorously evaluate DCS^{2}Net's performance on the challenging task of salient object detection, shadow detection, and defocus blur detection. This study highlights the power of a unified complex-valued framework for extracting and processing spatial-spectral information, leading to more robust and accurate predictions for dense prediction tasks.

The recent emergence of large-scale foundation models, such as the Segment Anything Model (SAM), has revolutionized many computer vision tasks by providing powerful, generalizable representations. However, these models, primarily trained on vast real-valued datasets, can sometimes struggle with highly specialized or challenging domains, such as camouflage object detection. Instead of retraining the entire foundation model, which is computationally expensive, we propose to inject complex-valued high-frequency features using the wavelet transform into the encoder of models like SAM using our proposed approach, SAMwave. The rationale is that high-frequency components often encode crucial textural details, edges, and fine structures critical for discriminating objects in challenging scenarios (e.g., camouflaged objects blending with their background). By converting these high-frequency components of images into a complex-valued format and feeding them to specifically designed complex-valued adapters of SAM's architecture, we aim to augment the model's existing real-valued representations with richer information.

This research lays a robust foundation for the broader application of complex-valued neural networks in computer vision. It can be extended in several promising directions in the future. Firstly, we can explore the utility of complex-valued representations for adapting Vision-Language Models (VLMs) to problems such as audio-guided object segmentation and audio-visual scene understanding. The inherent ability of complex numbers to represent relationships could be highly beneficial in such tasks. Secondly, we can investigate the use of complex-valued representations for learning with limited ground-truth labels in semi-supervised and unsupervised paradigms for iCNNs. The richer representational capacity of iCNNs might enable them to extract better patterns and features from unlabeled data, thereby reducing the reliance on extensive annotations. Thirdly, we can explore more diverse and sophisticated complex-valued representations for visual data, going beyond color models and transformations to capture multi-scale and multi-spectral information. Finally, we can leverage these color mod-

els and transformations for image generation and explore generative AI in the complex domain.

Publications

Published/Accepted

- Saurabh Yadav and Koteswar Rao Jerripothula, "*FCCNs: Fully Complex-valued Convolutional Networks using Complex-valued Color Model and Loss Function*," IEEE/CVF International Conference on Computer Vision (**ICCV**), Paris, France, 2023. [**CORE A***]
- Saurabh Yadav, Avi Gupta and Koteswar Rao Jerripothula, "*SAMWave: Adapting Segment Anything Model to difficult tasks*," British Machine Vision Conference (**BMVC**), Sheffield, UK, 2025. (**Oral**) [**CORE A**]

Under Submission

- Saurabh Yadav and Koteswar Rao Jerripothula, "*DCS^{2}Net: Deep Complex Spatio-Spectral Network*". [Submitted to IEEE TIP]
- Saurabh Yadav, Avi Gupta and Koteswar Rao Jerripothula, "*SAMwave: Wavelet-Driven Feature Enrichment for Effective Adaptation of Segment Anything Model*," [Submitted to IJCV]

Others (unrelated to Thesis)

- Avi Gupta, Saurabh Yadav, Koteswar Rao Jerripothula, and Tammam Tillo, "*SELECT: SELEctive Context Transfer for Class Incremental Semantic Segmentation*." British Machine Vision Conference (**BMVC**), Lancaster, UK, 2026. [**CORE A**]

Table of Contents

Declaration **i**

Declaration Regarding Use of AI **ii**

Thesis Certificate **iii**

Page

1 Introduction **1**

1.1 Motivation and Background 1

1.2 Complex-valued Neural Networks: Foundations and Challenges . . . 2

1.3 Related Work 3

1.4 Bridging the Real and Complex Domains 5

1.5 Research Objectives and Contributions 5

1.5.1 FCCNs: Fully Complex-valued Convolutional Networks using iHSV 6

1.5.2 DCS^{2}Net: Deep Complex Spatio-Spectral Networks 6

1.5.3 SAMwave: Adapting Large Vision Models using Complex Wavelets 7

1.6 Future Directions 8

1.7 Thesis Organization 8

1.8 Summary 8

2 Background **9**

2.1 Complex Numbers: Definitions and Properties 10

2.1.1 Fundamental Representations 10

2.1.2 Basic Arithmetic Operations 10

2.1.3 The $\mathbb{R}^2$ Isomorphism 11

2.2 Real-to-Complex Image Representations 11

2.2.1 Baseline Encoding Strategies 12

2.2.2 Transform-Domain Representations 12

2.2.3 Color-Specific and Hypercomplex Representations 13
2.3 Applications of iCNNs in Computer Vision 13
2.3.1 Image Recognition 14
2.3.2 Binary segmentation 16
2.3.3 Object Discovery and the Binding Problem 17
2.4 Advanced Complex Architectures for Visual Representation 18
2.4.1 Complex Transformers and Attentional Learning 18
2.4.2 Few-Shot and Metric Learning in Vision 19
2.5 Complex-Valued Deep Learning in Tangential Domains 19
2.5.1 Link Prediction in Knowledge Graphs 20
2.5.2 Natural Language Processing (NLP) 20
2.5.3 Audio and Speech Processing 20
2.6 Summary and Remaining Gaps 21

3 FCCNs: Fully Complex-valued Convolutional Networks 23
3.1 Related Work 25
3.2 Proposed Method 26
3.2.1 Fixed Hue Based Channel (I_h^*) 27
3.2.2 Fixed Saturation Based Channel (I_s^*) 27
3.2.3 Fixed Value Based Channel (I_v^*) 28
3.2.4 iHSV Color 29
3.2.5 Complex-valued Inputs 29
3.2.6 FCCN Architecture 30
3.2.7 Complex-valued Cross-entropy Loss ($\mathscr{L}_{comp}$) 31
3.3 Experiments 33
3.3.1 Datasets & Model Architectures 33
3.3.2 Results on Benchmark Datasets 34
3.3.3 Results on ImageNet 35
3.3.4 Results on Complex-valued data 35
3.3.5 Ablation Study 36
3.3.6 Model performance Analysis 37
3.3.7 Addition Observations 39

3.3.8 Applicability to Dense Prediction Tasks 40
3.4 Summary 40

4 DCS^{2}Net: Deep Complex Spatio-Spectral Networks 41
4.1 Related Work 44
4.2 Method 45
4.2.1 R2C: Complex-valued color transform 45
4.2.2 Intensity Channel (Z_I) 45
4.2.3 Chromatic Channel (Z_C) 46
4.2.4 Invertibility of R2C Transform 47
4.2.5 Simplified pseudo code 49
4.2.6 DCS^{2}Net: Deep Complex Spatio-Spectral Networks 49
4.2.7 Objective function 54
4.3 Experiments 55
4.3.1 Experimental Setup 57
4.3.2 Implementation Details 57
4.3.3 Experimental Results 58
4.3.4 Ablation Study 60
4.4 Summary 63

5 SAMwave: Wavelet-Driven Feature Enrichment for Effective Adaptation of Segment Anything Model 64
5.1 Related Works 66
5.2 Methodology 67
5.2.1 Motivation for High-Frequency Feature Utilization 67
5.2.2 SAM as the Backbone 68
5.2.3 Wavelet High-Frequency Module (WHF) 69
5.2.4 SAMWave Architecture 70
5.2.5 Objective Function 72
5.3 Experimental Details and Results 72
5.3.1 Datasets and Evaluation Metrics 73
5.3.2 Implementation Details 74
5.3.3 Experimental Results 75

5.3.4 Ablation Study 77
5.4 Summary 78

6 Conclusion & Future Work **79**
6.1 Overview 79
6.2 Summary of Contributions 79
6.2.1 Fully Complex-valued Convolutional Networks (FCCNs) . . . 79
6.2.2 Deep Complex Spatio-Spectral Networks (DCS^{2}Net) 80
6.2.3 SAMWave: Adapting Foundation Models using Complex-valued Wavelets 81
6.2.4 Quantitative Synthesis of Performance Gains 81
6.3 Limitations of the Proposed Methods 82
6.4 Future Work 83
6.5 Concluding Remarks 83

List of Tables

TABLE **Page**

1.1 Summary of representative prior studies in complex-valued learning, detailing their core contributions and primary limitations when applied to computer vision. 4

3.1 RGB →HSV→ iHSV conversion of basic colors 29

3.2 Comparing classification accuracies of different approaches using CIFARnet [138] while varying the input strategy on five benchmark real-valued datasets. The values in **blue** are best, and the ones in **red** are second best. Our FCCN approach with iHSV inputs obtains the best results on every dataset. The second best results are achieved using our FCCN approach with sliding [138] input strategy on 3/5 datasets. . . . 34

3.3 Comparing classification accuracies(%) of different approaches using CDS-large[138], a large network architecture, on CIFAR-10 dataset. Our proposed FCCN outperforms all other approaches. The #Params is 1.7M for all the approaches. 34

3.4 Comparing Top-1 and Top-5 classification accuracies of different approaches using multiple large network architectures on large Imagenet-1k dataset. FCCN performs best again regardless of the network architecture used. 35

3.5 Comparing classification accuracies of different approaches on MSTAR, a complex-valued benchmark dataset. 36

3.6 Results of our ablation study that highlights the value of different parts of our iHSV color representation, which can be divided in two ways: channel-wise $\{I_h^*, I_s^*, I_v^*\}$ and component-wise $\{I_{re}, I_{im}\}$. We conduct this study using a wide-range of standard architectures on CIFAR-10 dataset. 37

3.7 Results of our ablation study that highlights the value of different components of our proposed method: iHSV, f_{conv} and $\mathscr{L}_{comp}$. We conduct this study using a wide-range of standard architectures on CIFAR-10 dataset. 37

3.8 All our improvements are over DCN [151], our baseline method. Here, we study the overall affect of our improvements using a wide-range of standard architectures on different benchmark datasets. 38

3.9 Transfer learning experiment on Modern Office-31 dataset 39

4.1 In the following table, we present sample images, their corresponding complex channels obtained using our R2C transform, the reconstructed images from our inverse R2C transform, and the SSIM comparison scores between the original and reconstructed images. 51

4.2 Quantitative comparison of our proposed DCS^{2}Net with other real-valued RGB SOD methods on five benchmark datasets. "-R" means the ResNet50. ■: best, ■: second best. 56

4.3 Comparing classification accuracy(%) of our encoder with only other complex-valued method and other real-valued encoders on ImageNet-1k[30] 58

4.4 Comparison on Complex-valued Building InSAR dataset [14] for foreground extraction. ■: best, ■: second best. 59

4.5 Computation comparison between DCS^{2}Net and FCCN[173] 59

4.6 Comparison with real and complex-valued methods on defocus blur detection. ■: best, ■: second best. 60

4.7 Comparison with real and complex-valued approaches on shadow detection on BER (↓) score. ■: best, ■: second best. 60

4.8 Additional ablation study to compare our Fourier filter module and self-attention. 61

4.9 Results of ablation study on various datasets when we take different inputs to DCS^{2}Net. For each input we trained the network from scratch. 61

4.10 Results of ablation study that highlights the importance of our loss $\mathscr{L}_{idense}$, and dense token(T^S). All results are calculated without pre-trained encoder. 61

4.11 Results of ablation study highlighting the effect of resolution levels using pre-trained encoder in our proposed DCS^{2}Net. 62

5.1 Result of low frequency (LFreq) & high frequency (HFreq) on camouflaged object detection on three separate datasets. We also show the difference in the performance of both approaches. For each metric, the shaded cell marks the better-performing setting (green for HFreq, orange for LFreq), while the red values in the last row report the signed difference between the two settings (HFreq − LFreq). 67

5.2 Comparison of Camouflaged Object Detection performance on three datasets against task-specific and finetuning methods. Our SAMwave approach demonstrates improved performance across all datasets compared to existing methods. For every metric, the best and second-best scores are highlighted within each dataset independently, not cumulatively across datasets (■: best, ■: second best). 73

5.3 Comparison of state-of-the-art approaches on shadow detection.(■: best, ■: second best) 76

5.4 Comparison with state-of-the-art approaches on blur detection.(■: best, ■: second best) 76

5.5 Comparison with state-of-the-art methods on PolyP detection. (■: best, ■: second best) 76

5.6 The results of our ablation study highlight the importance of adapters for efficient finetuning; it also highlights the importance of each low-frequency subband (*LL*) and the rest of the high-frequency subbands (*HL*, *LH*, *HH*) in the image in the wavelet domain. The ablation experiment is done for camouflage object detection, and SAM is taken as the backbone. 77

List of Figures

FIGURE **Page**

1.1 A complex-valued neuron with input signal z_i, connection weight w_i, activation function f, and output o. 2

1.2 Summary of the contributions in the thesis shown across 3 distinct themes. 8

2.1 Illustration of a complex-valued convolution operation as proposed by [151]. To convolve a complex input feature map ($h = x + iy$) with a complex weight matrix ($W = A + iB$), the operation leverages the distributive property of complex multiplication. The output is computed using standard real-valued convolutions, cross-combining the terms to produce the final real part ($A * x - B * y$) and imaginary part ($A * y + B * x$). This forces the network to learn rich, mathematically constrained relationships between the real and imaginary channels. 15

3.1 We introduce Fully Complex-valued Convolutional Networks (FCCNs), which leverage (i) complex-valued color inputs using our novel iHSV color model, (ii) complex-valued 1x1 convolutions to turn even fully-connected layers into convolutional layers, and (iii) our novel complex-valued loss function to tackle the deficit between $|y|e^{i0}$ (complex-valued labels) and $|z|e^{i\theta}$ (complex-valued outputs). Hence, FCCNs operate entirely in complex-domain. 24

3.2 Proposed iHSV color model: Given an HSV color $I_{hsv}(p) = \{I_h(p), I_s(p), I_v(p)\}$ for a pixel p, we keep each of its elements fixed one at a time and derive three argand planes from the HSV color model. The locations of the color on these planes provide us the required complex-valued color representation $\{I_h^*(p), I_s^*(p), I_v^*(p)\}$. Note that (p) has been omitted in the symbols present in the figure for clarity. *Best viewed in color.* . . . 26

3.3 An illustration of how, using iHSV, an RGB image can be converted into complex-valued inputs required by an iCNN. 30

3.4 Real (I_{re}) and imaginary (I_{im}) components of sample images (I). . . 30

3.5 Blueprint of FCCNs: First, complex-valued convolutions are performed to obtain complex-valued feature maps. Then, adaptive average pooling is done to reduce the spatial dimensions to 1×1. At the end, 1×1 complex-valued convolutions are performed to reduce the channel-size to the number of classes. 31

3.6 Accuracy curves obtained using DCN, CNN, and FCCN approaches, revealing the faster learning capability of FCCN. 39

4.1 We introduce a novel token-based Deep Complex Spatio-Spectral Networks (DCS^{2}Net), which leverage (i) complex R2C transform for producing complex-valued inputs, (ii) Spatio-spectral module for capturing global context using complex Fourier transform, (iii) Complex T2T for progressively reducing tokens and its inverse Complex RT2T, and (iv) our novel complex-valued objective to optimize complex-valued output for binary segmentation. 42

4.2 Proposed R2C color transformation: Given an RGB color $I_{RGB}(p) = \{I_r(p), I_g(p), I_b(p)\}$ for a pixel p. We have vectors $\vec{v} = \vec{CF}$ & $\vec{u} = \vec{EF}$. (a) Using θ, we project $\vec{v}$ onto $\vec{d}_{achro}$, which gives us the real and imaginary components of the first complex number. (b) Using ϕ between the reference vector $\vec{R}$ and vector $\vec{u}$. In the argand plane $\mathbf{P}$, we locate p, giving us the second complex number. . . . 46

4.3 The overall architecture of our proposed DCS^{2}Net. It first encodes input image patch sequences to generate tokens to multiple resolutions (T_1^E, T_2^E, T_3^E), using Complex T2T. Then, we add a dense token (T^S) to the output (T^ε) of the encoder. Finally, the decoder progressively upsamples the tokens using Complex RT2T while predicting the saliency map at each step. We also optimize the generated map using our proposed loss to improve the predicted map. 52

4.4 Spatio-Spectral Module takes a complex-valued input, applies a learnable complex filter in the frequency domain, and gives a complex-valued output. . . . 53

4.5 Complex T2T and Complex RT2T modules. T2T reduces token count ($l \rightarrow l'$) by restructuring tokens into a spatial map and then repatching. RT2T reverses this process. 54

4.6 Complex ground truth formulation, where foreground pixels are actively mapped to 1 and background pixels to i. 54

4.7 Qualitative results for salient object detection. Noticeable improvements can be seen after zooming. 58

4.8 Multi-scale resolution comparison on 2 difference datasets 62

5.1 Learned weights for the previous high frequency extraction method, compared with SAMWave's real and complex adapter weights. Unlike the previous method, SAMWave's weights are not concentrated around 0. 65

5.2 Overview of SAMWave for difficult vision tasks. **Left** shows that the image is passed through SAM and our WHF (wavelet high frequency) module, where the high-frequency features are extracted. The extracted details are then combined with patch embeddings obtained from the transformer encoder. Note that the encoder is frozen; we only updated the decoder for our training. **Right** shows the overview of a single adapter used; for complex adapters, we replace the weights of MLP layers with complex values. 67

5.3 An overview of the WHF module, we first use the wavelet transform to obtain high- and low-frequency features. Then, we combine the high-frequency features to create a feature map. For complex-valued features, the methodology remains the same but utilizes a complex wavelet transform for the initial decomposition step. 69

5.4 Visual comparison of our proposed SAMWave with existing methods for the COD task. 75

CHAPTER 1

Introduction

1.1 Motivation and Background

Over the past decade, deep learning has fundamentally reshaped the landscape of computer vision. Convolutional neural networks (CNNs) [79, 79] and their successors, transformers [33] and diffusion models [58], have delivered remarkable performance across tasks such as image classification [31], segmentation [96], and object detection [124]. These models, built primarily on real-valued computations, have demonstrated impressive capacity to learn hierarchical representations from large-scale visual data [137]. However, despite their empirical success, real-valued neural networks often struggle to fully capture certain aspects of visual information, particularly those related to phase, directionality, and spectral properties.

In parallel, complex-valued neural networks (iCNNs) [55, 152] have emerged as a mathematically appealing alternative. Rooted in complex algebra, iCNNs process both magnitude and phase, allowing them to model richer, multidimensional relationships in data. Complex numbers provide an inherent way to encode amplitude (intensity) and phase (structural or directional information) jointly, a property that naturally aligns with the characteristics of many visual and physical signals. This dual representation has been shown to yield faster learning [4, 28], better regularization, and improved generalization in certain domains such as radar imaging [130], speech enhancement [24], and medical imaging [160].

Biologically, complex-valued processing also resonates with neural phenomena. Studies in neuroscience suggest that phase synchronization plays a crucial role in information transmission between cortical regions [125]. The relative phase of oscillatory signals encodes relationships among neurons, paralleling the way complex-valued representations handle coupled magnitude and phase information. These insights further strengthen the theoretical basis for exploring complex-valued computation within artificial neural networks.

Despite such promise, complex-valued deep learning remains underexplored in mainstream computer vision. The vast majority of vision models continue to rely on real-valued arithmetic. Two principal challenges hinder broader adoption. First, naturally

occurring complex-valued visual data are scarce, images captured by conventional cameras are inherently real-valued. Second, extending deep learning operations (e.g., convolution, activation, and loss computation) into the complex domain introduces non-trivial mathematical and optimization complexities. As a result, iCNNs in vision research are typically restricted to niche applications or synthetic experiments, leaving their full potential largely untapped.

This thesis addresses these challenges through a systematic investigation into how complex-valued neural networks can be effectively designed, trained, and applied to a broad class of computer vision problems. The central premise is that with careful construction of complex-valued inputs, architectures, and loss functions, one can unlock the representational richness of the complex domain for standard visual data, thereby improving both accuracy and robustness across diverse vision tasks.

1.2 Complex-valued Neural Networks: Foundations and Challenges

Complex-valued neural networks extend the standard deep learning framework by allowing neurons, weights, and activations to take complex values. Each parameter or activation consists of a real and an imaginary component, processed jointly under complex arithmetic. This extension offers several theoretical benefits. First, the ability to represent signals in terms of magnitude and phase enables more expressive feature learning, particularly for data with oscillatory or spatial-frequency characteristics. Second, the algebraic structure of complex numbers can help constrain network representations, improving generalization and stability.

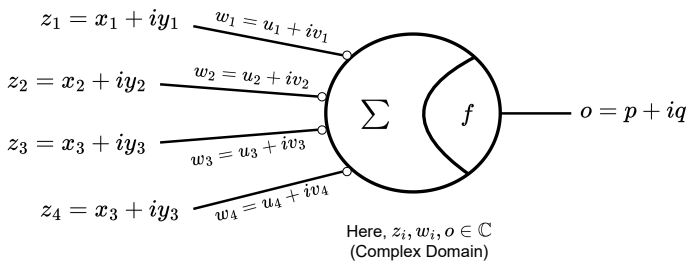


Figure 1.1: A complex-valued neuron with input signal z_i, connection weight w_i, activation function f, and output o.

To understand the essence of an iCNN, consider a single neuron whose input (z) and weight (w) are complex numbers producing a complex output (o), as illustrated in Fig. 1.1. Each neuron performs multiplication and summation in the complex domain, followed by a complex-valued activation function f. The real and imaginary

parts jointly determine the neuron's response, encoding both amplitude and phase information.

Despite the theoretical advantages of iCNNs, several practical barriers have limited their usage in computer vision:

- **Representation gap:** Standard RGB images are real-valued, and direct conversion to complex form often destroys semantic or color relationships.
- **Incomplete complex flow:** Many early iCNNs pass data through complex-valued convolutions, but prematurely force the output into standard real-valued layers or loss functions. For example, an architecture might generate rich complex feature maps, but then take only their absolute magnitude to feed into a conventional real-valued Softmax classifier. This breaks the computational chain and don't utilize the full complex-valued representation.

As a result, most previous models cannot be considered truly "complex-valued." Addressing these inconsistencies is essential for realizing the full power of complex-valued learning.

1.3 Related Work

Recent studies highlight the benefits of complex-valued learning. Works such as [4, 28, 68, 98, 178] show that iCNNs can outperform their real-valued counterparts by improving gradient regularization and robustness under adversarial settings. iCNNs also exhibit superior generalization capacity [57]. Despite these theoretical merits, their practical adoption in computer vision remains fundamentally hindered by how standard image data is processed.

While prior literature has attempted to bridge this gap, existing approaches routinely compromise either mathematical rigor or visual semantics. For example, in [151], the authors attempted image classification by treating RGB as the real part of an image and learned features as the imaginary part. However, this violates the orthogonality principle, the geometric requirement that the real and imaginary components of a complex number must reside on perpendicular, mutually independent axes. In practice, this means that knowing a value on the real axis should give zero information about its coordinate on the imaginary axis. Because the learned features in [151] are mathematically derived from the RGB inputs, the two halves remain intrinsically tied. Singhal *et al.* [138] proposed encodings such as $\{L^* + i0, a + ib\}$ and $\{R + iG, G + iB\}$, but these function as simple mathematical packaging rather than genuine complex-valued representations, as they cannot be unpacked back into meaningful, independent real and imaginary images. Similarly, Jiang *et al.* [64] combined RGB images with discrete Fourier transform coefficients for saliency prediction, yet the resulting representation remained strictly real-valued.

Even when architectures successfully maintain complex representations in deeper layers, such as the manifold learning approach in fully connected networks by Chakraborty *et al.* [12], they are ultimately forced to discard the imaginary components before the final output due to the absence of native complex-valued loss functions. *A structured comparison of these prior studies, highlighting their specific contributions and architectural bottlenecks, is summarized in Table 1.1.*

Table 1.1: Summary of representative prior studies in complex-valued learning, detailing their core contributions and primary limitations when applied to computer vision.

Study / Focus	**Core Contribution**	**Identified Limitation**
Foundational iCNNs [4, 28, 57, 68, 98, 178]	Demonstrated foundational theoretical merits of complex learning (gradient regularization, adversarial robustness, superior generalization).	Primarily theoretical or sequence-focused; lacked a principled mechanism to map standard RGB visual data into the complex plane.
Trabelsi *et al.* [151]	Pioneered deep complex-valued convolutional architectures for benchmark image classification.	Violates the orthogonality principle by directly tying RGB inputs to mathematically derived imaginary feature maps.
Singhal *et al.* [138]	Proposed multi-channel complex color encodings (e.g., $\{L^* + i0, a + ib\}$).	Functions as mathematical packaging; cannot be unpacked back into physically meaningful, independent real/imaginary images.
Jiang *et al.* [64]	Combined standard RGB spatial images with Discrete Fourier Transform (DFT) frequency coefficients.	The computational flow prematurely collapses, leaving the resulting representation strictly real-valued.
Chakraborty *et al.* [12]	Successfully maintained complex-valued representations inside fully connected layers via manifold learning.	Forced to discard imaginary phase components prior to the final layer due to a lack of native complex loss functions.

Consequently, the literature reveals a persistent pattern: prior methods address complex-valued learning in isolated architectural pockets, but inevitably collapse back into real-valued processing at either the input stage (via artificial encodings) or the output stage (via magnitude projection). In direct contrast to these fragmented pipelines, this thesis proposes an unbroken, end-to-end framework. By jointly designing principled color transformations, token-based spatial processing, and native complex loss formulations, we resolve the historical trade-off between mathematical validity and practical computer vision performance.

1.4 Bridging the Real and Complex Domains

To harness the benefits of complex-valued computation for computer vision, one must first bridge the representational gap between real-valued images and complex-valued inputs. A naive mapping of RGB values to complex components fails to preserve key perceptual relationships among color channels. Moreover, the real and imaginary parts of a complex representation should ideally be orthogonal and carry distinct, complementary information.

A central premise of this research is that meaningful complex representations can be constructed from standard RGB images by exploiting the geometry of existing color spaces. This thesis introduces two distinct real-to-complex transformations tailored to different analytical purposes:

- **iHSV Color Model:** Inspired by the cylindrical structure of the HSV color space, iHSV embeds hue, saturation, and value into orthogonal Argand planes by systematically holding variables constant one at a time, producing three interpretable complex-valued channels. This model preserves perceptual relationships, successfully decomposes an image into meaningful real and imaginary spatial components for the first time, and enables true end-to-end complex processing.
- **R2C (Real-to-Complex) Transform:** A compact, invertible mapping that converts three real-valued RGB channels into two complex channels, specifically, an *intensity channel* and a *chromaticity channel*, while preserving geometric consistency, orthogonality, and invertibility. In this formulation, the intensity channel captures overall pixel luminance (e.g., distinguishing a red car parked in deep shadow from one in direct sunlight), whereas the chromaticity channel isolates pure color identity independent of brightness (e.g., recognizing that both the shadowed and sunlit panels share the exact same underlying red hue).

These transformations not only provide a natural entry point for visual data into the complex domain, but also establish a principled framework for color and feature encoding compatible with modern end-to-end iCNN architectures.

1.5 Research Objectives and Contributions

This thesis is driven by three persistent gaps in complex-valued computer vision: **(i) the representational gap**, no perceptually grounded, invertible way to map real RGB images into the complex domain; **(ii) architectural instability**, hybrid real-complex pipelines that break the complex algebra midway; and **(iii) the optimization gap**, the absence of native complex-valued loss functions. The overarching goal is to close these gaps with end-to-end complex-valued neural networks, developed in three distinct stages that each target the gaps directly.

1.5.1 FCCNs: Fully Complex-valued Convolutional Networks using iHSV

The first phase of this research introduces Fully Complex-valued Convolutional Networks (FCCNs), the first deep architecture to maintain an unbroken, end-to-end flow of complex-valued information for visual data. FCCN directly resolves the three gaps identified above [151]:

1. **Ad-hoc Data Ingestion:** Unlike prior works that rely on mathematically convenient but perceptually flawed real-to-complex conversions (e.g., simplistic $R + iG$ mappings), we introduce the rigorous iHSV color model to provide truly complex-valued image inputs.
2. **Hybrid Architectures:** While legacy networks disruptively flatten complex features into real-valued vectors for final classification layers, FCCNs replace fully connected layers with complex-valued 1×1 convolutions to ensure all operations remain within the complex field.
3. **Real-Valued Optimization:** We overcome the traditional reliance on standard real-valued losses by formulating a native, differentiable complex cross-entropy loss that backpropagates gradients through both magnitude and phase simultaneously.

To achieve this, the iHSV color model is deployed to map standard RGB images into the complex plane. By leveraging the geometry of the HSV space, iHSV constructs orthogonal Argand planes through systematic variable isolation, with each projection yielding a distinct complex channel. This transformation preserves vital perceptual color dynamics while introducing phase angles naturally structured for complex-domain convolutions.

Extensive experiments across standard benchmarks confirm that FCCNs consistently outperform both their real-valued counterparts and legacy complex architectures. Crucially, FCCNs provide a scalable architectural blueprint, enabling benchmark validation on large-scale datasets such as ImageNet for the first time in complex vision literature.

1.5.2 DCS^{2}Net: Deep Complex Spatio-Spectral Networks

Building upon the spatial-domain foundations of FCCNs, the second contribution investigates the mathematical synergy between spatial and frequency representations. Unlike prior complex-valued CNNs that rely exclusively on computationally heavy, dense spatial convolutions, this culminates in DCS^{2}Net (Deep Complex Spatio-Spectral Network), the field's first token-based complex-valued architecture for dense visual prediction.

The foundational input layer of DCS^{2}Net utilizes the compact R2C transformation to compress RGB images into a two-channel complex tensor comprising an *intensity*

channel and a *chromaticity channel*. This mapping guarantees strict invertibility and geometric interpretability.

Internally, in stark contrast to standard dense convolutional routing, DCS^{2}Net introduces a novel spatio-spectral module that processes compressed spatial features and discrete Fourier transforms concurrently using learnable complex filters. This dual-domain mechanism allows the network to resolve fine-grained local textures while simultaneously capturing global, long-range spatial dependencies. A custom loss formulation evaluates network predictions against complex-encoded ground-truth masks (utilizing a $1 - i$ projection), explicitly enforcing geometric orthogonality between foreground objects and background noise.

DCS^{2}Net achieves state-of-the-art performance among complex-valued neural networks across salient object detection, shadow detection, and defocus blur segmentation tasks, proving that complex feature spaces possess robust representation capacity for dense pixel-level mapping.

1.5.3 SAMwave: Adapting Large Vision Models using Complex Wavelets

While earlier complex transformers were largely restricted to training from scratch on natively complex audio or sequence data, the final contribution pioneers the scaling of complex-valued representation theory to large, pre-trained vision foundation models. While the Segment Anything Model (SAM) has redefined generalizable image segmentation, its strictly real-valued architecture limits its sensitivity to high-frequency visual phenomena, creating bottlenecks in specialized tasks such as camouflaged object detection.

To resolve this limitation, this thesis introduces SAMwave, a hybrid framework that augments frozen foundational backbones using complex wavelet transforms and dedicated complex adapters. The core architecture initially utilizes a Wavelet High-Frequency (WHF) module to extract multi-scale spatial details via the discrete wavelet transform (DWT), injecting them into SAM's visual encoder through lightweight parameter-efficient adapters.

Unlike traditional adaptation methods that rely on static or real-valued frequency masks, SAMwave advances this paradigm by deploying the dual-tree complex wavelet transform (DT-CWT). By routing data through complex-valued adapters, the network captures both the structural magnitude and the multi-directional phase of high-frequency image contours, vastly enriching the foundational feature space. Extensive evaluations on camouflaged object detection, shadow detection, polyp segmentation, and defocus blur mapping demonstrate that SAMwave consistently outperforms existing adaptation techniques across both SAM and SAM2 backbones.

Together, these three contributions form a unified methodological progression: transitioning from foundational pipeline design (FCCNs), through specialized architectural synthesis (DCS^{2}Net), to large-scale foundation model adaptation (SAMwave), estab-

lishing a comprehensive paradigm for complex-valued deep learning in mainstream computer vision (see summary of contribution in Fig. 1.2).

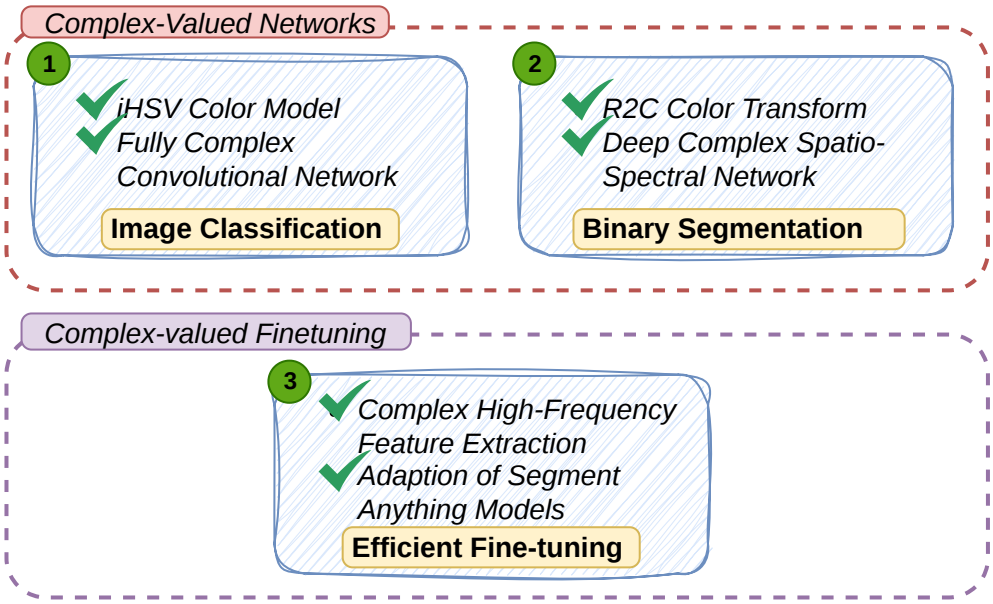


Figure 1.2: Summary of the contributions in the thesis shown across 3 distinct themes.

1.6 Future Directions

The developments presented in this thesis lay a strong foundation for integrating complex-valued computation into mainstream computer vision. Beyond the explored tasks, several directions emerge:

- **Vision-Language and Multimodal Models:** Complex numbers can model semantic and cross-modal alignments between visual and linguistic modalities.
- **Semi/Unsupervised Learning:** Phase-sensitive representations may enhance learning from unlabeled data, reducing annotation dependence.
- **Complex-valued Generative Models:** Extending GANs and diffusion models to the complex domain can improve fidelity by modeling structural and phase information.
- **Foundational Extensions:** The design principles, color transformations, spatio-spectral processing, and complex losses, offer building blocks for next-generation architectures, representation and optimization in the complex domain.

1.7 Thesis Organization

The remainder of this thesis is structured as follows:

- **Chapter 2:** It reviews the evolution of complex-valued deep learning and its emerging significance for computer vision.
- **Chapter 3:** It presents the formulation of *Fully Complex-valued Convolutional Networks (FCCNs)* with the iHSV color model and complex loss.
- **Chapter 4:** It describes our *Deep Complex Spatio-Spectral Network (DCS^{2}Net)* and R2C transformation.
- **Chapter 5:** It introduces *SAMWave*, extending complex-valued learning to foundation models.
- **Chapter 6:** It concludes the thesis and discusses possible future directions.

1.8 Summary

In essence, this thesis presents the systematic study of complex-valued neural networks for computer vision, bridging the gap between theoretical promise and practical realization (summary of contributions shown in Fig. 1.2). By introducing novel representations, architectures, and learning paradigms, it establishes complex-valued learning as a powerful and generalizable framework capable of enhancing visual perception and understanding, going beyond the usual real-valued neural networks.

CHAPTER 2

Background

In recent years, deep learning has been a driving force in artificial intelligence research. While application areas where real-valued data is available can swiftly adopt deep learning, not all domains can seamlessly harness the advantages offered by existing real-valued neural networks. Particularly, domains dealing with complex-valued data face challenges in integrating conventional real-valued networks into their applications. Active research in complex-valued applications within the realm of deep learning includes radio frequency (RF) signal classification [112], synthetic aperture radar (SAR) target detection [16, 67], and MRI [144]. However, the prevalent approach involves breaking the data into independent real and imaginary channels or discarding the phase of the complex data when feeding it to neural networks.

Early studies in complex-valued deep learning revealed that complex networks could capitalize on mathematical properties, such as the complex identity theorem, to learn more efficient transform functions compared to their real-valued counterparts [41, 53]. These initial works experimented with various complex-valued activation functions [41, 54]. However, these attempts encountered limited success when applied to real-world scenarios. More recent efforts in complex-valued deep learning are directed towards exploring its applications in solving computer vision problems.

Computer vision encompasses the utilization of computers to interpret and comprehend information present in images, involving tasks such as recognition, manipulation, segmentation, and addressing various image-related challenges. The advent of deep learning in computer vision has significantly broadened the scope of research in this field, traditionally relying on real numbers in deep learning architectures for solutions. While studies exploring alternative methods have existed for some time [53, 54, 56], research primarily focused on applying deep learning architectures with complex numbers in domains inherently featuring complex-valued data [26, 40, 41, 49, 160]. Recently, a shift has occurred with a few studies directing their attention towards applying these networks in computer vision tasks where data is real-valued [22, 64, 66, 98, 138, 151]. This shift highlights the untapped potential in this research area, sparking a new wave of applications in various domains, which will be discussed in the following sections.

2.1 Complex Numbers: Definitions and Properties

Before getting into the motivation and background for complex-valued deep learning in computer vision, we should first lay out the basics of complex numbers and their properties.

2.1.1 Fundamental Representations

A complex number, $z \in \mathbb{C}$, consists of a real component, $x \in \mathbb{R}$, and an imaginary component $y \in \mathbb{R}$.

$$z = x + iy \tag{2.1}$$

where $i = \sqrt{-1}$. We can also write the same complex number in polar form in terms of magnitude $m \in \mathbb{R}^+$, and phase $\theta \in \mathbb{R}$, and expand it using Euler's formula:

$$z = me^{i\theta} = m(\cos\theta + i\sin\theta) \tag{2.2}$$

A complex number z can be visualized as a point (x, y) in a 2D Cartesian plane called the complex plane or Argand plane. The horizontal axis represents the real part $\mathrm{Re}(z)$, and the vertical axis represents the imaginary part $\mathrm{Im}(z)$.

The **complex conjugate** $(\bar{z})$ of a complex number z is a reflection of the complex point z over the real axis. It may be computed by negating the imaginary component or the phase:

$$\bar{z} = x - iy \tag{2.3}$$

$$\bar{z} = me^{-i\theta} \tag{2.4}$$

The real and imaginary components, magnitude, and phase of z can be accessed as follows:

$$\text{Real:} \quad x = \mathrm{Re}(z) = \frac{1}{2}(z + \bar{z}) \tag{2.5}$$

$$\text{Imaginary:} \quad y = \mathrm{Im}(z) = \frac{1}{2i}(z - \bar{z}) \tag{2.6}$$

$$\text{Magnitude:} \quad m = |z| = (z\bar{z})^{\frac{1}{2}} = \sqrt{x^2 + y^2} \tag{2.7}$$

$$\text{Phase:} \quad \theta = \angle z = \text{atan2}(y, x) \tag{2.8}$$

Note that we use the $\text{atan2}(y, x)$ function to define the phase, as it correctly computes the principal argument of z in the interval $(-\pi, \pi]$.

2.1.2 Basic Arithmetic Operations

The fundamental operations of a neural network layer involve addition and multiplication. For two complex numbers, $z_1 = x_1 + iy_1$ and $z_2 = x_2 + iy_2$, these are defined as:

- **Addition:** $z_1 + z_2 = (x_1 + x_2) + i(y_1 + y_2)$
- **Multiplication (Cartesian):** $z_1 z_2 = (x_1 x_2 - y_1 y_2) + i(x_1 y_2 + y_1 x_2)$

Expressed in polar coordinates, $z_1 = m_1 e^{i\theta_1}$ and $z_2 = m_2 e^{i\theta_2}$, multiplication has a clear geometric interpretation as a scaling and rotation:

$$z_1 z_2 = (m_1 e^{i\theta_1})(m_2 e^{i\theta_2}) = (m_1 m_2) e^{i(\theta_1 + \theta_2)} \tag{2.9}$$

This property is a key motivator for complex-valued neural networks, as a single complex-valued weight can simultaneously scale and rotate its input, a richer representation than a single real-valued weight.

2.1.3 The $\mathbb{R}^2$ Isomorphism

A complex number $z = x + iy$ is isomorphic to a 2D real-valued vector $\mathbf{z} \in \mathbb{R}^2$. This identification is the basis for the common workaround of simulating complex operations within a real-valued network:

$$z = x + iy \quad \Longleftrightarrow \quad \mathbf{z} = \begin{bmatrix} x \\ y \end{bmatrix} \tag{2.10}$$

Using this isomorphism, a complex linear operation $w \cdot z$ (where $w = u + iv$) can be expressed as a 2×2 real-valued matrix operation.

$$wz = (u + iv)(x + iy) = (ux - vy) + i(vx + uy) \tag{2.11}$$

This corresponds to the following matrix-vector product in $\mathbb{R}^2$:

$$\begin{bmatrix} u & -v \\ v & u \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} ux - vy \\ vx + uy \end{bmatrix} \tag{2.12}$$

This shows that true complex multiplication is equivalent to an operation with a highly *constrained* 2×2 real matrix (a rotation-scaling matrix). A standard real-valued neural networks, in contrast, would use an unconstrained 2×2 matrix, learning a different, and potentially less suitable, function.

2.2 Real-to-Complex Image Representations

A critical prerequisite for applying iCNNs to computer vision is the method of converting real-valued images into a complex-valued format. This encoding step is not trivial; the chosen representation dictates what information is fed into the network's real and imaginary channels and can significantly impact model performance. These methods range from simple baseline mappings to sophisticated, domain-aware transforms.

2.2.1 Baseline Encoding Strategies

The most direct methods involve mapping pixel intensities directly to the components of a complex number.

- **Real-Part Encoding:** The most straightforward approach is to place the real-valued pixel intensity x (or a single channel) into the real part, setting the imaginary part to zero: $z = x + i \cdot 0$. This is a common baseline but fails to leverage the expressive capacity of the complex domain.
- **Phase Encoding:** An alternative is to use the pixel value to modulate the phase of a complex number, while keeping the magnitude constant: $z = M \cdot e^{i \cdot \theta(x)}$. Here, M is a constant (e.g., $M = 1$) and $\theta(x)$ is a scaled mapping of the intensity, often to the $[0, 2\pi)$ interval. This approach is common in physics and signal processing.
- **Dual-Channel Encoding:** For 2-channel data (e.g., two different sensor modalities, or two color channels), one can map each channel to one component: $z = \text{channel}_1 + i \cdot \text{channel}_2$.

2.2.2 Transform-Domain Representations

A common and powerful family of methods involves first converting the image into a different domain, many of which are naturally complex-valued.

- **Fourier Transform:** The Fourier Transform (FT) is a powerful tool for analyzing an image's frequency content. The resulting spectrum is complex, with a magnitude and a phase. However, the FT is a *global* transform; its basis functions (sines and cosines) span the entire image. This makes it less effective for capturing localized color and spatial variations, which are crucial for many image processing tasks.
- **Wavelet and Gabor Transforms:** To address the localization issue of the FT, local transforms like the Gabor transform or other complex-valued wavelets are a strong alternative. A Gabor filter, for example, is a complex sinusoid modulated by a Gaussian envelope. This allows it to capture localized frequency *and* orientation information, making it highly effective for texture analysis and edge detection.
- **Hilbert Transform:** Other methods, such as the Hilbert transform, are primarily defined for 1D signals. When applied to 2D images, they are typically restricted to processing one color channel or one dimension at a time, limiting their ability to model holistic 2D or 3D color structures.

2.2.3 Color-Specific and Hypercomplex Representations

For color images, more sophisticated methods have been proposed to explicitly handle the 3-channel (e.g., RGB) data.

- **Quaternion Representation:** One prominent approach is the quaternion representation, which uses a 4D hypercomplex number. A color pixel is often encoded as a "pure" quaternion: $q = 0 + iR + jG + kB$. While this elegantly packages all three channels into a single algebraic object, it has been critiqued for operating on a 4D hypercomplex space without explicitly considering the inherent perceptual geometry of the underlying color space [44, 193].
- **Geometric Color Mappings:** Recent advancements have focused on mapping the geometry of standard color spaces directly into the 2D complex plane. [138], for example, proposed two such strategies:
  - **Lab Chrominance Mapping:** This approach leverages the CIELAB (Lab) color space, which is perceptually uniform and designed to decouple luminance (L*) from the two chrominance axes (a* and b*). This structure is then mapped into a two-channel complex representation, with one channel representing luminance ($L^* + i0$) and the second representing chrominance ($a^* + ib^*$).
  - **RGB Sliding Window:** This second approach operates directly on the RGB color space. It uses a "sliding window" concept across the three channels to create two overlapping, complex-valued channels: $z_1 = R + iG$ and $z_2 = G + iB$. This representation is designed to capture the immediate correlations between adjacent color channels (Red-Green and Green-Blue).

2.3 Applications of iCNNs in Computer Vision

The application of Complex-Valued Neural Networks (iCNNs) in computer vision and image processing has a history spanning several decades, preceding the current deep learning boom. Early research in the 1990s, for example, employed iCNNs for tasks such as optical flow [106] and explored their integration with holographic movies [149]. These networks also demonstrated effectiveness in classical image processing tasks like grayscale image reconstruction [147] and image deblurring [1, 3].

More recently, research has moved beyond using iCNNs for specific image tasks to exploring their fundamental architectural advantages. A notable example is presented in [4], where the authors investigated the use of unitary weight matrices, a concept intrinsically linked to complex operations, to address the vanishing gradient problem. This work highlighted the potential for complex-valued representations to improve the training stability and gradient flow of deep neural networks.

Beyond standard gradient-based deep learning, alternative complex-valued paradigms have also been successfully applied. For instance, the complex neural network with multi-valued neurons (MLMVN) was proposed as an intelligent image filter [2]. In this non-gradient-based approach, integer pixel intensities from an $n \times n$ region were first mapped to complex inputs. The MLMVN then acted as a filter, and the outputs from overlapping regions were averaged and transformed back to integer intensities. This method proved to be highly efficient, capable of producing excellent nonlinear filters after training on as few as 400 images.

In the subsequent subsections, we will delve into the application of iCNNs in several other crucial computer vision problems.

2.3.1 Image Recognition

Image recognition, a foundational challenge in computer vision focused on image identification and categorization, has been dominated by deep learning solutions since 2015. A significant line of research has explored adapting successful real-valued architectures directly into the complex domain. Notably, [46], building on the foundational work of [151], implemented a complex-valued network. In this approach, core components of the real-valued architecture, including 2-D convolution operations (Fig. 2.1), batch normalization, and ReLU activation functions, were adapted to operate on complex numbers. The resulting complex-valued model reportedly outperformed its real-valued counterpart in both training and testing accuracy when evaluated on the CIFAR-10 benchmark.

However, this direct adaptation has been a subject of further investigation. Subsequent work by [138] suggested that these earlier models [46, 151] are sensitive to complex-valued scaling. Other related frameworks, such as the manifold-valued deep learning approach in [12], were also noted for capturing only the manifold aspect of the data while neglecting its underlying complex algebra. To address these limitations, [138] proposed novel layer functions for complex-valued deep learning, exploring how they preserve co-domain symmetry, particularly in the context of small, lightweight networks using various color encodings.

Shifting from convolutional to recurrent architectures, [4] assessed a different aspect of iCNNs on the MNIST handwriting benchmark. Their unitary-RNN was evaluated in two modes: one reading pixels sequentially and another reading them in an arbitrary, permuted order. While a real-valued LSTM performed slightly better in the standard sequential mode, the unitary-RNN exhibited superior performance in the more challenging permuted mode. This result was particularly significant as the unitary-RNN achieved this with less than a quarter of the parameters of its real-valued LSTM counterpart, demonstrating a high degree of parameter efficiency.

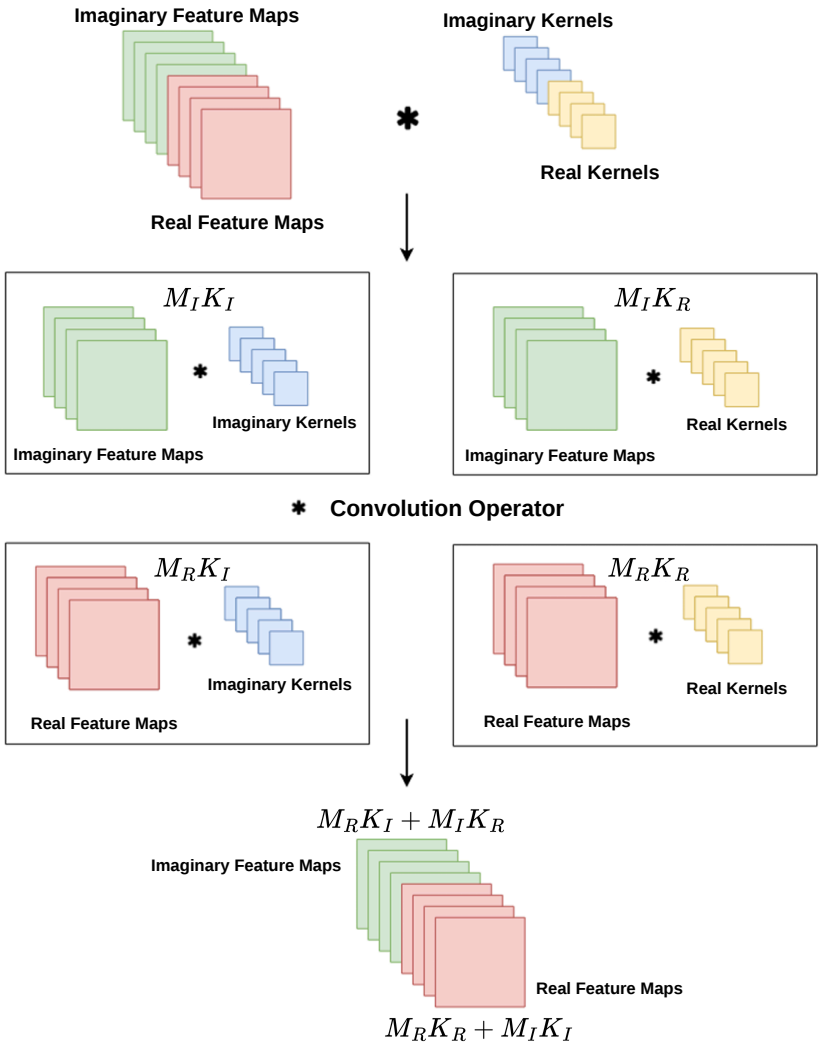


Figure 2.1: Illustration of a complex-valued convolution operation as proposed by [151]. To convolve a complex input feature map ($h = x + iy$) with a complex weight matrix ($W = A + iB$), the operation leverages the distributive property of complex multiplication. The output is computed using standard real-valued convolutions, cross-combining the terms to produce the final real part ($A * x - B * y$) and imaginary part ($A * y + B * x$). This forces the network to learn rich, mathematically constrained relationships between the real and imaginary channels.

2.3.2 Binary segmentation

Saliency prediction, which aims to model and predict human visual attention, is a widely explored task in computer vision. Over the past decade, methods operating in a transformed domain (such as the Fourier domain) have proven highly successful. The intuition behind this success is that a transform, like the Discrete Fourier Transform (DFT), decouples an image's spatial information into its constituent components: the amplitude spectrum (representing the "what," or the prevalence of certain frequencies) and the phase spectrum (representing the "where," or the spatial arrangement and structure). Many saliency cues are believed to be more easily distinguished in this separated domain than in the pixel domain alone.

Drawing inspiration from general iCNN principles ([151]), [65] introduced the Sal-DCNN method to explicitly test this hypothesis. Their approach was novel in that it learned features from both the pixel and transformed domains simultaneously. Their investigation yielded several valuable insights into how saliency is encoded:

- The phase spectrum encodes the vast majority of saliency cues. This is highly intuitive, as phase corresponds to object structure and location, which are primary drivers of human attention.
- A specific pattern in the amplitude spectrum is also crucial for accurate prediction.
- The transformed domain representation is significantly more robust to noise and down-sampling, making it a more stable feature space.

Building on these findings, the study proposed the Sal-DCNN encoder-decoder architecture, which uses a complex dense encoder to effectively extract and fuse features from both the pixel and DFT domains.

This line of research was further extended from static images to video saliency in [66]. This work introduced the Deep Complex-valued Neural Network with Learnable Transform (DeepCT). Instead of relying on a fixed transform like the DFT, DeepCT takes the concept a step further: it learns the optimal spatio-temporal transform for the task.

At the core of DeepCT are two key components:

- A Complex-valued Transform Module (CTM), which learns a custom complex-valued transform to capture both temporal and multi-scale information from the input video.
- An inverse Complex-valued Transform Module (iCTM), which serves as the learnable inverse, reconstructing the spatio-temporal data from the complex feature space.

To ensure that the CTM and iCTM truly learn to be a pair of inverse structures, [66] introduced a 2-cycle consistency loss. This loss enforces reconstruction in both directions:

(1) a frame reconstruction loss (pixel $\rightarrow$ transform $\rightarrow$ pixel) and (2) a complex feature reconstruction loss (transform $\rightarrow$ pixel $\rightarrow$ transform). This encourages the network to find a powerfully expressive and stable complex representation that is fully invertible.

Camouflaged Object Detection (COD) distinguishes objects seamlessly blended into their background [38, 47, 83], requiring models to capture subtle boundary cues [113, 179]. Defocus Blur Detection (DBD) identifies out-of-focus image regions by leveraging variations in local blur kernels and edges [69, 134, 148]. Shadow Detection (SD) distinguishes shadows from true objects by analyzing illumination patterns and local spatial context [60, 192]. PolyP Detection (PD) delineates polyp regions in endoscopic images, assisting clinicians in the early detection and treatment of colorectal abnormalities [132, 154]. Forgery Detection (FD) locates manipulated pixels, where inconsistencies in noise or texture often manifest as high-frequency anomalies [27, 39].

2.3.3 Object Discovery and the Binding Problem

Object discovery, a pivotal capability in human perception and cognition [73, 74], is closely related to the long-standing binding problem [45]: how does the brain group disparate features (e.g., color, shape, motion) into coherent objects, using a network of relatively fixed connections?

In machine learning, the dominant effort to solve this problem has revolved around slot-based approaches [9, 52]. These models explicitly partition a latent representation into "slots," where each slot is dedicated to representing a distinct object. While highly interpretable, this strategy often requires adding a specialized, object-centric module to a model that may not possess inherent object-centric features, which can present integration challenges.

In contrast, [98] proposes a more direct solution by using complex-valued activations to generate distributed, object-centric representations implicitly. Their model, a Complex AutoEncoder (CAE), is based on a hypothesis of how biological neurons might use phase to encode information. The core concept is that a single complex-valued activation conveys two distinct messages:

- The magnitude ($|z|$) indicates the presence or intensity of a particular feature.
- The phase ($\angle z$) acts as a binding signal. Features with similar phases are grouped together as belonging to the same object.

This approach allows the network to learn object groupings in a fully unsupervised, end-to-end fashion. The authors argue that this method leads to substantial improvements in performance and efficiency over previous complex-valued methods for this task. This work was subsequently improved by [142], which introduced key architectural changes and a new contrastive loss. This additional loss, applied alongside the

main reconstruction loss, encourages pixels with similar phases (which are hypothesized to be from the same object) to cluster more tightly, thereby enforcing a clearer separation between distinct objects in the representation space.

In a separate, though architecturally related, line of study, [5] also utilized a complex-valued Convolutional AutoEncoder (CV-CAE). This work, however, was focused on data reconstruction for complex-valued SAR imagery (CV-SAR) rather than on solving the object discovery problem in natural images.

2.4 Advanced Complex Architectures for Visual Representation

While much of the early literature focused on standard convolutional networks, recent advancements have expanded into architectural paradigms highly relevant to the core methodologies of this thesis. Specifically, the development of complex-valued attentional mechanisms and robust metric learning strategies directly informs the adaptation of foundation models and the extraction of highly discriminative visual features.

2.4.1 Complex Transformers and Attentional Learning

While CNNs, RNNs, and LSTMs have long been foundational in deep learning, the introduction of the transformer model has caused a transformative shift, with its architecture being widely adopted across diverse domains. As this thesis explores in later chapters through the adaptation of the Segment Anything Model (SAM), extending attention mechanisms into the complex domain offers a rich representational potential that real-valued models cannot natively capture.

Adapting the Transformer Architecture

Endeavors to create a truly complex-valued transformer have historically been limited to signal-based domains. A pioneering effort in this area was presented by [177], which introduced a Complex Transformer. This work utilized the transformer as a foundational framework for sequence modeling, developing the necessary complex-valued attention mechanisms and encoder-decoder networks designed to operate directly on complex inputs.

More recently, [34] focused on establishing a robust generalization of the transformer's core building blocks in the complex domain. Their work provided experimental evidence that a properly formulated complex transformer can improve resilience to overfitting while maintaining performance comparable to its real-valued counterpart, a property highly desirable for scaling vision models.

Complex Attention in Meta-Learning

Beyond adapting the full transformer, the core concept of complex attention is also being applied to solve other fundamental challenges, such as data scarcity. To address this challenge, particularly for *few-shot signal recognition*, [32] proposed the Complex-valued Attentional Meta Learner (CAMEL). This novel strategy integrates complex-valued attention mechanisms with a model-agnostic meta-learning (MAML) framework. By operating in the complex domain, CAMEL more effectively captures the invariant representations of complex-valued signals, allowing it to learn robustly from a very limited number of training samples.

2.4.2 Few-Shot and Metric Learning in Vision

In contrast to conventional few-shot learning (FSL) methods, which are typically based on real-valued features, the investigations in [95] and [169] proposed a move into the complex domain for visual tasks. Their core idea aligns closely with the representation objectives of this thesis: leveraging complex-valued CNNs (CV-CNNs) to generate richer feature representations that explicitly encode both amplitude and phase information.

Simply extracting complex features, however, is insufficient without a method to properly compare them. Therefore, the primary contribution of this work was an innovative metric learning method designed specifically for complex embeddings. This distance metric gauges the similarity between samples by evaluating both their amplitude and phase information, rather than treating them as a simple 2D real vector.

This framework was encapsulated in a novel Complex Metric Module (CMM), which seamlessly integrates the deep complex-valued CNN with the new complex-valued distance metric. By operating holistically in the complex domain, this module allows for a more effective use of transductive inference, where the entire query set is considered, thereby successfully addressing the data scarcity challenges inherent in visual few-shot problems.

2.5 Complex-Valued Deep Learning in Tangential Domains

Beyond the visual and architectural domains directly related to the methodologies developed in this thesis, the theoretical principles of complex-valued deep learning, such as phase-based relational binding and frequency-domain processing, have also been successfully applied to several tangential fields. These applications demonstrate the broader utility of complex numbers in fields where data exhibits inherent relational, sequential, or spectral properties.

2.5.1 Link Prediction in Knowledge Graphs

Knowledge Graphs (KGs) organize data as directed graphs, connecting nodes (entities) with labeled edges (relations). A key task in this domain is link prediction, which aims to infer missing entries (links) by uncovering regularities in the graph.

A primary method for link prediction involves learning low-dimensional vector representations, or embeddings, for all entities and relations. A simple scoring function, like the dot product, is efficient but deeply flawed in the real domain because a real-valued dot product is *symmetric*, meaning it cannot distinguish the direction of a relation. To address this, [156] (the ComplEx model) moved from real-valued to complex-valued embeddings ($v \in \mathbb{C}^k$). This allowed the model to use the Hermitian (or sesquilinear) dot product. This operation, which involves the conjugate-transpose of one vector, is inherently *asymmetric*. This asymmetry is the key, allowing the model to effectively capture antisymmetric relations (like `isChildOf`) while retaining high efficiency.

A subsequent concise study [155] compared ComplEx with another state-of-the-art model, HOLE [109], and established their mathematical equivalence. Venturing beyond standard complex numbers, [182] explored even more expressive hypercomplex representations, employing quaternion embeddings to capture even richer relational patterns.

2.5.2 Natural Language Processing (NLP)

A persistent challenge for word embeddings is capturing the nuanced, compositional meaning that emerges from combinations of words. Traditional co-occurrence models may learn that "tigers" and "fly" are related, but they fail to capture the semantic nuance that this combination is factually incorrect.

[86] posits that words do not have a single, fixed polarity. Instead, a word's contribution to a phrase's meaning depends entirely on its contextual pairing. Drawing a parallel to quantum mechanics, they proposed that complex-valued embeddings can model this phenomenon using *phase* as a mechanism for constructive or destructive interference. In this model, the phases of word vectors interact, allowing the network to learn nuanced relationships, such as "tigers" + "fly" resulting in a low-probability, "interference" state.

In a different line of inquiry, [163] addressed the modeling of word positions by extending each word embedding from a discrete vector into a continuous complex-valued function over the "position" variable. This allows the model to learn a continuous correlation between a word and its neighbors, rather than relying on discrete positional jumps.

2.5.3 Audio and Speech Processing

Audio signals can be mathematically represented in two equivalent ways: as a real-valued waveform in the time domain, or as a complex-valued spectrum in the frequency

domain. This complex spectrum, composed of magnitude and phase, is particularly powerful as the human auditory system processes sound primarily in this domain [129].

A major challenge in speech enhancement has been the "phase problem." Early models focused exclusively on estimating the spectrogram's *magnitude* while reusing the *phase* from the original noisy signal, limiting achievable quality. To address this, [24] introduced a Deep Complex U-Net (DCUnet) that operates entirely in the complex domain. By treating the spectrum as a complex entity, the network learns to estimate both the clean magnitude and the correct corresponding phase simultaneously, significantly improving speech enhancement performance. This trend is supported by [171], which provides a critical assessment of complex-valued atomic units against their real-valued counterparts for audio tasks.

Expanding on this, [48] proposed a filter-and-sum framework for multi-channel speech enhancement. Their method uses a complex-valued spatial autoencoder to estimate a *complex-valued mask* for each microphone. Unlike a real-valued mask which can only attenuate or amplify, a complex-valued mask manipulates both the amplitude and the phase, allowing the network to exploit spatial characteristics (e.g., sound source direction) for superior speech quality.

2.6 Summary and Remaining Gaps

In summary, this chapter reviewed the evolution of complex-valued deep learning and its emerging significance for computer vision. We established the mathematical foundations of complex numbers and their relevance to neural computation, emphasizing how complex operations enable joint scaling and rotation, an expressive capacity absent in standard real-valued models. We also explored various real-to-complex encoding strategies, alongside the historical progression of complex-valued networks across vision tasks such as recognition, segmentation, and phase-based object discovery.

Despite the extensive theoretical merits documented in the literature, this review reveals several critical gaps that continue to impede the practical, mainstream adoption of iCNNs in computer vision:

- **The Representational Gap:** Because complex-valued networks were historically developed for domains with natively complex data (e.g., SAR, MRI), converting real-valued RGB images into complex tensors remains a fundamental bottleneck. Prevalent baseline methods, such as mapping pixels to the real axis and zeroing out the imaginary component, fail to leverage the true expressive capacity of the complex domain. Conversely, hypercomplex mappings, such as quaternions, process data in 4D space without explicitly considering the inherent perceptual geometry of standard color spaces.
- **Architectural Instability:** Direct adaptations of real-valued convolutional architectures to the complex domain have proven to be highly sensitive to complex-

valued scaling. Furthermore, alternative frameworks (such as manifold-valued deep learning) often capture only the manifold aspect of the visual data while entirely neglecting the rigorous algebra required for true complex-valued computation.

- **Incomplete Computational Flow:** There is a persistent tendency in existing literature to either break data into completely independent real and imaginary channels, or to prematurely discard the phase of the complex data before feeding it into downstream neural network operations. This collapses the representation back into a real-valued state, effectively destroying the rich geometric and relational properties that iCNNs are fundamentally designed to protect.

These persistent limitations, the lack of perceptually grounded complex color transformations, the architectural instability of direct adaptations, and the inability to maintain an unbroken complex algebraic flow, highlight the acute need for a unified framework. Resolving these specific gaps forms the primary motivation for this thesis, setting the stage for the novel real-to-complex representations and end-to-end architectures developed in the subsequent chapters.

CHAPTER 3

FCCNs: Fully Complex-valued Convolutional Networks

In the ever-evolving deep learning era, complex-valued neural networks (iCNNs) show great promise, given their ability to learn complex-valued representations. Nevertheless, iCNNs have some inconsistencies which should be addressed: (i) Input images are real-valued. (ii) They are not fully complex-valued. (iii) They have real-valued losses. Because of these inconsistencies, we don't witness an end-to-end flow of complex-valued information in these networks. As a result, it's difficult to declare them as truly complex-valued. This study aims to weed out these inconsistencies and uncover their true potential.

However, several challenges are involved: (i) No complex-valued color model is readily available to allow us to have real and imaginary components of an image, as shown in Fig. 3.1. (ii) No prior study exists on how fully-connected layers can be made complex-valued to have an end-to-end flow of complex-valued information. (iii) There are no well-tested complex-valued loss functions for deep learning approaches.

Despite these challenges, what motivates us to explore complex-valued representation for color images is the sheer potential complex-valued networks have shown in other domains such as MRI [26, 160], radar signals [40, 41] and audio signals[49, 61], where complex-valued inputs are readily available. As per the recent studies conducted in [4, 28, 68, 98], iCNNs have performed better than their real-valued counterparts. Moreover, they are biologically motivated [125] and possess greater generalization capacity[57], which inspires us to explore HSV color space for such inputs since it aligns with how human vision perceives color-making attributes.

In the past, in [151], authors have tried image classification using iCNNs inputting RGB as the real part and learned features from RGB as the imaginary part of an image. However, we argue that components of complex numbers are supposed to be orthogonal, which means they are mutually independent. So, it should not be possible to derive one part from another. In [138], authors look for orthogonality and propose to use $\{L^* + i0, a+ib\}$ and $\{R+iG, G+iB\}$ as possible complex-valued inputs. Note that these are more like complex-valued encodings than a complex-valued representation of an image. That is because we cannot decompose an image based on them into real and imaginary

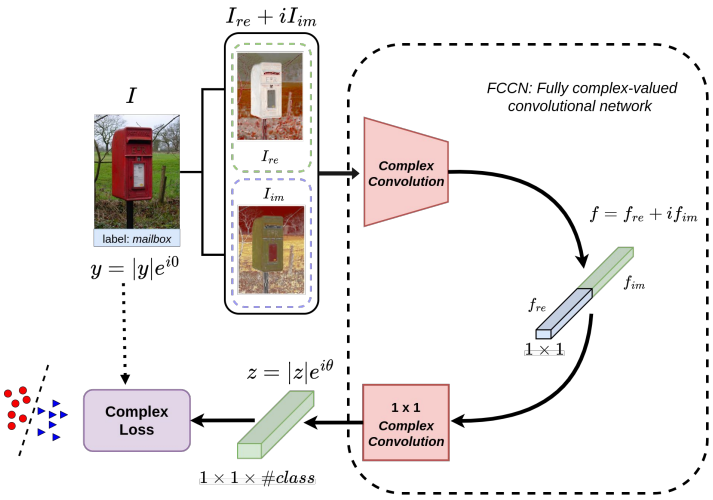


Figure 3.1: We introduce Fully Complex-valued Convolutional Networks (FCCNs), which leverage (i) complex-valued color inputs using our novel iHSV color model, (ii) complex-valued 1x1 convolutions to turn even fully-connected layers into convolutional layers, and (iii) our novel complex-valued loss function to tackle the deficit between $|y|e^{i0}$ (complex-valued labels) and $|z|e^{i\theta}$ (complex-valued outputs). Hence, FCCNs operate entirely in complex-domain.

parts (as shown in Fig. 3.1) that look like images by themselves. Moreover, HSV color space is not explored yet for orthogonality and complex-valued representation.

There have also been attempts to preserve complex-valued information in fully-connected layers through manifold learning [12]. Nevertheless, the complex-valued representation is sacrificed, and that was done due to the lack of complex-valued losses; otherwise, imaginary or phase information would be required to be left out at the end. Thus, there is a need for complex-valued loss functions to preserve complex-valued representations throughout, and we can derive complex-valued loss function from prevalent cross-entropy loss itself. Moreover, since fully-connected layers can be implemented through convolution operations and complex-valued convolutions already exist, we can have fully complex-valued networks.

More specifically, we develop FCCNs (Fully Complex-valued Convolutional Networks), which take complex-valued images as inputs, perform only complex-valued operations, and have a complex-valued loss function. For inputs, we observe that any color in the cylindrical HSV color model has intersecting color planes. For e.g. (H_0, S_0, V_0) color will have $H = H_0$, $S = S_0$ and $V = V_0$ planes. If we consider these planes as argand planes, we can obtain complex numbers using the locations of that color on them. We use these complex numbers to represent any color. We call this color model iHSV. Following this, we can generate a complex-valued representation of any image $I \rightarrow I_{re} + iI_{im}$, as shown in Fig. 3.1, where we can separate an image into real and imaginary parts (images). When images are supplied in this form to FCCNs, obtained by

implementing fully-connected layers as convolutional layers, the complex-valued loss at the end ensures that there is only the flow of complex-valued information throughout.

Our contributions in this study are three-fold: (i) We propose iHSV, a novel complex-valued color model. (ii) We derive a complex-valued loss function from the cross-entropy loss. (iii) We turn iCNNs into FCCNs by implementing fully-connected layers as convolutional layers.

3.1 Related Work

Complex-valued Learning: In the realm of deep learning, the use of complex numbers as weights presents new opportunities for two-dimensional spectra exploration[56, 157]. Furthermore, [125] and [64] provide evidence for the importance of phase information in the firing rate of neurons, highlighting the potential benefits of using complex-valued representations in neural networks. Specifically, synchronized neurons with similar phases fire together, while asynchronous neurons with different phases interfere. This behavior aligns more closely with the functioning of biological neurons, and the passage of synchronized inputs through neurons can be likened to the gating mechanism in both deep feedforward and recurrent neural networks [70, 141, 158].

In recent years, complex-valued networks have been shown to have greater generalization capacity, as demonstrated in previous works [57, 68]. Complex-valued autoencoders have also been shown to outperform slot-attention in object-centric learning [98]. Additionally, complex-valued networks have been utilized to enhance saliency prediction [64, 66] and iris recognition [108]. The work presented in [22] further demonstrated that using a complex-valued vector can be helpful for learning to solve multiple tasks.

Complex-valued Representation: Incorporating the phase component in deep learning models is also motivated from a signal-processing perspective. [110] demonstrate the importance of phase in speech signals and images. Similarly, [23] use complex-valued U-net to improve speech enhancement by incorporating phase information. According to [110], humans rely more on the phase component for perception. Recently, [13] developed a new augmentation technique by recombining phase and magnitude, resulting in increased robustness of CNNs by focusing more on the phase information. Although their method uses Fourier transform to generate phase, it highlights the potential benefits of using complex representations in learning approaches. Complex-valued networks can directly process complex-valued inputs in the spatial domain, but as images are real-valued, a complex-valued color model is required. This research gap is addressed in this study through the proposed iHSV color model.

In the complex-valued convolutional neural network (iCNN) proposed by [151], the complex-valued activations from the convolutional layers are flattened and fed into real-valued fully connected layers by concatenating their real and imaginary parts. As a result, the complex-valued information is lost during propagation through fully connected layers. To address this issue, SurReal[12] used manifold learning, but this approach still

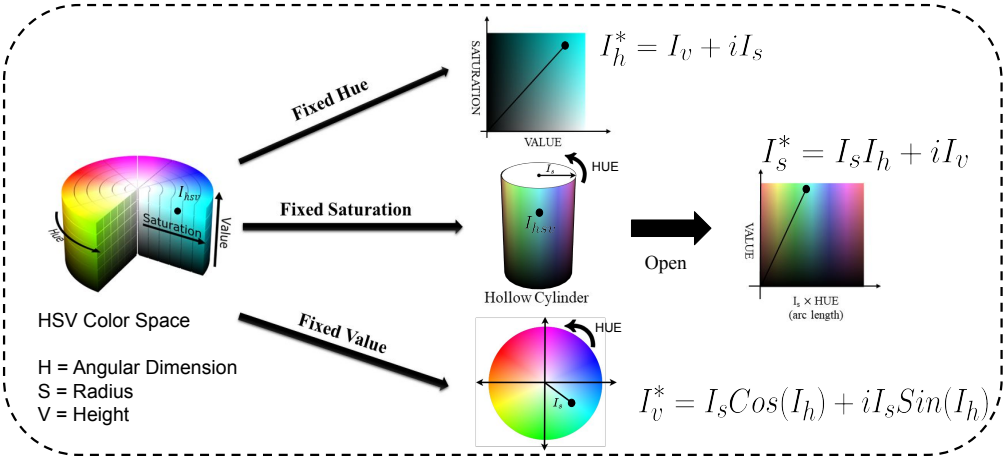


Figure 3.2: Proposed iHSV color model: Given an HSV color $I_{hsv}(p) = \{I_h(p), I_s(p), I_v(p)\}$ for a pixel p, we keep each of its elements fixed one at a time and derive three argand planes from the HSV color model. The locations of the color on these planes provide us the required complex-valued color representation $\{I_h^*(p), I_s^*(p), I_v^*(p)\}$. Note that (p) has been omitted in the symbols present in the figure for clarity. *Best viewed in color.*

resulted in real-valued propagation. To preserve complex-valued propagation in fully connected layers, we draw inspiration from [131], which showed that fully connected layers could be implemented as 1×1 convolutions. Therefore, we decided to adopt this approach and continue with complex-valued convolution operations until the end. Since complex-valued convolution operations have already been defined in [151], we simply have to continue with them until the end.

3.2 Proposed Method

We propose FCCN, a complex-valued fully convolutional network that ensures complex-valued information flow throughout the network. For generating complex-valued inputs, we propose a novel complex color model, iHSV. We also propose a complex-valued loss function to tackle the deficit between complex-valued outputs and labels. In addition, we implement fully connected layers as convolution layers, thus preserving complex-valued information flow.

Recall that hue (H), saturation (S), and value (V) represent the cylindrical HSV color model's angular, radial, and vertical axes. A point in this cylinder represents a color denoted by a set of three elements: H, S, and V values of that point. Let $I_{rgb} = \{I_r, I_g, I_b\}$ denote an RGB image, and $I_{hsv} = \{I_h, I_s, I_v\}$ its HSV equivalent. The location of HSV color of a pixel p in this cylinder can be denoted as

$\{I_h(p), I_s(p), I_v(p)\}$. This point in the cylinder is the intersection point of three color planes: $H = I_h(p)$, $S = I_s(p)$ and $V = I_v(p)$.

Geometrically, the HSV space forms a standard cylindrical coordinate system where S acts as the radial distance, H acts as the azimuthal angle, and V acts as the vertical height. A fundamental mathematical property of cylindrical coordinates is that their basis vectors are mutually orthogonal at every point in space. As shown in Fig. 3.2, each 2D color plane is obtained by fixing one of the elements, allowing us to leverage this inherent 3D orthogonality to construct valid 2D Argand planes.

Let us understand how the above discussion helps us generate the complex-valued color channels that are fed to FCCNs as inputs, our FCCN architecture, and the complex-valued loss function we develop.

3.2.1 Fixed Hue Based Channel (I_h^*)

We slice a rectangular plane by fixing H as $I_h(p)$. While we have fixed the value of H, the other two axes S and V still lie on that plane. Because S corresponds to the radial vector and V corresponds to the vertical vector, they remain strictly orthogonal to each other, as shown in Fig. 3.2. Let us treat it as an Argand plane with V as a real axis and S as an imaginary axis. Upon doing that, we can represent any 2-D location on that plane as a complex number, i.e., (V, S) can be represented as $V + iS$. Therefore, we can derive the following complex number from the location of $I_{hsv}(p)$ on $H = I_h(p)$ color plane:

$$I_h^*(p) = I_v(p) + iI_s(p) \tag{3.1}$$

where we simply substitute the color values and denote the derived complex number with $I_h^*(p)$, since it's obtained by fixing $H = I_h(p)$. Thus, we now have the first element of the complex representation of color $I_{hsv}(p)$ in the form of $I_h^*(p)$.

3.2.2 Fixed Saturation Based Channel (I_s^*)

We can similarly slice out a hollow cylinder when we fix S to $I_s(p)$. While we have fixed the value of S, the other two axes, H and V, still play a role in locating a point on that hollow cylinder. We open up this hollow cylinder to form a rectangular plane, as shown in Fig. 3.2. On this plane lie two axes, $I_s(p)H$ and V. Note here that the axis H is scaled by the factor of $I_s(p)$ (the radius of the cylinder) to represent arc length instead of the angle. Mathematically, arc length follows the tangential basis vector, which is strictly orthogonal to the vertical vector corresponding to V. Because unrolling a cylinder into a flat plane is an isometric mapping, this 90-degree intersection is perfectly preserved. We can treat this plane also as an Argand plane with $I_s(p)H$ as the real axis and V as the imaginary axis. Upon doing that, again, we can represent any 2-D location on that plane as a complex, i.e., the location ($I_s(p)H$,V) leads to $I_s(p)H + iV$. Therefore, the

Procedure 1 iHSV Color Model

Input: An image I

`/* convert RGB image to HSV image */`

Pre-processing: $I_{rgb} \rightarrow I_{hsv}$

Process: iHSV Generation

`/* convert each pixel from HSV → iHSV */`

foreach $p \in I_{hsv}$ **do**

$I_h^*(p) = I_v(p) + iI_s(p)$

$I_s^*(p) = I_s(p)I_h(p) + iI_v(p)$

$I_v^*(p) = I_s(p)cos(I_h(p)) + iI_s(p)sin(I_h(p))$

end

$I_{hsv}^* = \{I_h^*, I_s^*, I_v^*\}$

return I_{hsv}^*

complex number derived from the location of $I_{hsv}(p)$ on the color plane $S = I_s(p)$ is as follows:

$$I_s^*(p) = I_s(p)I_h(p) + iI_v(p) \tag{3.2}$$

where we substitute the color values and denote the derived complex number with $I_s^*(p)$, since it's obtained by fixing $S = I_s(p)$. Thus, $I_s^*(p)$ becomes the second element we require for complex representation of color $I_{hsv}(p)$.

3.2.3 Fixed Value Based Channel (I_v^*)

As shown in Fig. 3.2, we can also slice out a circular plane when we fix V as $I_v(p)$. While we have fixed the value of V, the other two axes, H and S, can still be used to find the polar coordinates of a point on that circular plane. Given that, we can convert these polar coordinates into a Cartesian coordinate system with $Scos(H)$ and $Ssin(H)$ as the two axes. By definition of the Cartesian mapping, these two constructed axes are strictly orthogonal. Let us treat this plane also as an Argand plane with $Scos(H)$ as a real axis and $Ssin(H)$ as an imaginary axis. Upon doing that, $Scos(H)+iSsin(H)$ can now represent location $(Scos(H), Ssin(H))$. Therefore, the complex number derived from the location of $I_{hsv}(p)$ on the color plane $V = I_v(p)$ is as follows:

$$I_v^*(p) = I_s(p)cos(I_h(p)) + iI_s(p)sin(I_h(p)) \tag{3.3}$$

where we substitute the color values and denote the derived complex number with $I_v^*(p)$, since it's obtained by fixing $V = I_v(p)$. Thus, we now have the third element of our complex representation of color $I_{hsv}(p)$ as $I_v^*(p)$.

Table 3.1: RGB →HSV→ iHSV conversion of basic colors

Color	RGB	HSV	iHSV		
			I_h^*	I_s^*	I_v^*
Red	$(255, 0, 0)$	$(0.00, 1.00, 1.00)$	$1.00 + i1.00$	$0.00 + i1.00$	$1.00 + i0.00$
Green	$(0, 255, 0)$	$(2.09, 1.00, 1.00)$	$1.00 + i1.00$	$2.09 + i1.00$	$-0.49 + i0.86$
Blue	$(0, 0, 255)$	$(4.19, 1.00, 1.00)$	$1.00 + i1.00$	$4.19 + i1.00$	$-0.50 - i0.86$
Cyan	$(0, 255, 255)$	$(3.14, 1.00, 1.00)$	$1.00 + i1.00$	$3.14 + i1.00$	$-1.00 + i1.22$
Yellow	$(255, 255, 0)$	$(1.04, 1.00, 1.00)$	$1.00 + i1.00$	$1.04 + i1.00$	$0.50 + i0.86$
Magenta	$(255, 0, 255)$	$(5.23, 1.00, 1.00)$	$1.00 + i1.00$	$5.23 + i1.00$	$0.50 - i0.86$
Black	$(0, 0, 0)$	$(0.00, 0.00, 0.00)$	$0.00 + i0.00$	$0.00 + i0.00$	$0.00 + i0.00$
Gray	$(128, 128, 128)$	$(0.00, 0.00, 0.50)$	$0.50 + i0.00$	$0.00 + i0.50$	$0.00 + i0.00$
White	$(255, 255, 255)$	$(0.00, 0.00, 1.00)$	$1.00 + i0.00$	$0.00 + i1.00$	$0.00 + i0.00$

3.2.4 iHSV Color

Now that we have the three complex-valued elements of color, we can combine them to form the set required for the complex representation of color $I^*_{hsv}(p)$ as the following:

$$I^*_{hsv}(p) = \{I^*_h(p), I^*_s(p), I^*_v(p)\} \tag{3.4}$$

which we represent as $I^*_{hsv}(p)$. This way, any colored pixel can get a complex-valued color representation, as we do in Algorithm 1 to generate complex image I^*_{hsv} from an RGB image I. The constituents of I^*_{hsv} are obtained using Eqns.(3.1)-(3.3), for which we need to first convert $I_{rgb} \rightarrow I_{hsv}$ through the pre-processing step.

Table 3.1 presents the iHSV values of several well-known colors to give readers a sense of the types of complex values we obtain. Notably, when the saturation I_s is zero, all three complex channels depend only on I_v. This behavior is expected since the iHSV color model is based on the HSV model, where, in the case of zero saturation, we are left with only the axis of the cylinder. Therefore, the value remains the only variable in this scenario.

3.2.5 Complex-valued Inputs

iCNNs require a real image and an imaginary image as inputs. To obtain the images, we stack the real components of our color elements to form the real image and the imaginary components for the imaginary image. Hence, the resulting complex image I can be written as:

$$I = I_{re} + iI_{im} \tag{3.5}$$

where, the real component of an image is $I_{re} = \{\mathrm{Re}(I^*_h), \mathrm{Re}(I^*_s), \mathrm{Re}(I^*_v)\}$, and the imaginary component is $I_{im} = \{\mathrm{Im}(I^*_h), \mathrm{Im}(I^*_s), \mathrm{Im}(I^*_v)\}$. In Fig. 3.3, given an image, we pictorially illustrate how one can obtain these components to form the two inputs required for iCNNs. Fig 3.4 shows image visualization of both components of iHSV.

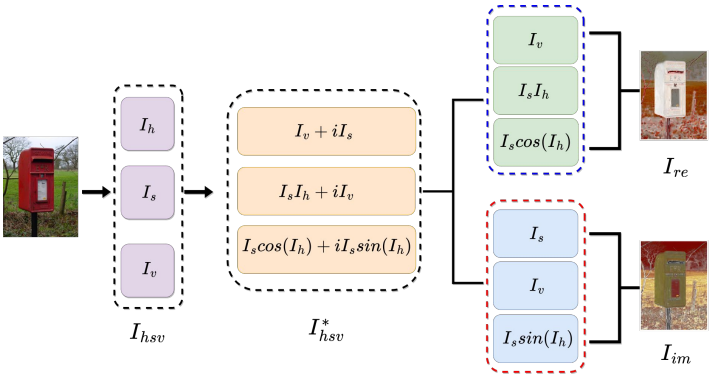


Figure 3.3: An illustration of how, using iHSV, an RGB image can be converted into complex-valued inputs required by an iCNN.

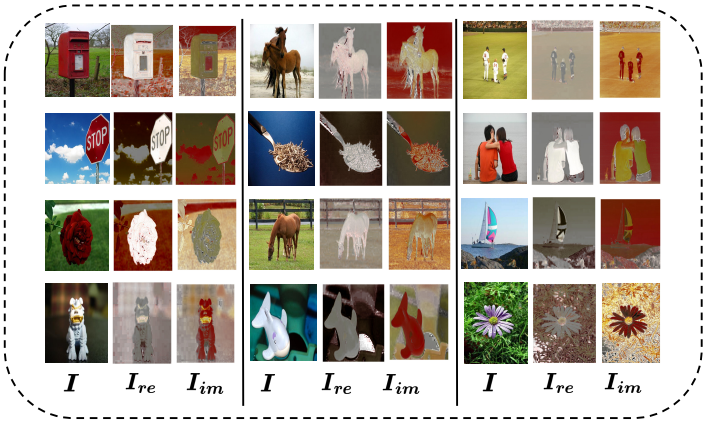


Figure 3.4: Real (I_{re}) and imaginary (I_{im}) components of sample images (I).

It is worth noting that the proposed conversion function, $I_{hsv} \rightarrow I^*_{hsv}$, is a one-to-one, invertible function. This can be proven by demonstrating that the original $I_{hsv}(p)$ values can be obtained from a given I^*_{hsv}. Specifically, we can obtain I_v as $\mathrm{Re}(I^*_h)$, I_s as $|I^*_v|$, and I_h as $\mathrm{Re}(I^*_s)/I_s$. Moreover, the resulting I_{hsv} can be further converted to the RGB color model using a standard conversion algorithm. Thus, we establish that our proposed conversion to the complex-valued color model is a one-to-one function.

3.2.6 FCCN Architecture

Generating probabilities and calculating losses in image classification tasks requires real values, which poses a challenge for complex-valued features. Current methods address this challenge by converting complex-valued features to real-valued ones using different strategies. Some methods, such as those based on DCN[151], treat the real and imaginary components of complex-valued features as two separate real-valued features.

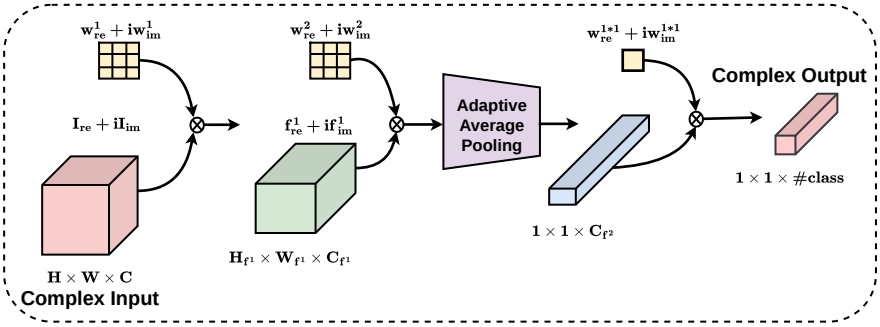


Figure 3.5: Blueprint of FCCNs: First, complex-valued convolutions are performed to obtain complex-valued feature maps. Then, adaptive average pooling is done to reduce the spatial dimensions to 1×1. At the end, 1×1 complex-valued convolutions are performed to reduce the channel-size to the number of classes.

Other methods, such as SurReal[12], employ manifold distance metrics. However, both of these strategies discard valuable phase information and undermine the end-to-end complex-valued information flow.

To preserve complex-valued properties, we propose using convolutional layers to replace fully-connected ones. In order to accomplish this, we use complex convolution layers to obtain features maps. Then, adaptive average pooling is done to reduce the spatial dimension of features to size 1×1 ($H \times W \times C \rightarrow 1 \times 1 \times C_f$, where C = #channels, H = height, W = width, and $C_f = H \times W \times C$). To emulate the operations in a fully connected layer, we can use 1×1 convolutions. This way, we can preserve the complex-valued flow and eventually reduce the number of channels to the number of classes ($1 \times 1 \times \#class$). We denote this 1×1 convolution part as f_{conv}. In Fig 3.5, we give a blueprint of this approach for better understanding of how FCCNs are fully complex-valued and convolutional in true sense.

3.2.7 Complex-valued Cross-entropy Loss ($\mathscr{L}_{comp}$)

Let's consider our FCCN as a classifier function $f_{comp} : \mathbb{C}^{H \times W \times 3} \rightarrow \mathbb{C}^c$, solving a c-class classification problem. Let α denote the set of parameters of the classifier, and let y_{jk} denote k^{th} element of one-hot encoded label of jth training sample. We cannot directly use the cross-entropy loss for complex-valued classifier f_{comp}, which has a complex-valued output. So, we modify cross entropy by decomposing $z_j = f_{comp}(x_j; \alpha)$, where z_j is the complex-valued vector output of f_{comp}. We now define our new complex loss function as $\mathscr{L}_{comp}$ and derive it as follows:

$$\begin{aligned}\mathscr{L}_{comp} &= -\frac{1}{n}\sum_{j=1}^{n}\sum_{k=1}^{c} y_{jk}\log z_{jk} \\ &= -\frac{1}{n}\sum_{j=1}^{n}\sum_{k=1}^{c} y_{jk}\log\left(|z_{jk}|e^{i\theta_{jk}}\right)\end{aligned} \tag{3.6}$$

where we express our z in polar form. This can now be rewritten as follows by considering $|z|$s as the required probabilities for the classification task.

$$\mathscr{L}_{comp} = \mathscr{L}_{CE} - i\frac{1}{n}\sum_{j=1}^{n}\sum_{k=1}^{c} y_{jk}\theta_{jk} \tag{3.7}$$

where $\mathscr{L}_{CE}$ is the usual real-valued cross-entropy loss, and the other term serves as imaginary loss.

It is worth noting that the magnitude of the complex-valued output vector z_j is constrained by $\Sigma_{k=1}^{c}|z_{jk}| = 1$ and $|z_{jk}| \geq 0\ \forall j,k$. To handle this constraint and ensure that the probability distribution of the output is valid, we first separate the magnitude and phase components of the complex vector. Next, we use softmax to create a probability distribution for the magnitude while leaving the phase information unchanged. Once we obtain the probability distribution in the form of magnitudes, we can apply the standard cross-entropy loss to the real part of $\mathscr{L}_{comp}$.

In complex domain, label vector y_j can be represented as $y_j e^{i0}$, suggesting that our loss should minimize the deviation between $y_j e^{i0}$ and z_j. It's clear that real part minimizes deviation between their magnitudes through $\mathscr{L}_{CE}$. On the other hand, when $y_{jk} = 1$, the imaginary part forces θ_{jk} to be 2π, which is equivalent to 0 because $e^{i(2\pi+\theta)} = e^{i\theta}$. That is when the minimum imaginary loss can occur, so $\mathscr{L}_{comp}$ also minimizes deficit between phases. Notably, our proposed loss function differs from traditional approaches that discard phase information in favor of simpler representations. Instead, by explicitly minimizing the imaginary loss, our model is able to effectively leverage phase information in the complex domain.

To establish the differentiability of this loss, we must ensure the differentiability of the function $\log z_{jk}$, which is the logarithm of a complex number.

Let $f(z) = \log(z)$, where $z = a + ib$. Note that if $f(z)$ can be expanded to a complex form $u(a,b) + iv(a,b)$, for it to be differentiable, it must satisfy Cauchy-Riemann equations[143], which are given as:

$$\frac{\partial u}{\partial a} = \frac{\partial v}{\partial b} \qquad \text{and} \qquad \frac{\partial v}{\partial a} = -\frac{\partial u}{\partial b} \tag{3.8}$$

Let's prove that logarithm of a complex number satisfies these equations.

Proof: Given function $f(z) = \log(z)$, where $z \in \mathbb{C}$, i.e., $z = a + ib$, we can compute the magnitude $|z|$ and phase θ of z as follows:

$$|z| = \sqrt{a^2 + b^2} \qquad \text{and} \qquad \theta = \tan^{-1}(b/a)$$

Since $z = |z|e^{i\theta}$, using above equations, we can expand $\log(z)$ into a complex form in the following manner:

$$\log(z) = (1/2)\log\Big(a^2 + b^2\Big) + i\tan^{-1}(b/a) \tag{3.9}$$

Thus, $u = (1/2)\log(a^2 + b^2)$ and $v = \tan^{-1}(b/a)$. Now, let's compute the four partial derivatives required in Cauchy-Riemann equations 3.8:

$$\frac{\partial u}{\partial a} = \frac{\partial}{\partial a}(1/2)\log\Big(a^2 + b^2\Big) = \frac{a}{a^2 + b^2} \tag{3.10}$$

$$\frac{\partial v}{\partial b} = \frac{\partial}{\partial b}\tan^{-1}(b/a) = \frac{a}{a^2 + b^2} \tag{3.11}$$

$$\frac{\partial v}{\partial a} = \frac{\partial}{\partial a}\tan^{-1}(b/a) = \frac{-b}{a^2 + b^2} \tag{3.12}$$

$$\frac{\partial u}{\partial b} = \frac{\partial}{\partial b}(1/2)\log\Big(a^2 + b^2\Big) = \frac{b}{a^2 + b^2} \tag{3.13}$$

From the above equations, we can conclude that the logarithm of a complex number indeed satisfies Cauchy-Riemann equations 3.8. Hence, $\log(z)$ is differentiable, and so is our proposed loss $\mathscr{L}_{comp}$.

Optimization: There are two types of gradients at the last layer: one for $|z|$s and another for θs. In our loss function, since the real part entirely relies on $|z|$s and the imaginary part entirely relies on θs, we assign the gradients of real part to $|z|$s and the gradients of imaginary part to θs.

3.3 Experiments

In this section, we present the evaluation of our proposed Fully Convolutional Complex-valued Networks (FCCNs) and compare their performance with other methods on both real-valued and complex-valued datasets.

3.3.1 Datasets & Model Architectures

We employ a total of five diverse image classification datasets to evaluate the performance of our proposed FCCNs: CIFAR-10 [77], CIFAR-100 [78], STL-10 [25], SVHN [107], and Tiny-ImageNet [81]. CIFAR-10 consists of ten categories, with each category having 5k training and 1k testing images. SVHN also has ten categories, but it contains a larger number of images: 73.2k images for training and 26k for testing in total. Similarly, STL-10 contains ten categories, with each category having 0.5k images

for training and 0.8k for testing. On the other hand, CIFAR-100 has 100 categories, with each category having 0.5k training and 0.1k testing images. Finally, Tiny-ImageNet, which is a subset of the Original ImageNet, contains 200 categories.

In our experiments, we adopt the Adam optimizer[71] with a learning rate of 10^{-3} and weight decay of 0.1 for training all models to 100 epochs, unless otherwise stated. We use $\mathbb{C}$ReLU activation and complex batch normalization in our architectures. To demonstrate the effectiveness of our proposed approach, we evaluate our method on both smaller and larger networks and compare our results with existing approaches. For smaller networks, we experiment with different versions of CIFARnet and provide them with various inputs, including Lab and sliding inputs generated by following the approach in [138]. Similarly, following [138], for larger networks, we use the CDS-large as the baseline for comparison.

We aim to demonstrate the scalability and applicability of our proposed method FCCN. Hence, we also conduct experiments on large-scale ImageNet[30] dataset and MSTAR[127] (SAR dataset). The ImageNet dataset comprises 1,281,167 training images and 50,000 validation images with 1000 classes, while the MSTAR dataset consists of naturally occurring complex images.

Table 3.2: Comparing classification accuracies of different approaches using CIFARnet [138] while varying the input strategy on five benchmark real-valued datasets. The values in **blue** are best, and the ones in **red** are second best. Our FCCN approach with iHSV inputs obtains the best results on every dataset. The second best results are achieved using our FCCN approach with sliding [138] input strategy on 3/5 datasets.

Method	STL-10					CIFAR-10					CIFAR-100					SVHN					Tiny-ImageNet				
	RGB	HSV	LAB	Sliding	iHSV	RGB	HSV	LAB	Sliding	iHSV	RGB	HSV	LAB	Sliding	iHSV	RGB	HSV	LAB	Sliding	iHSV	RGB	HSV	LAB	Sliding	iHSV
DCN[151]	31.80	31.25	32.20	32.42	33.81	65.17	64.37	58.64	63.83	66.71	32.52	31.46	27.36	28.87	32.94	85.26	85.31	84.43	87.44	87.59	19.19	19.07	18.49	18.30	19.89
SurReal[12]	-	-	-	-	-	50.68	-	53.02	54.61	-	23.57	-	25.97	26.66	-	80.51	-	53.48	80.79	-	-	-	-	-	-
CNN[76]	45.14	44.03	46.66	40.47	41.45	64.43	63.82	63.00	63.43	64.80	31.57	30.16	31.72	31.93	33.28	87.47	87.05	84.93	87.37	88.49	21.29	20.16	20.47	20.75	21.17
CDS[138]	41.23	40.32	46.54	47.38	45.60	69.23	66.74	67.58	69.19	68.47	41.83	38.31	39.52	42.08	39.24	89.39	87.64	88.86	90.25	89.41	25.60	21.76	24.35	25.51	25.63
FCCN	**59.60**	53.67	54.53	58.84	**60.61**	77.85	74.23	76.76	**78.01**	**78.23**	43.40	41.59	42.03	**44.19**	**47.34**	89.88	88.17	89.16	**90.81**	**90.95**	**31.11**	24.91	30.92	30.10	**33.76**

Table 3.3: Comparing classification accuracies(%) of different approaches using CDS-large[138], a large network architecture, on CIFAR-10 dataset. Our proposed FCCN outperforms all other approaches. The #Params is 1.7M for all the approaches.

Method	DCN[151]	CNN	CDS[138]	FCCN
Accuracy (%)	92.8	93.0	93.7	**94.3**

3.3.2 Results on Benchmark Datasets

We compare our proposed method with existing CIFARnet architecture approaches, as shown in Table 3.2. On CIFAR-10, CIFAR-100, STL-10, SVHN, and Tiny-ImageNet under different inputs, our FCCN obtains the highest accuracy than other methods (Table. 3.2). LAB input for networks is obtained as $\{L + i0, a + ib\}$, while sliding input

is created as $\{R + iG, G + iB\}$. FCCN shows highest improvement on STL-10 by an average margin of 13.17, with similar improvements on other datasets. We show highest improvement with our proposed iHSV input. We also compare FCCN with CDS-Large[138] architecture and obtain higher accuracy than all other methods, including real-valued CNN (Table. 3.3).

3.3.3 Results on ImageNet

We are the first to perform a large-scale experiment using complex-valued networks for color images. For training networks, we use four NVIDIA A100 GPUs in this experiment. All the models are trained for 100 epochs with a batch size of 256 using SGD optimizer, with 0.9 momentum. The input size is set as 224×224 for all images. Top-1 and Top-5 classification accuracies are reported in Table 3.4.

In this study, we evaluate the performance of our proposed FCCN on large standard architectures, including ResNet18, ResNet-50, and ResNet-152, and compare it with the real-valued CNN. Moreover, we compare the results of our method with other complex-valued methods on the CDS-large architecture, as reported in [138]. The experimental results demonstrate that our proposed FCCN consistently outperforms the real-valued CNN on large architectures, including ResNet-152. Furthermore, our method outperforms the other complex-valued methods as well.

Table 3.4: Comparing Top-1 and Top-5 classification accuracies of different approaches using multiple large network architectures on large Imagenet-1k dataset. FCCN performs best again regardless of the network architecture used.

	CDS-large			ResNet18		ResNet50		ResNet152	
	DCN	CDS	FCCN	CNN	FCCN	CNN	FCCN	CNN	FCCN
Top-1	49.76	53.25	**64.92**	72.64	**73.41**	75.67	**76.26**	76.62	**77.27**
Top-5	62.38	76.59	**85.24**	90.73	**91.95**	94.19	**94.70**	95.45	**95.84**

3.3.4 Results on Complex-valued data

The MSTAR (Moving and Stationary Target Acquisition and Recognition)[127] is a widely used benchmark dataset for evaluating SAR images. In this study, we follow the methodology proposed by [164] and perform experiments on a subset of the dataset consisting of 10 classes, each with images of size 128×128.

Our experimental results, summarized in Table 3.5, demonstrate that our approach outperforms the baselines. Furthermore, complex-valued methods generally perform better than their real-valued counterparts.

Table 3.5: Comparing classification accuracies of different approaches on MSTAR, a complex-valued benchmark dataset.

Method	CNN	SurReal[12]	DCN[151]	CDS[138]	FCCN
Accuracy (%)	66.9	94.9	89.1	96.1	**97.4**

3.3.5 Ablation Study

We conduct four ablation studies to highlight our contributions. Following [151]; we also conduct three ablation studies on CIFAR-10 dataset to evaluate our three contributions: complex-valued color model iHSV, convolutional implementation of fully-connected layers(f_{conv}) and complex-valued loss($\mathscr{L}_{comp}$). We perform a fourth ablation study on five datasets and eight different architectures to demonstrate the improvement over previous methods.

Effects of different channels: In our ablation study, we analyze the contribution of each complex-valued channel in I^*_{hsv} by inputting either only one or a pair of them to FCCN. Table 3.6 summarizes the results. It is observed that I^*_h and I^*_s perform better individually than I^*_v, but still require the assistance of the other complex-valued channels to achieve optimal performance. Moreover, different pair-wise combinations of channels peak with different architectures, and once again, the third channel's contribution is found to be crucial for the best performance. These results suggest that all three complex-valued channels play an essential role in improving the overall performance of FCCN.

Effects of I_{re} and I_{im}: We also examined the individual contribution of the real and imaginary components of the I^*_{hsv} image representation. We input only one component at a time to FCCN and evaluated their impact on the overall performance. The results, as summarized in Table 3.6, indicate that both the real and imaginary components make almost equal contributions.

Effects of $iHSV$, f_{conv} and $\mathscr{L}_{comp}$: We conduct an additional study where we add each component one by one to assess the significance of each contribution, (i) complex-valued color model ($iHSV$), (ii) 1×1 convolution (f_{conv}), and (iii) complex-valued loss function ($\mathscr{L}_{comp}$). When iHSV is absent, complex input has only RGB as real part. When f_{conv} is absent, we use the same strategy as DCN[151]. While in the absence of $\mathscr{L}_{comp}$, real-valued vectors are obtained from complex-valued via weighted average of both both components. The results in Table 3.7 demonstrate that removing any of the three proposed strategies results in performance degradation.

Overall Effect of our improvements over DCN[151]: In this study, we experiment on five different datasets. As evident from the results, we can conclude that even on standard architectures of a wide range of depth, FCCN is able to generalize better and improve upon DCN[151].

Table 3.6: Results of our ablation study that highlights the value of different parts of our iHSV color representation, which can be divided in two ways: channel-wise $\{I_h^*, I_s^*, I_v^*\}$ and component-wise $\{I_{re}, I_{im}\}$. We conduct this study using a wide-range of standard architectures on CIFAR-10 dataset.

I_h^*	I_s^*	I_v^*	I_{re}	I_{im}	AlexNet	VGG-16	VGG-19	ResNet-18	ResNet-50
✓					81.66	90.07	91.80	89.81	89.24
	✓				81.12	90.97	91.49	88.73	88.86
		✓			71.15	81.17	80.85	78.23	79.22
	✓	✓			86.75	88.58	91.16	91.41	88.39
✓		✓			85.08	90.21	91.36	90.91	88.45
✓	✓				82.79	90.06	91.03	90.18	90.53
			✓		79.33	91.64	91.81	90.45	91.84
				✓	79.38	91.36	91.31	91.54	91.69
✓	✓	✓	✓	✓	**89.36**	**92.71**	**93.25**	**93.35**	**93.58**

Table 3.7: Results of our ablation study that highlights the value of different components of our proposed method: iHSV, f_{conv} and $\mathscr{L}_{comp}$. We conduct this study using a wide-range of standard architectures on CIFAR-10 dataset.

iHSV	f_{conv}	$\mathscr{L}_{comp}$	AlexNet	VGG-16	VGG-19	ResNet-18	ResNet-50
			79.71	90.26	90.35	90.54	90.17
✓			80.54	90.61	90.29	90.01	90.87
✓	✓		82.33	91.54	91.05	91.17	91.79
✓	✓	✓	**89.36**	**92.71**	**93.25**	**93.35**	**93.58**

3.3.6 Model performance Analysis

Scalability: Our experimental evaluation is organized into three parts, focusing on small models with small datasets, large models with small datasets, and large models with large datasets. As illustrated in Table 3.2, we show that our proposed approach achieves a significant improvement for small-scale datasets, such as STL-10, where the number of training examples is restricted. Furthermore, our findings reveal that larger models also perform better in such scenarios (Table 3.8). Overall, our results showcase the effectiveness of FCCN in situations where training examples are limited, irrespective of the model's size.

Faster Convergence and Improved Accuracy: We also plot the accuracy curves of CNN, DCN, and FCCN approaches in Fig. 3.6 while keeping the learning rate constant (0.1) up to 30 epochs. We use the ResNet50 architecture and the ImageNet-1K dataset for this experiment. It is clear that our FCCN approach not only exhibits a significantly faster learning capability compared to the other two baselines, but also achieves a higher

Table 3.8: All our improvements are over DCN [151], our baseline method. Here, we study the overall affect of our improvements using a wide-range of standard architectures on different benchmark datasets.

Dataset	Method	AlexNet	VGG-16	VGG-19	ResNet-18	ResNet-50
CIFAR-10	DCN[151]	80.55	91.74	92.26	86.89	88.61
	FCCN	**89.36**	**92.71**	**93.25**	**93.35**	**93.58**
STL-10	DCN[151]	66.97	72.84	70.49	65.31	66.32
	FCCN	**71.03**	**76.42**	**76.62**	**76.48**	**76.22**
SVHN	DCN[151]	92.16	95.51	95.34	94.58	95.41
	FCCN	**92.54**	**94.76**	**95.12**	**96.31**	**96.38**
CIFAR-100	DCN[151]	58.14	67.18	66.13	68.45	68.63
	FCCN	**65.17**	**69.31**	**71.77**	**73.63**	**74.96**
Tiny-ImageNet	DCN[151]	46.09	50.82	52.33	54.26	55.28
	FCCN	**48.93**	**55.38**	**55.57**	**59.84**	**60.70**

overall accuracy ceiling.

Both the accelerated convergence and the superior final accuracy can be attributed to three key mathematical and architectural advantages inherent to our fully complex formulation:

1. **Decoupled Input Representation (I^*_{hsv}):** By processing visual data through our orthogonal iHSV color model, the input I^*_{hsv} provides perceptually decoupled features from the very first layer. Unlike raw RGB, where color and illumination are heavily entangled, the orthogonal Argand planes of I^*_{hsv} prevent distinct color channels from interfering with one another. This allows the network to bypass the epochs typically wasted trying to disentangle raw inputs, and provides a geometrically stable foundation that yields highly discriminative, accurate feature maps.

2. **Richer Gradient Signals:** Our complex cross-entropy loss ($\mathscr{L}_{comp}$) provides a more directional and informative gradient during backpropagation. While standard CNNs update weights based solely on scalar magnitude errors, FCCN simultaneously optimizes both magnitude (class probability) and phase (structural alignment). Penalizing these two decoupled geometric properties at once steers the network toward optimal weights more directly and prevents the model from stalling in suboptimal local minima, thereby raising the final accuracy.

3. **Constrained Optimization Space:** Complex arithmetic intrinsically ties the real and imaginary feature maps together. Because a single complex weight update simultaneously performs scaling and rotation, it acts as a natural structural regularizer. This restricts the parameter search space compared to unconstrained

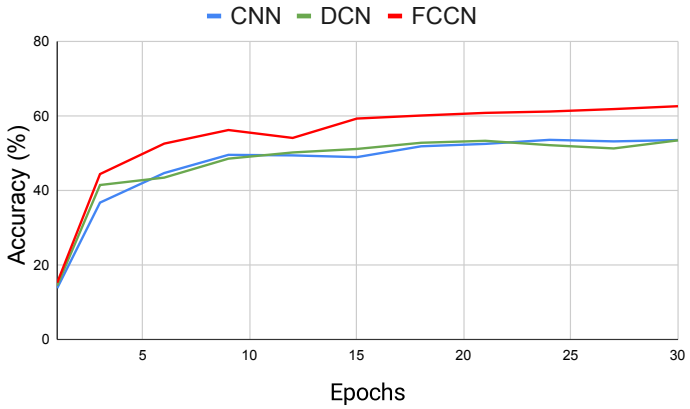


Figure 3.6: Accuracy curves obtained using DCN, CNN, and FCCN approaches, revealing the faster learning capability of FCCN.

real-valued networks, heavily mitigating the risk of overfitting on the training set and ensuring that the FCCN generalizes with higher accuracy to unseen test data.

Underlying all three factors is the richer representational capacity of the complex domain itself. A real-valued unit stores a single scalar, whereas a complex-valued unit stores a magnitude and a phase, encoding not just how strongly a feature is present but also its geometric orientation. Because a single complex weight performs a coupled scaling-and-rotation, each FCCN unit can express relationships (such as the relative alignment between color and structure) that a real-valued unit of the same width cannot. This higher information density per parameter is what lets FCCN reach a good solution in fewer epochs and at a higher final accuracy.

3.3.7 Addition Observations

Table 3.9: Transfer learning experiment on Modern Office-31 dataset

Modern Office-31	DCN[152]	CNN[76]	FCCN
Accuracy	93.71	96.61	**97.23**

For observing the generalization capability of our method, we show results on Fine-tuned ResNet152, which was pre-trained on ImageNet on webcam subset of Modern Office-31 dataset. We show our results for different methods in Table 3.9.

3.3.8 Applicability to Dense Prediction Tasks

Although FCCN is introduced for image classification, it is already *fully convolutional*: every fully connected layer is replaced by complex-valued 1×1 convolutions, so spatial feature maps are preserved throughout and never collapsed. It can therefore be extended to dense tasks in exactly the way real-valued CNNs are. Segmentation follows the CNN→FCN route: the classification head is replaced by a pixel-wise complex output and $\mathscr{L}_{comp}$ is applied per pixel, with the magnitude giving the class probability (e.g., foreground vs. background) and the phase acting as a structural binding signal that enforces boundary coherence. Detection follows the CNN→YOLO route: the same convolutional backbone regresses complex-valued box and objectness maps over a spatial grid. Neither extension alters the complex-valued core, which is what makes the FCCN blueprint a foundation for the dense architectures explored in the subsequent chapters of this thesis.

3.4 Summary

In this chapter, we have presented fully complex-valued convolutional network (FCCN), which operate entirely in the complex domain. It takes complex-valued image inputs using our novel complex-valued color model, iHSV, and is optimized using our novel complex-valued loss function. By maintaining such end-to-end consistency in complex-valued nature, our proposed approach outperforms previous works on complex-valued networks with significant margins on numerous datasets, including ImageNet-1k.

CHAPTER 4

DCS^{2}Net: Deep Complex Spatio-Spectral Networks

Deep learning has revolutionized computer vision, yet the exploration of alternative numerical representations, such as complex numbers, remains a compelling frontier. Complex-valued neural networks (iCNNs) [138, 151] offer the potential for richer feature representations by leveraging both magnitude and phase, a capability largely untapped in mainstream vision tasks typically reliant on real-valued inputs and processing. Despite promising theoretical advantages and success in domains with inherently complex data [26, 40, 61, 118], their application to core computer vision problems using standard real-valued (e.g., RGB) images is nascent and faces fundamental challenges [12, 138, 173].

Binary segmentation, a canonical task requiring clear understanding of global and local image context [126, 168], serves as an ideal testbed for investigating the end-to-end capabilities of iCNNs on visual data. A central motivation is to explore whether transforming real-valued images into the complex domain and processing them with appropriately designed iCNNs can unlock more expressive features and lead to improved performance. This requires addressing three critical research questions: *(i) How can real-valued input like RGB images be efficiently and meaningfully encoded into the complex domain? (ii) What novel iCNN architectures are required to effectively process these complex representations while preserving their unique properties? (iii) How can task objectives, such as segmentation masks, be represented in a complex-valued format to enable a truly end-to-end complex learning pipeline?*

Existing attempts to represent RGB images in the complex domain, such as the iHSV model ([V+iS, SH+iV, S$\exp(i\text{H})$]) [173] or [R+iG, G+iB] encoding [138], often introduce redundancy or significantly increase channel dimensionality, posing efficiency concerns. We argue that a more principled and compact real-to-complex transformation is crucial. Furthermore, while iCNNs have shown promise in terms of parameter efficiency and generalization [4, 44, 57], adapting recent architectural advancements to the complex domain for vision tasks remains an open and exciting research avenue. Successfully navigating these challenges could pave the way for harnessing the full potential of complex-valued representations in computer vision, moving beyond domains

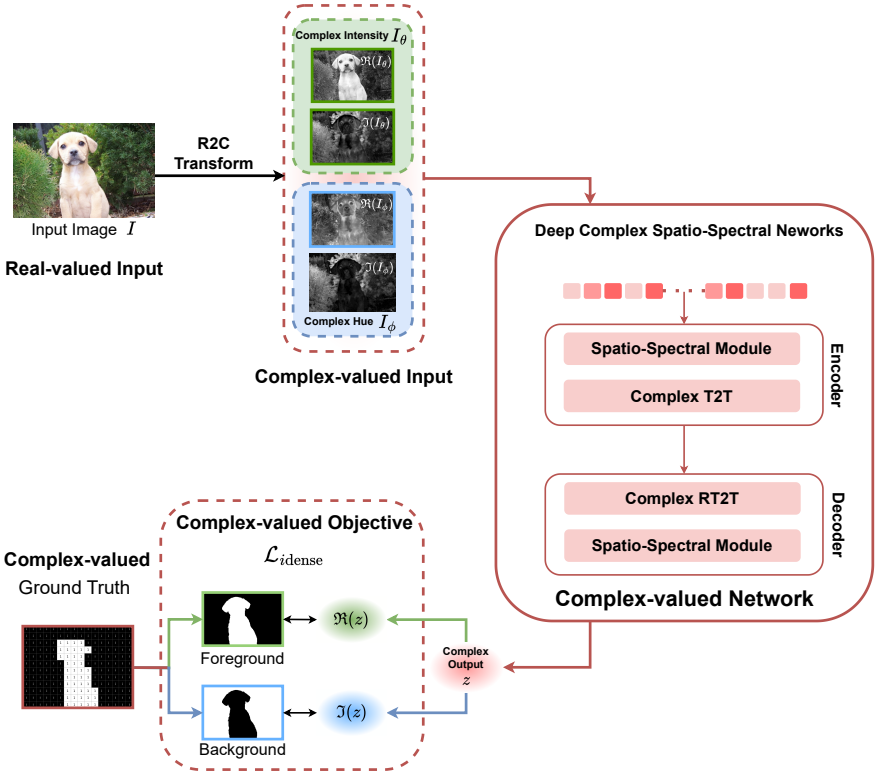


Figure 4.1: We introduce a novel token-based Deep Complex Spatio-Spectral Networks (DCS^{2}Net), which leverage (i) complex R2C transform for producing complex-valued inputs, (ii) Spatio-spectral module for capturing global context using complex Fourier transform, (iii) Complex T2T for progressively reducing tokens and its inverse Complex RT2T, and (iv) our novel complex-valued objective to optimize complex-valued output for binary segmentation.

where inputs are naturally complex.

In this work, we address these challenges by first proposing **R2C (real-to-complex)**, a novel and efficient color transform. R2C transforms each RGB pixel vector $(I_r, I_g, I_b)^T$ into two complex values, $\{Z_I, Z_C\}$. Each serves their own purpose. Z_I is obtained by projecting RGB vector onto a grayscale line($\vec{d}_{achro}$), trying to capture the achromatic property (intensity) of the color. While Z_C is obtained in the plane of form $x' + y' + z' = D$, which has $\vec{d}_{achro}$ as normal vector. Z_C in this plane represents the chromatic property of the color. This provides a compact, two-channel complex representation from three-channel RGB data which is also invertible. Moreover, this also provides an intuitive understanding of how each channel capture different property efficiently.

Building on this, we introduce **DCS^{2}Net (Deep Complex Spatio-Spectral Network)**, the first token-based iCNN designed for vision tasks. DCS^{2}Net leverages a novel spatio-spectral module as its core building block. This module transforms complex-valued tokens to the frequency domain, applies learnable complex-valued Fourier filters, and maps them back to the spatial domain, enabling the network to capture and integrate information across both domains while maintaining the integrity of complex-valued features throughout the architecture.

Finally, to facilitate end-to-end complex-valued learning for binary segmentation, we propose a novel objective minimization strategy using our proposed $\mathscr{L}_{i\text{dense}}$ loss. This loss directly compares the complex-valued output of DCS^{2}Net with a complex-encoded binary ground truth, where foreground and background are represented by distinct complex numbers (e.g., 1 and i). Since foreground and background are complementary to each other. This design is inspired by the inherent orthogonality of these states, aligning well with the properties of complex numbers. Our comprehensive approach is visually summarized in Fig. 4.1.

Our primary contributions are: ❶ **Efficient Complex Input:** A novel R2C transform to convert real-valued RGB images into a compact complex-valued representation. ❷ **Novel Complex Network Architecture:** DCS^{2}Net, the first token-based iCNN incorporating spatio-spectral modules for effective complex information processing. ❸ **Principled Complex Objective:** A new loss minimization strategy while comparing complex-valued outputs with complex encodings of binary masks. ❹ **Complex-valued Performance:** Extensive empirical validation showing DCS^{2}Net significantly outperforms existing complex-valued methods in various binary segmentation tasks. Rather than solely pursuing state-of-the-art benchmarks, this work prioritizes pushing the field of complex-valued computer vision a significant step forward through foundational methodological advancements.

4.1 Related Work

Complex-valued Deep Learning: The integration of complex numbers as weights in deep learning introduces novel possibilities for delving into two-dimensional spectra, as highlighted in previous works [56, 157]. Moreover, the significance of phase information in the firing rate of neurons is underscored by studies such as [125] and [65], emphasizing the potential advantages of employing complex-valued representations in neural networks. Specifically, the observed behavior of synchronized neurons with similar phases firing together, contrasted with asynchronous neurons with differing phases causing interference, bears a closer resemblance to the dynamics of biological neurons. This synchronization of inputs through neurons draws parallels to the gating mechanism found in both deep feedforward and recurrent neural networks [70, 141, 158]. Additionally, the advantages of complex parameters in recent state-space models have been proven in recent studies [118].

Recent studies have showcased the superior generalization capacity of complex-valued networks, as evidenced by prior research [57, 68, 138]. Notably, complex-valued autoencoders have outperformed slot-attention in the domain of object-centric learning [99, 142]. Moreover, the application of complex-valued networks has proven beneficial in diverse areas, including saliency prediction [65, 66] and iris recognition [108]. The findings presented in [22] further support the notion that employing a complex-valued vector can enhance the learning process for addressing multiple tasks.

Real to Complex conversions: Prior works have primarily explored transformations such as the Fourier transform, Hilbert transform, and quaternion representations to model color and spatial information. While the Fourier transform is a powerful tool for frequency domain analysis, its global nature makes it less effective in capturing localized color and spatial variations, which are crucial for image processing tasks. Others, such as the Hilbert transform, only focus on one color channel at a time. On the other hand, quaternion representation uses a 4D hypercomplex number $[0 + iR + jG + kB]$ without considering the inherent color space geometry[44, 193]. Recent advancements, such as iHSV [173], have proposed complex-valued models for color representation, yet they fall short of capturing the nuanced relationships between luminance and chrominance. Unlike such methods, the R2C method offers a unique advantage by providing an intuitive, dual-complex, efficient representation of color that better captures the geometry of the RGB color space while leading to broader application domains and applications in various image processing algorithms. This contribution significantly advances the ability to generate complex-valued images from real-valued ones.

4.2 Method

4.2.1 R2C: Complex-valued color transform

We define a transformation $T : \mathbb{R}^3 \to \mathbb{C}^2$ that maps an RGB pixel vector $\vec{F} = (I_r, I_g, I_b)^T$ to two complex values, $Z_I(\vec{F})$ (Intensity Channel) and $Z_C(\vec{F})$ (Chromatic Channel). We assume $1 \geq I_r, I_g, I_b \geq 0$.

Preliminary Definitions and Notations:
Let $\vec{F} = (I_r, I_g, I_b)^T$ be the vector for pixel p, $\vec{C} = (0, 0, 0)^T$ be the origin of the RGB space, and $\vec{d}_{\text{achro}} = (1, 1, 1)^T$ be the direction vector of the achromatic axis (grayscale line), please refer to Figure 4.2.

1. **Projection onto Achromatic Axis ($\vec{E}$):** The vector projection of $\vec{F}$ onto the achromatic axis ($\vec{d}_{\text{achro}}$) is $\vec{E} = \text{proj}_{\vec{d}_{\text{achro}}} \vec{F} = \frac{\vec{F} \cdot \vec{d}_{\text{achro}}}{\|\vec{d}_{\text{achro}}\|} \frac{\vec{d}_{\text{achro}}}{\|\vec{d}_{\text{achro}}\|} = \frac{I_r + I_g + I_b}{3}(1, 1, 1)^T$. Let $I_{\text{avg}} = \frac{I_r + I_g + I_b}{3}$. Then $\vec{E} = (I_{\text{avg}}, I_{\text{avg}}, I_{\text{avg}})^T$, and Its norm is $\|\vec{E}\| = \sqrt{3}|I_{\text{avg}}|$. Since $I_r, I_g, I_b \geq 0$, $I_{\text{avg}} \geq 0$, so $\|\vec{E}\| = \sqrt{3} I_{\text{avg}}$.

2. **Vector $\vec{EF}$ (Chromatic Component Vector):** $\vec{EF} = \vec{F} - \vec{E} = (I_r - I_{\text{avg}}, I_g - I_{\text{avg}}, I_b - I_{\text{avg}})^T$. Note that $\vec{E} \cdot \vec{EF} = \vec{E} \cdot (\vec{F} - \vec{E}) = \vec{E} \cdot \vec{F} - \vec{E} \cdot \vec{E}$. Since $\vec{E}$ is the projection of $\vec{F}$ onto the span of $\vec{d}_{\text{achro}}$, $(\vec{F} - \vec{E})$ is orthogonal to $\vec{E}$, so $\vec{E} \cdot (\vec{F} - \vec{E}) = 0$. Thus $\vec{E}$ and $\vec{EF}$ are orthogonal.

4.2.2 Intensity Channel (Z_I)

Definition 4.1 (**Intensity Channel** $Z_I(\vec{F})$)**.** Let $\theta = \angle ECF$ be the angle between vector $\vec{v} = \vec{F}$ (from Fig. 4.2) and its projection $\vec{E}$. The Intensity Channel $Z_I(\vec{F})$ is defined as:

$$Z_I(\vec{F}) = \|\vec{F}\| \cos\theta + i\|\vec{F}\| \sin\theta = \|\vec{F}\| e^{i\theta} \tag{4.1}$$

From the geometry of vector projection, $\vec{E}$ is the component of $\vec{F}$ along $\vec{d}_{\text{achro}}$, so $\|\vec{F}\| \cos\theta = \|\vec{E}\|$. The component of $\vec{F}$ orthogonal to $\vec{E}$ (and lying in the plane defined by $\vec{E}$, $\vec{F}$) is $\vec{EF}$. Thus, its magnitude is $\|\vec{F}\| \sin\theta = \|\vec{EF}\|$.
Substituting these into the definition of $Z_I(\vec{F})$: $Z_I(\vec{F}) = \|\vec{E}\| + i\|\vec{EF}\|$.

Let $L_F = \|\vec{E}\|$ and $M_F = \|\vec{EF}\|$ for simplification. Since norms are non-negative, $L_F \geq 0$ and $M_F \geq 0$. The magnitude is $|Z_I(\vec{F})| = \sqrt{L_F^2 + M_F^2} = \sqrt{\|\vec{E}\|^2 + \|\vec{EF}\|^2} = \sqrt{\|\vec{F}\|^2} = \|\vec{F}\|$. The argument is $\arg(M_F, L_F) = tan^{-1} \frac{\|\vec{EF}\|}{\|\vec{E}\|}$, which is θ.

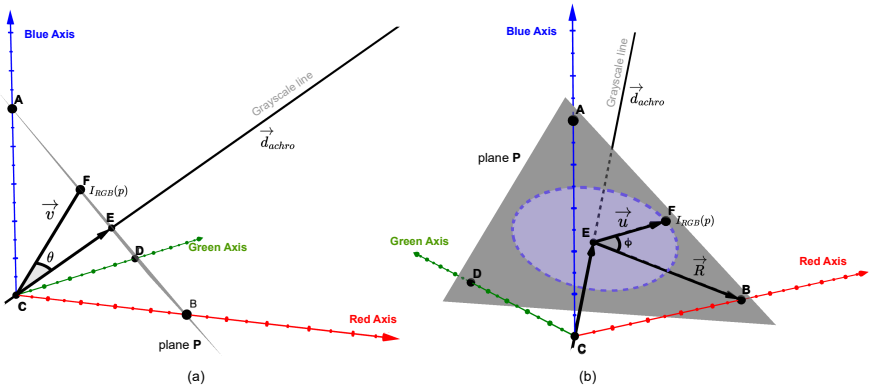


Figure 4.2: Proposed R2C color transformation: Given an RGB color $I_{RGB}(p) = \{I_r(p), I_g(p), I_b(p)\}$ for a pixel p. We have vectors $\vec{v} = \vec{CF}$ & $\vec{u} = \vec{EF}$. (a) Using θ, we project $\vec{v}$ onto $\vec{d}_{achro}$, which gives us the real and imaginary components of the first complex number. (b) Using ϕ between the reference vector $\vec{R}$ and vector $\vec{u}$. In the argand plane $\mathbf{P}$, we locate p, giving us the second complex number.

4.2.3 Chromatic Channel (Z_C)

The construction of the chromatic channel relies on mapping three-dimensional RGB space into an orthogonal, two-dimensional complex representation. This design is grounded in three core geometric justifications:

1. **Orthogonality to the Achromatic Axis:** In standard RGB space, the diagonal vector $\vec{d}_{\text{achro}}$ (where $R = G = B$) represents pure intensity (grayscale). Any movement along this diagonal changes pixel brightness without altering its color identity. By constructing the chromatic plane $\mathbf{P}$ to be strictly orthogonal to $\vec{d}_{\text{achro}}$, we geometrically guarantee that every point on $\mathbf{P}$ shares the exact same intensity ($3I_{\text{avg}}$). This ensures complete independence between brightness and color, preventing luminance changes from corrupting chromaticity features.

2. **Red-Axis Baseline Anchoring:** To measure phase angles consistently across different images, the coordinate system requires a fixed, universal starting line ($\phi = 0$). We define our reference vector $\vec{R}$ at the intersection of plane $\mathbf{P}$ with the red color axis. This choice aligns with traditional colorimetry (where red serves as the 0-degree baseline in HSV and HSL color wheels) and provides an intuitive orientation where phase shifts correspond directly to standard perceptual color transitions.

3. **Argand Plane Isomorphism:** Although plane $\mathbf{P}$ is sliced from a 3D RGB cube, it is mathematically constrained to a flat, two-dimensional subspace defined by the linear equation $x'+y'+z' = 3I_{\text{avg}}$. Because all chromatic vectors share a common origin E on this flat surface, any point on $\mathbf{P}$ can be uniquely and isometrically mapped using 2D polar coordinates: its radial distance from the center ($\|\vec{EF}\|$)

representing color saturation, and its angular rotation (ϕ) representing hue. This 1:1 spatial correspondence justifies treating plane $\mathbf{P}$ directly as a complex Argand plane without any loss of geometric fidelity.

Definition 4.2 (**Chromatic Channel** $Z_C(\vec{F})$)**.** Let $\vec{u} = \vec{EF}$ (from Fig 4.2) be the chromatic component vector and $M_F = \|\vec{u}\|$. Depending on the saturation of the pixel, we evaluate two cases:
(i) Grayscale Pixels ($M_F = 0$)**:** If the pixel lies directly on the achromatic diagonal, it possesses no color saturation. Hence, we define its chromatic channel as the complex origin: $Z_C(\vec{F}) = 0 + i0$.
(ii) Chromatic Pixels ($M_F > 0$)**:** The plane $\mathbf{P}$ is defined as the unique plane containing $\vec{u}$, centered at origin point E, and strictly orthogonal to the achromatic diagonal $\vec{d}_{\text{achro}}$. Its Cartesian plane equation for any point (x', y', z') is given by $x' + y' + z' = 3I_{\text{avg}}$.

A reference vector $\vec{R} = \vec{EB}$ (from Fig. 4.2) is established within plane $\mathbf{P}$, where point B marks the exact intersection of $\mathbf{P}$ with the RGB red axis. If $\phi = \angle FEB$ represents the true rotational phase angle from this reference vector, the Chromatic Channel $Z_C(\vec{F})$ is defined as:

$$Z_C(\vec{F}) = \|\vec{EF}\|e^{i\phi} = \|\vec{EF}\|(\cos\phi + i\sin\phi) \tag{4.2}$$

To compute this representation programmatically, let ϕ' be the shortest geometric angle between the chromatic vector $\vec{u}$ and the reference vector $\vec{R}$, obtained via the standard vector dot product: $\cos\phi' = \frac{\vec{u}\cdot\vec{R}}{\|\vec{u}\|\|\vec{R}\|}$, where $\phi' \in [0, \pi]$.

Because the inverse cosine function is restricted to the upper half-circle $[0, \pi]$, relying solely on ϕ' creates a geometric ambiguity; it cannot differentiate between colors that mirror each other across the red reference axis (e.g., confusing blues with greens). To capture the true directional rotation around the full color wheel, we must expand the angle to the continuous range $\phi \in [0, 2\pi)$. By evaluating the relative dominance of the blue versus green pixel intensities, we resolve this reflection ambiguity and define the true phase angle as:

$$\phi(\vec{F}) = \begin{cases} \phi' & \text{if } I_b(\vec{F}) \geq I_g(\vec{F}) \\ 2\pi - \phi' & \text{if } I_b(\vec{F}) < I_g(\vec{F}) \end{cases} \tag{4.3}$$

By treating the 2D subspace $\mathbf{P}$ as an Argand plane, assigning saturation $\|\vec{EF}\|$ to magnitude and hue rotation ϕ to phase, we obtain a geometrically consistent, mathematically complete complex chromaticity representation:

$$Z_C(\vec{F}) = \|\vec{EF}\|e^{i\phi} = \|\vec{EF}\|(\cos\phi + i\sin\phi) \tag{4.4}$$

4.2.4 Invertibility of R2C Transform

Lemma 4.3 (Invertibility of T)**.** *The transformation $T : \mathbb{R}^3_{\geq 0} \to \mathbb{C}^2$ is injective. It is a bijection from its domain $\mathbb{R}^3_{\geq 0}$ to its range $R_T = T(\mathbb{R}^3_{\geq 0}) \subset \mathbb{C}^2$, where $R_T =$*

$\{(L + iM, Me^{i\phi}) \mid L, M \geq 0, \phi \in [0, 2\pi)$ *such that the inverse map yields* $\vec{F} \in \mathbb{R}^3_{\geq 0}\}$. *Thus* T *is invertible from* $\mathbb{R}^3_{\geq 0}$ *to* R_T.

Proof. Let the given complex values from the range R_T be $Z_I = L + iM'$ and $Z_C = X_C + iY_C$. The inverse transformation $T^{-1} : R_T \to \mathbb{R}^3_{\geq 0}$ aims to recover $\vec{F} = (I_r, I_g, I_b)^T$.

1. **Recover I_{avg} and M_F (chromatic magnitude):** From Definition 3.1, $L = \text{Re}(Z_I) = \|\vec{E}\|$ and $M' = \text{Im}(Z_I) = \|\vec{EF}\|$. So, $L_F = L$ and $M_F = M'$. Since L_F, M_F are derived from norms, any valid $(Z_I, Z_C) \in R_T$ must have $L \geq 0$ and $M' \geq 0$. From $L = \|\vec{E}\| = \sqrt{3}I_{\text{avg}}$ (since $I_{\text{avg}} \geq 0$), we get $I_{\text{avg}} = L/\sqrt{3}$.

2. **Consistency Check:** We must have $M_F = |Z_C|$. Thus, for (Z_I, Z_C) to be in R_T, it must satisfy $\text{Im}(Z_I) = |Z_C|$. If $\text{Im}(Z_I) \neq |Z_C|$, the input is not in R_T. We assume this consistency holds.

3. **Handle Grayscale Case ($M_F = 0$):** If $M_F = M' = 0$, then $\vec{EF} = \vec{0}$, meaning $\vec{F} = \vec{E}$. Thus $I_r = I_g = I_b = I_{\text{avg}} = L/\sqrt{3}$. In this case, Z_C must be $0 + i0$ (since $M_F = 0$ is its magnitude). So, if $Z_I = L + i0$ (i.e., $M' = 0$), then $I_r = I_g = I_b = L/\sqrt{3}$. Z_C must be 0.

4. **Handle Chromatic Case ($M_F > 0$):** If $M_F = M' > 0$: $I_{\text{avg}} = L/\sqrt{3}$ is known. From $Z_C = M_F(\cos\phi + i\sin\phi)$, we have:

$$\cos\phi = \text{Re}(Z_C)/M_F = X_C/M_F$$

$$\sin\phi = \text{Im}(Z_C)/M_F = Y_C/M_F$$

This uniquely determines $\phi \in [0, 2\pi)$ since $M_F > 0$. The vector $\vec{EF}$ is reconstructed using orthonormal basis vectors in the chromatic plane. $\vec{R} = \overrightarrow{EB} = B - E = (3I_{\text{avg}} - I_{\text{avg}}, 0 - I_{\text{avg}}, 0 - I_{\text{avg}})^T = (2I_{\text{avg}}, -I_{\text{avg}}, -I_{\text{avg}})^T$. Let $\mathbf{u}_{\vec{R}} = \frac{1}{\sqrt{6}}(2, -1, -1)^T$ (unit vector for $\vec{R}$, assuming $I_{\text{avg}} > 0$). Let $\mathbf{u}_{\vec{S'}} = \frac{1}{\sqrt{2}}(0, -1, 1)^T$ be the unit vector for the effective imaginary axis for ϕ, consistent with Eq. (4.3). Then $\vec{EF} = M_F(\cos\phi\,\mathbf{u}_{\vec{R}} + \sin\phi\,\mathbf{u}_{\vec{S'}})$. The components of $\vec{EF} = (I_r - I_{\text{avg}}, I_g - I_{\text{avg}}, I_b - I_{\text{avg}})$ are:

$$I_r - I_{\text{avg}} = M_F(\cos\phi\frac{2}{\sqrt{6}})$$

$$I_g - I_{\text{avg}} = M_F(-\frac{\cos\phi}{\sqrt{6}} - \frac{\sin\phi}{\sqrt{2}})$$

$$I_b - I_{\text{avg}} = M_F(-\frac{\cos\phi}{\sqrt{6}} + \frac{\sin\phi}{\sqrt{2}})$$

Reconstructing I_r, I_g, I_b:

$$I_r = I_{\text{avg}} + M_F\frac{2\cos\phi}{\sqrt{6}}$$

$$I_g = I_{\text{avg}} + M_F(-\frac{\cos\phi}{\sqrt{6}} - \frac{\sin\phi}{\sqrt{2}})$$
$$I_b = I_{\text{avg}} + M_F(-\frac{\cos\phi}{\sqrt{6}} + \frac{\sin\phi}{\sqrt{2}})$$

These equations provide a unique (I_r, I_g, I_b) for given I_{avg} (from L), M_F (from M'), and ϕ (from Z_C/M_F).

5. **Uniqueness and Injectivity Conclusion:** The forward transformation $\vec{F} \mapsto (Z_I(\vec{F}), Z_C(\vec{F}))$ involves steps that are well-defined. The inverse mapping $(Z_I, Z_C) \mapsto \vec{F}$ derived above is also well-defined for $(Z_I, Z_C) \in R_T$ and produces a unique $\vec{F} \in \mathbb{R}^3$. The conditions defining R_T ensure that $L \geq 0$, $M' \geq 0$, $\text{Im}(Z_I) = |Z_C|$, and that the reconstructed I_r, I_g, I_b are non-negative. This explicit construction of a unique inverse T^{-1} for every element in R_T demonstrates that T is a bijection from $\mathbb{R}^3_{\geq 0}$ to R_T. Hence, T is invertible on this domain and range. □

4.2.5 Simplified pseudo code

In order to further simplify the understanding, we also provide a simpler pseudo-code for inverse R2C (C2R) transformation in Algo. 2. To validate that the reconstructed image using inverse R2C transform is correct, we calculate the SSIM score for a few sample images and present them in Table 4.1.

Given the complex image I_{iRGB}, we convert each pixel value from complex to real. First, we find $u = \vec{EF}$ (Fig. 4.2) using its components along $\vec{R}$ and $\vec{R}^{\perp}$, then using the component of $\vec{v}$ along $\vec{d}_{chrom} = \vec{CE}$, we obtain $\vec{v} = \check{\text{C}} + \check{\text{E}}$ and hence the point F which is the location of pixel p in RGB space.

4.2.6 DCS^{2}Net: Deep Complex Spatio-Spectral Networks

Drawing inspiration from the success of transformer architectures [33, 180] while addressing the computational challenges faced by complex-valued transformers on image data [32, 34, 177], we propose Deep Complex Spatio-Spectral Networks (DCS^{2}Net). Our architecture leverages Fourier filters as an efficient alternative to self-attention, enabling the model to operate entirely within the complex domain and capture global context effectively.

As depicted in Figure 4.3 (architecture overview figure, reference assumed), DCS^{2}Net takes a complex-valued input of size $H \times W \times 2$ (assuming channels). It processes this input through an encoder-decoder structure. The encoder progressively reduces the spatial resolution by tokenizing patches and employing modules that operate in both spatial and frequency domains. The decoder mirrors this process, progressively increasing the spatial resolution to produce a dense complex-valued output.

The encoder is designed to learn robust complex-valued feature embeddings suitable for binary segmentation. Unlike conventional approaches that might utilize pre-trained

Procedure 2 Pseudo code for inverse R2C Color Transformation

Input: A complex-valued Image, I_{iRGB}

Process: Inverse R2C transform

I_{rgb} **foreach** $p \in I_{iRGB}$ **do**

$C_1 = \|\vec{v}\| \cos\theta + i\|\vec{v}\| \sin\theta$

$C_2 = \|\vec{u}\| \cos\phi + i\|\vec{u}\| \sin\phi$

point E = $(\mathrm{Re}(C_1), \mathrm{Re}(C_1), \mathrm{Re}(C_1))$

plane P = plane formed perpendicular to vector $\vec{d}_{chrom}$ at point E

point B = Intersection of Plane P and Red axis

vector $\vec{R} = \vec{EB}$

vector $\vec{R}^{\perp}$ = lies in plane P, and has angle $+\pi/2$ with $\vec{R}$

$u_1 = \mathrm{Re}(C_2)\hat{\vec{R}}$

$u_2 = \mathrm{Im}(C_2)\ \hat{\vec{R}^{\perp}}$

$\vec{u} = \vec{EF} = u_1 + u_2$

$\vec{CE} = \mathrm{Re}(C_1)\hat{d}_{chrom}$

$\vec{v} = \vec{CF} = \vec{CE} + \vec{EF} = [I_r(p), I_g(p), I_b(p)]$

point $F = (I_r(p), I_g(p), I_b(p))$

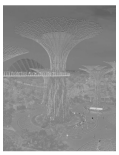
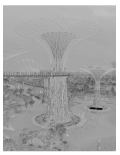
end

$I_{RGB} = \{I_r, I_g, I_b\}$

real-valued encoders, DCS^{2}Net is built from the ground up to handle complex-valued information end-to-end. The complex input image $I \in \mathbb{C}^{H\times W\times 2}$ is initially processed by complex convolutions to generate initial complex patch features, effectively creating tokens of a certain size, e.g., $\frac{H}{4} \times \frac{W}{4}$. The encoder then employs a sequence of two key complex-valued modules: the Complex Token-to-Token (T2T) module for spatial token reduction and local structure modeling, and the Spatio-Spectral Module (SSM) for capturing global and local context via frequency-domain filtering.

After obtaining the compact complex feature embeddings T^{ε} from the encoder, the decoder aims to progressively upsample these features to the original spatial resolution $H \times W$ to predict the dense salient mask output. The encoded tokens T^{ε} have a significantly reduced spatial length, e.g., corresponding to $\frac{H}{16} \times \frac{W}{16}$ resolution. The decoder utilizes the reverse operation of the Complex T2T module, termed Complex Reverse T2T (RT2T), to upsample the complex tokens. The Complex RT2T module effectively expands each complex token into multiple subtokens while preserving the continuity of information. To enrich the upsampling process and recover fine-grained details lost during encoding, the decoder leverages low-level complex tokens from the encoder via skip connections. Progressively, the embedded tokens T^{ϵ} are upsampled, and information from intermediate encoder token levels (e.g., T_1, T_2) is fused using concatenation followed by complex linear projection.

Table 4.1: In the following table, we present sample images, their corresponding complex channels obtained using our R2C transform, the reconstructed images from our inverse R2C transform, and the SSIM comparison scores between the original and reconstructed images.

Original Image	Intensity channel		Chromatic channel		Reconstructed Image	SSIM
	$\mathrm{Re}(I_\theta)$	$\mathrm{Im}(I_\theta)$	$\mathrm{Re}(I_\phi)$	$\mathrm{Im}(I_\phi)$		
						**1.0**
						**1.0**
						**1.0**
						**1.0**
						**1.0**
						**1.0**
						**1.0**

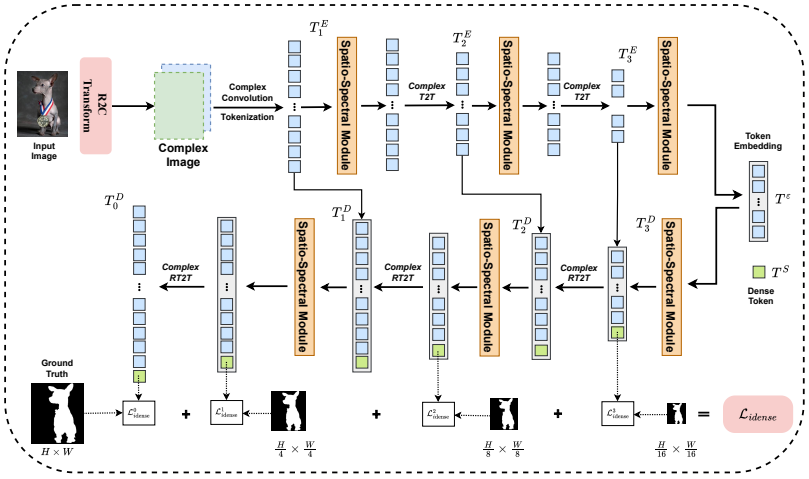


Figure 4.3: The overall architecture of our proposed DCS^{2}Net. It first encodes input image patch sequences to generate tokens to multiple resolutions (T_1^E, T_2^E, T_3^E), using Complex T2T. Then, we add a dense token (T^S) to the output (T^ε) of the encoder. Finally, the decoder progressively upsamples the tokens using Complex RT2T while predicting the saliency map at each step. We also optimize the generated map using our proposed loss to improve the predicted map.

Spatio-Spectral Module (SSM)

The Spatio-Spectral Module (SSM) is a core component that enables global receptive fields within the complex domain. It comprises a Fourier filtering operation followed by a complex Multi-Layer Perceptron (MLP). Given a complex input feature map $x \in \mathbb{C}^{H\times W\times D}$, where D is the feature dimension, we first transform it to the frequency domain using the 2D Discrete Fourier Transform (DFT) applied spatially: $X = \mathscr{F}[x] \in \mathbb{C}^{H\times W\times D}$ Here, $\mathscr{F}[\cdot]$ denotes the 2D complex DFT. The complex tensor X represents x in the frequency domain. A learnable complex filter $K \in \mathbb{C}^{H\times W\times D}$ is then applied through element-wise multiplication: $\hat{X} = K \odot X$ where $\odot$ signifies element-wise multiplication. Since K shares the dimensions of X, this operation acts as a global filter in the frequency domain. Finally, we return to the spatial domain using the inverse DFT: $x' = \mathscr{F}^{-1}[\hat{X}]$. This process is illustrated in Figure 4.4. The multiplication by a frequency-domain filter K is equivalent to a convolution with a spatial filter of size $H \times W$. Operating in the complex domain further allows us to preserve information regarding both magnitude and phase across different spatial frequencies.

While mathematically a frequency-domain filter corresponds to a global spatial convolution, the network can learn filters K that effectively concentrate energy in lower frequencies, corresponding to smoother or more localized structures in the spatial domain. We hypothesize that the network can implicitly learn to synthesize spatial filters with varying effective receptive fields, including those capable of extracting local features.

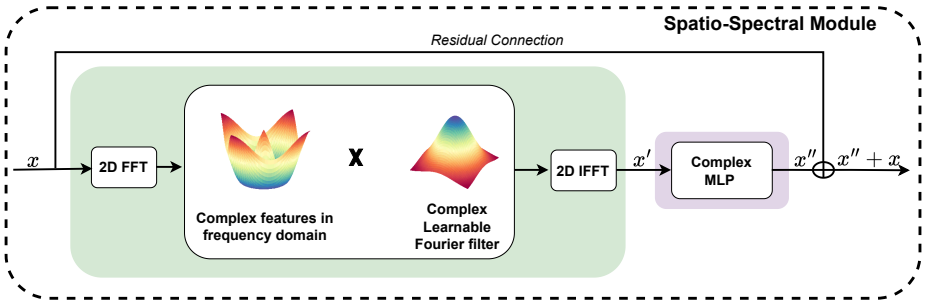


Figure 4.4: Spatio-Spectral Module takes a complex-valued input, applies a learnable complex filter in the frequency domain, and gives a complex-valued output.

Complex T2T and RT2T Modules

Inspired by the [180], we introduce the Complex T2T module to progressively reduce the sequence length of complex tokens while integrating local spatial context in the encoder. The Complex RT2T module performs the inverse operation in the decoder to upsample features, shown in Figure 4.5.

Complex T2T Module: Given tokens $T \in \mathbb{C}^{l \times c}$, it iteratively applies Structurization (reshape T to $I \in \mathbb{C}^{h \times w \times c}$, $l = hw$) and De-structurization (patching I into $k \times k$ patches with stride s and padding p, unfolding to $T' \in \mathbb{C}^{l' \times ck^2}$), reducing the sequence length to $l' = \lfloor \frac{h+2p-k}{k-s} + 1 \rfloor \times \lfloor \frac{w+2p-k}{k-s} + 1 \rfloor$.

Complex RT2T Module: The reverse process, Complex RT2T, upsamples the information. It first projects input tokens $T \in \mathbb{C}^{l \times d'}$ to a lower dimension $c = 64$ (e.g., from $d = 384$). Then, a complex linear projection expands the embedding dimension from c to ck^2. Similar to T2T de-structurization, each expanded token is treated as a $k \times k$ patch with s overlap. These patches are then folded into an image, potentially with p zero-padding, and reshaped to form the upsampled token sequence $T' \in \mathbb{C}^{l' \times c'}$ with increased length l'.

Token-based Mask Generation

Following the approach in several transformer-based architectures [33, 34, 93, 180], we incorporate a dense token to aid in generating the final dense prediction mask, as illustrated in Figure 4.3. We design a learnable complex-valued dense token $T^S \in \mathbb{C}^{1 \times d_{emb}}$, where d_{emb} is the embedding dimension. This token is prepended or added to the sequence of encoded patch tokens T_i^D at each level $i \in \{0, 1, 2, 3\}$ within the decoder pathway. By processing these modified token sequences through the subsequent SSM and Complex RT2T layers, the dense token T^S is designed to progressively accumulate global and local feature information necessary for dense prediction. The final stage of the decoder involves sending the upsampled tokens from an intermediate level (e.g., T_1^D) through an SSM and a final Complex RT2T module to produce the complex out-

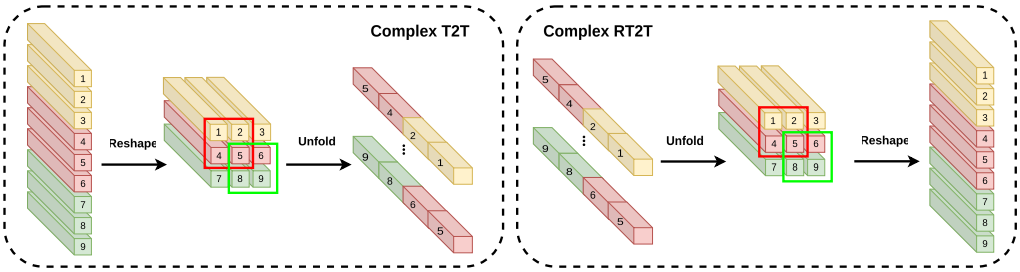


Figure 4.5: Complex T2T and Complex RT2T modules. T2T reduces token count ($l \rightarrow l'$) by restructuring tokens into a spatial map and then repatching. RT2T reverses this process.

put at full spatial resolution ($1/4$ to $1/1$ resolution). This final output is then used to generate the binary mask.

4.2.7 Objective function

Because our network produces a complex-valued output tensor, standard binary segmentation loss functions which operate exclusively on scalar real numbers cannot be applied directly. To resolve this without discarding complex information, we propose a novel loss minimization strategy centered on a specialized complex loss, $\mathscr{L}_{i\text{dense}}$.

Let z_j be the complex-valued output predicted by our network for the j-th input image. Given the standard real-valued ground truth binary mask $y_j \in \{0, 1\}$, we construct a complex-valued ground truth tensor by mapping foreground pixels to the real unit vector $(1 + 0i)$ and background pixels to the imaginary unit vector $(0 + 1i)$, as illustrated in Figure 4.6. Mathematically, this complex target can be compactly expressed as $y_j + i(1 - y_j)$.

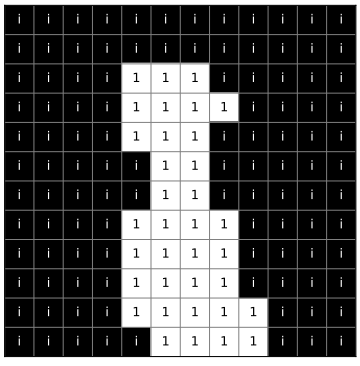


Figure 4.6: Complex ground truth formulation, where foreground pixels are actively mapped to 1 and background pixels to i.

In conventional real-valued segmentation, assigning foreground to 1 and background to 0 places both classes on a single scalar continuum. This treats the background as "absence of signal," which can lead to asymmetric optimization: multiplying by zero during loss backpropagation often passively suppresses gradient updates for background regions.

By contrast, encoding foreground as 1 and background as i is grounded in three distinct mathematical principles:

1. **Geometric Orthogonality:** On the Argand plane, the real axis (1) and imaginary axis (i) are strictly perpendicular (90° apart). This enforces the orthogonality principle, geometrically modeling foreground and background as mutually independent entities rather than opposing ends of a single scalar line.

2. **Symmetric Gradient Flow:** Because $|1| = |i| = 1$, both foreground and background regions possess a constant, non-zero unit magnitude. This ensures that background pixels actively contribute to gradient backpropagation with the exact same structural weight as foreground pixels, eliminating the passive suppression caused by zero-targets.

3. **Phase-Based Boundary Separation:** This formulation decouples class separation from magnitude optimization. While every pixel is driven toward a target magnitude of 1, classification is solved purely through phase rotation, optimizing foreground pixels toward an angle of 0 radians and background pixels toward $\pi/2$ radians (90°). This sharp angular distinction prevents structural collapse along ambiguous object boundaries.

To train the network using this orthogonal representation, our proposed loss $\mathscr{L}_{i\text{dense}}$ simultaneously minimizes the error along both axes by comparing the predicted real component $\mathrm{Re}(z_j)$ against the foreground mask y_j, and the imaginary component $\mathrm{Im}(z_j)$ against the background mask $(1 - y_j)$:

$$\mathscr{L}_{i\text{dense}}(z_j) = \sum_{j=1}^{N} \sum_{k=0}^{3} \Big(\mathscr{L}(y_j^k, \mathrm{Re}(z_j^k)) + \mathscr{L}((1 - y_j^k), \mathrm{Im}(z_j^k)) \Big) \tag{4.5}$$

where $\mathscr{L}$ represents the combined binary cross-entropy and IoU loss, k indicates the feature scale at decoder token levels T_k^D, and N is the total number of training images in the batch.

Once the network is trained, generating the final real-valued binary prediction mask from the complex output z_j is straightforward. Because foreground is driven to $\mathrm{Re}(z_j) \to 1$ and background is driven to $\mathrm{Im}(z_j) \to 1$ (meaning its complement $(1 - \mathrm{Im}(z_j)) \to 1$ for foreground), we fuse both orthogonal channels by taking their average: $mask = \frac{1}{2}(\,\mathrm{Re}(z_j) + (1 - \mathrm{Im}(z_j)))$. This dual-channel consensus provides a cleaner, more robust segmentation boundary than relying on a single scalar activation.

4.3 Experiments

In this section, we present the evaluation of our proposed Deep Complex Spatio-Spectral Networks (DCS^{2}Net) and compare their performance with other methods (both real and complex-valued) for multiple tasks.

Table 4.2: Quantitative comparison of our proposed DCS^{2}Net with other real-valued RGB SOD methods on five benchmark datasets. "-R" means the ResNet50. **■**: best, **■**: second best.

		DUTS[167]				ECSSD[175]				HKU-IS[84]				PASCAL-S[89]				DUT-O[176]				Average			
Method	Param(M)	$S_m\uparrow$	maxF↑	$E_\xi^{max}\uparrow$	MAE↓	$S_m\uparrow$	maxF↑	$E_\xi^{max}\uparrow$	MAE↓	$S_m\uparrow$	maxF↑	$E_\xi^{max}\uparrow$	MAE↓	$S_m\uparrow$	maxF↑	$E_\xi^{max}\uparrow$	MAE↓	$S_m\uparrow$	maxF↑	$E_\xi^{max}\uparrow$	MAE↓	$S_m\uparrow$	maxF↑	$E_\xi^{max}\uparrow$	MAE↓
Real-valued																									
PiCANet$_{2018}$[92]	47.22	0.863	0.840	0.915	0.040	0.916	0.929	0.953	0.035	0.905	0.913	0.951	0.031	0.846	0.824	0.882	0.072	0.826	0.767	0.865	0.054	0.871	0.854	0.913	0.046
BASNet$_{2019}$[115]	87.06	0.866	0.838	0.902	0.047	0.916	0.931	0.951	0.037	0.909	0.919	0.952	0.032	0.837	0.819	0.868	0.083	0.836	0.779	0.872	0.057	0.872	0.856	0.909	0.051
PoolNet$_{2019}$[91]	68.26	0.879	0.853	0.917	0.041	0.917	0.929	0.948	0.042	0.916	0.920	0.955	0.032	0.852	0.830	0.880	0.076	0.832	0.769	0.869	0.056	0.879	0.864	0.914	0.049
EGNet-R$_{2019}$[183]	111.64	0.887	0.866	0.926	0.039	0.925	0.936	0.955	0.037	0.0918	0.923	0.956	0.031	0.852	0.825	0.874	0.080	0.841	0.778	0.878	**0.053**	0.886	0.868	0.918	0.048
MINet-R$_{2020}$[114]	162.38	0.884	0.864	0.926	0.037	0.925	0.938	0.957	0.034	0.919	0.926	0.960	0.031	0.856	0.831	0.883	0.071	0.833	0.769	0.869	0.056	0.885	0.868	0.919	0.046
GateNet-R$_{2020}$[188]	128.63	0.891	0.874	0.932	0.038	0.924	0.935	0.955	0.038	0.921	0.926	0.959	0.031	0.863	0.836	0.886	0.071	0.840	0.782	0.878	0.055	0.888	0.874	0.922	0.047
VST$_{2021}$[93]	44.48	0.896	0.877	**0.939**	0.037	**0.932**	0.944	**0.964**	0.034	**0.928**	0.937	**0.968**	**0.030**	**0.873**	0.850	**0.900**	0.067	0.850	0.800	**0.888**	0.058	**0.895**	0.881	**0.932**	0.045
GFNet$_{2021}$[120]	51.91	0.860	0.830	0.909	0.041	0.905	0.917	0.945	0.038	0.900	0.909	0.950	0.033	0.840	0.838	0.891	0.068	0.818	0.751	0.851	0.060	0.865	0.849	0.909	0.048
UIUNet$_{2023}$[172]	50.54	0.844	0.807	0.883	0.055	0.900	0.909	0.935	0.046	0.894	0.897	0.938	0.039	0.829	0.811	0.870	0.083	0.819	0.753	0.854	0.064	0.857	0.835	0.896	0.057
PriorNet$_{2024}$[190]	-	**0.897**	**0.899**	-	**0.033**	0.931	**0.952**	-	**0.031**	0.921	**0.940**	-	**0.027**	0.865	**0.884**	-	**0.058**	**0.848**	**0.827**	-	**0.051**	0.892	**0.900**	-	**0.040**
Complex-valued																									
DCN$_{2018}$[151]	49.12	0.802	0.743	0.860	0.074	0.860	0.869	0.910	0.084	0.859	0.845	0.911	0.059	0.799	0.774	0.842	0.103	0.776	0.687	0.819	0.080	0.819	0.784	0.868	0.080
FCCN$_{2023}$[173]	48.61	0.811	0.750	0.872	0.070	0.874	0.876	0.918	0.082	0.867	0.857	0.919	0.058	0.807	0.783	0.854	0.100	0.787	0.695	0.827	0.079	0.824	0.813	0.878	0.078
SCVUNet$_{2023}$[170]	54.15	0.824	0.769	0.874	0.063	0.882	0.887	0.922	0.062	0.877	0.872	0.929	0.052	0.815	0.795	0.865	0.089	0.788	0.697	0.875	0.077	0.831	0.827	0.893	0.069
DCS^{2}Net (ours)	**44.32**	**0.905**	**0.886**	**0.957**	**0.035**	**0.943**	**0.947**	**0.969**	**0.028**	**0.931**	**0.939**	**0.972**	**0.027**	**0.875**	**0.858**	**0.908**	**0.059**	**0.849**	**0.805**	**0.893**	**0.051**	**0.901**	**0.887**	**0.940**	**0.039**

4.3.1 Experimental Setup

We evaluate our method, DCS^{2}Net, on binary segmentation tasks across real-valued and complex-valued datasets.

Real-Valued Datasets: For salient object detection, we follow the experimental setup of [93] and evaluate on five benchmark datasets: ECSSD [175] (1,000 images), PASCAL-S [89] (850 images), HKU-IS [84] (4,447 images), DUT-O [176] (5,168 images), and DUTS [167] (10,553 images). For defocus blur detection, we use the CUHK [133] (704 images) and DUT [186] (1,100 images) datasets. We train on 604 images from CUHK and 600 from DUT, and test on the remainder. For shadow detection, we train and test separately on the SBU [162] and ISTD [165] datasets.

Complex-Valued Dataset: We use the Simulated InSAR Building dataset [14] for building detection. We modify the dataset by treating the layover class as foreground. This dataset contains 270 training images and 42 test images. The SAR images are inherently complex and are processed as such by our network.

Evaluation Metrics: We employ task-specific metrics for evaluation. For salient object detection, we use the S_m measure [35], maximum F-measure (maxF), maximum enhanced-alignment measure (E_ξ^{max}) [37], and mean absolute error (MAE). For defocus blur detection, we report F-measure (F_β) and MAE. For shadow detection, we use the balanced error rate (BER), following prior work. For the InSAR building dataset, we use mean Intersection over Union (mIoU).

4.3.2 Implementation Details

Implementation details: In order to have a pretrained encoder for better results, we first train the encoder part of DCS^{2}Net on ImageNet1k[30]. The result of the trained DCS^{2}Net encoder. For this, we resize input images to 256×256 and then randomly crop 224×224 image regions as the model input and use random flipping as data augmentation. Also, for some of the complex-valued methods, we implemented the decoder part and extended them for comparison on various tasks. We randomly initialize weights and use Adam [71] optimizer. Similarly, we take inputs of size 224×224 for binary segmentation tasks. We use the PyTorch library to implement complex-valued operations, which provide native support for complex tensors and complex operations, including convolution and complex FFT.

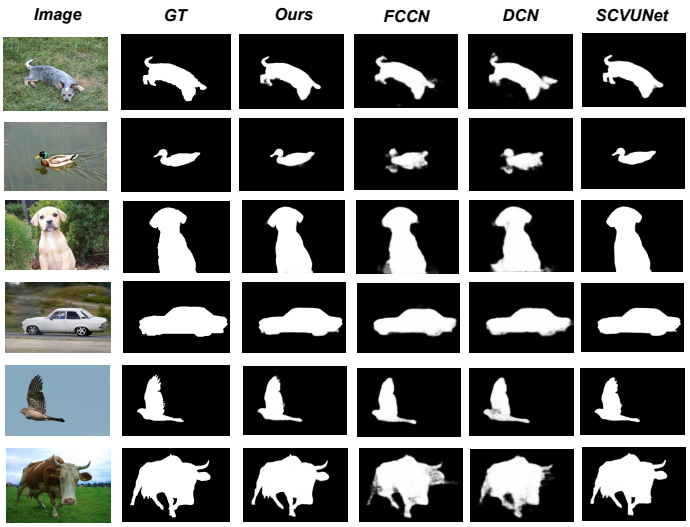


Figure 4.7: Qualitative results for salient object detection. Noticeable improvements can be seen after zooming.

Table 4.3: Comparing classification accuracy(%) of our encoder with only other complex-valued method and other real-valued encoders on ImageNet-1k[30]

Model	Top-1 Acc. (%)
ResNet50[50]	76.1
Vit-S/16[33]	78.1
GFNet-xs[121]	78.6
ResMLP-12[150]	76.6
ResNet152 (DCN)[151]	72.6
ResNet152 (FCCN)[173]	77.3
DCSNet Encoder	**78.8**

4.3.3 Experimental Results

Comparison on real-valued data: We have performed experiments on three different tasks that use real-valued datasets: salient object detection, defocus blur detection, and shadow detection. For salient object detection, we have compared 10 different real-valued methods, including a few of the latest methods. For all five datasets, we see the best or second-best results. We obtain the best average results for 3 out of the four metrics. We also compared with the latest complex-valued methods. We implemented a CNN-based encoder-decoder approach for the complex-valued methods: FCCNs and DCN since we only had encoders for both. On the other hand, SCVUNet is directly used for real-valued datasets. We see a significant improvement over all complex-valued methods, providing a powerful token-based complex-valued method as shown

Table 4.4: Comparison on Complex-valued Building InSAR dataset [14] for foreground extraction. ■: best, ■: second best.

Method	Param (M)	Building mIoU ↑
DCN_{2018}[151]	49.12	0.79
$FCCN_{2023}$[173]	48.61	0.82
$SCVUNet_{2023}$[170]	54.15	**0.86**
DCS^2Net (ours)	**44.32**	**0.91**

in Table 4.2. We also present a few of the qualitative results in Figure 4.7. For defocus blur detection, we compared five real-valued and three complex-valued methods in Table 5.4. For this task, we also observe a similar pattern showing clear improvements over complex-valued networks and the best overall performance on both datasets. For shadow detection, when comparing six real-valued and three complex-valued methods, we observe that we obtain the best results while FDRNet achieves the second-best results in Table 4.7.

Comparison on complex-valued data: For comparison on complex-valued data in Table 4.4, we use the InSAR building dataset to segment the foreground building structure from the complex-valued image. We also benchmark our approach against SCVUNet [170]. Originally developed for ship target segmentation in Synthetic Aperture Radar (SAR) imagery, SCVUNet introduces a Saliency-combined Complex-Valued U-Net architecture. We consider existing complex-valued methods for this task and find that we can easily outperform them. Furthermore, we provide a direct comparison of FLOPs, GPU memory usage, and inference time (for a batch size of 1) between our proposed DCS^2Net and our baseline FCCN in Table 4.5. The results demonstrate that DCS^2Net is significantly more computationally inexpensive.

Table 4.5: Computation comparison between DCS^2Net and FCCN[173]

Method	gFLOPS↓	GPU memory(MB)↓	inference time(ms)↓
FCCN	59	573	84
DCS^2Net	25	291	32

Analysis of Efficiency Gains: The substantial reduction in computational overhead, cutting gFLOPS by more than half and reducing inference time from 84 ms to 32 ms, stems directly from DCS^2Net's architectural evolution. First, at the data ingestion level, DCS^2Net utilizes the compact R2C transformation, compressing the visual input into a two-channel complex tensor. In contrast, FCCN relies on the three-channel iHSV representation; this immediate reduction in input dimensionality propagates through the entire network, permanently shrinking the required feature map depths.

Second, whereas FCCN employs dense, high-resolution complex convolutions that

Table 4.6: Comparison with real and complex-valued methods on defocus blur detection. ■: best, ■: second best.

		BTBNet[186]	DeFusionNet[148]	CENet[187]	DAD[185]	EFENet[184]	DCN[151]	FCCN[173]	SCVUNet[170]	DCS^{2}Net (ours)
		Real-valued					**Complex-valued**			
Params (M)		-	-	-	44.13	43.61	49.12	48.61	54.15	**44.32**
DUT	$F_\beta \uparrow$	0.827	0.823	0.817	0.794	**0.854**	0.857	0.865	0.848	**0.898**
	MAE $\downarrow$	0.138	0.118	0.135	0.153	**0.094**	0.108	0.100	0.096	**0.053**
CUHK	$F_\beta \uparrow$	0.889	0.818	0.906	0.884	**0.914**	0.878	0.899	0.901	**0.921**
	MAE $\downarrow$	0.082	0.117	0.059	0.079	**0.053**	0.089	0.081	0.078	**0.041**

Table 4.7: Comparison with real and complex-valued approaches on shadow detection on BER ($\downarrow$) score. ■: best, ■: second best.

	Stacked CNN[162]	BDRAR[191]	DSC[60]	DSD[189]	MTMT[20]	FDRNet[192]	DCN[151]	FCCN[173]	SCVUNet[170]	DCS^{2}Net (ours)
	Real-valued						**Complex-valued**			
Params (M)	-	42.45	122.49	58.15	44.12	-	49.12	48.61	54.15	44.32
ISTD	8.60	2.69	3.42	2.17	1.72	**1.55**	1.83	1.78	1.91	**1.45**
SBU	-	3.89	5.59	3.45	3.15	**3.04**	3.24	3.22	3.25	**3.01**

are notoriously memory-intensive, DCS^{2}Net adopts a token-based architecture. By compressing spatial features into localized tokens before passing them through the dual-domain spatio-spectral module, the network fundamentally bypasses the massive spatial redundancy inherent in standard convolutions. This shift from dense pixel-wise filtering to compressed token-based frequency routing drastically reduces the sheer volume of complex multiplications required, resulting in a leaner, faster, and memory-efficient approach.

4.3.4 Ablation Study

We conduct four ablation studies to highlight the importance of our contributions. The first two ablation studies are conducted to show the importance of our proposed R2C transform. While the other two focus on the architectural contribution of DCS2Net. All the models for the ablation study are trained from scratch to maintain fairness since we have a wide range of input and architectural changes in the ablation. We also present an additional ablation study to compare complex-valued self-attention with our proposed learnable Fourier filter. For this ablation, we replace the spatio-spectral module with the complex-valued self-attention (SA) module as proposed by [34, 177]. We can notice a performance decline in most cases when the spatio-spectral module is replaced. Table 4.8 empirically validates our proposed method for capturing global information in a complex domain.

Effects of complex-valued input: In this ablation study, we analyze the contribution of various complex-valued inputs. Results are shown in Table 4.9. We compare with 6 complex-valued inputs: RGB+$i0$, HSV+$i0$, iHSV[173], LAB[138], {R+iG, G+iB}[138], and FFT. Across all of the inputs, our iRGB input performs best.

Effect of each channel: For this study, we take each one of the channels of I_{iRGB}

Table 4.8: Additional ablation study to compare our Fourier filter module and self-attention.

Metric	DUTS		ECSSD		HKU-IS		PASCAL-S		DUT-O	
	SA	Ours	SA	Ours	SA	Ours	SA	Ours	SA	Ours
$S_m \uparrow$	0.735	**0.740**	0.828	**0.831**	**0.748**	0.733	0.745	**0.747**	0.738	**0.739**
maxF $\uparrow$	0.651	**0.654**	0.820	**0.825**	0.688	**0.689**	0.711	**0.713**	**0.645**	**0.645**
$E_\xi^{max} \uparrow$	0.788	**0.801**	0.875	**0.884**	0.763	**0.791**	0.787	**0.801**	0.788	**0.795**
MAE $\downarrow$	**0.104**	0.107	0.091	**0.083**	0.170	**0.157**	0.146	**0.131**	0.112	**0.109**

Table 4.9: Results of ablation study on various datasets when we take different inputs to DCS^{2}Net. For each input we trained the network from scratch.

Dataset	DUTS				ECSSD				PASCAL-S				DUT-O			
	$S_m \uparrow$	maxF$\uparrow$	$E_\xi^{max} \uparrow$	MAE$\downarrow$	$S_m \uparrow$	maxF$\uparrow$	$E_\xi^{max} \uparrow$	MAE$\downarrow$	$S_m \uparrow$	maxF$\uparrow$	$E_\xi^{max} \uparrow$	MAE$\downarrow$	$S_m \uparrow$	maxF$\uparrow$	$E_\xi^{max} \uparrow$	MAE$\downarrow$
RGB	0.731	0.641	0.774	0.147	0.820	0.801	0.857	0.114	0.732	0.658	0.847	0.139	0.731	0.625	0.781	0.136
HSV	0.729	0.637	0.768	0.148	0.817	0.800	0.857	0.115	0.725	0.649	0.841	0.141	0.727	0.622	0.776	0.139
iHSV	0.735	0.648	0.780	0.128	0.824	0.813	0.864	0.105	0.742	0.710	0.813	0.128	0.738	0.631	0.789	0.112
L+i0, a+ib	0.733	0.648	0.779	0.128	0.821	0.813	0.864	0.106	0.740	0.711	0.813	0.129	0.741	0.632	0.786	0.112
R+iG, G+iB	0.736	0.651	0.780	0.126	0.820	0.815	0.867	0.105	0.742	0.714	0.811	0.128	0.744	0.634	0.789	0.112
FFT	0.728	0.6374	0.765	0.150	0.814	0.795	0.854	0.116	0.719	0.641	0.835	0.144	0.721	0.615	0.770	0.142
ours	**0.740**	**0.654**	**0.801**	**0.107**	**0.831**	**0.825**	**0.884**	**0.083**	**0.747**	**0.713**	**0.851**	**0.131**	**0.739**	**0.645**	**0.795**	**0.109**

input. Each I_{iRGB} channel performs well; however, we get the best results when both are taken together in Table 4.10. When considering only one channel, we focus only on the complex intensity or the complex hue, which provides incomplete information.

Effects of $\mathscr{L}_{i\text{dense}}$ & T^S: To analyze the importance of $\mathscr{L}_{i\text{dense}}$, we just use binary cross-entropy loss instead and discard the imaginary component of the output. Similarly, to observe the importance of dense token T^S, we remove it. The results of these ablation studies are shown in Table 4.10.

Effects of number of resolutions: Using exactly four resolutions was an architectural design choice that effectively leverages four distinct resolutions for multi-scale feature representation. We conducted additional experiments using two, three, and five resolutions and present a comparison for two datasets in Fig. 4.8. The extensive result of the experiment is provided in Tab. 4.11.

Applicability to Multi-Class Segmentation: While the experiments above focus on binary dense prediction tasks the architectural blueprint of DCS^{2}Net is inherently adapt-

Table 4.10: Results of ablation study that highlights the importance of our loss $\mathscr{L}_{idense}$, and dense token(T^S). All results are calculated without pre-trained encoder.

Dataset	DUTS				ECSSD				PASCAL-S				DUT-O			
	$S_m \uparrow$	maxF$\uparrow$	$E_\xi^{max} \uparrow$	MAE$\downarrow$	$S_m \uparrow$	maxF$\uparrow$	$E_\xi^{max} \uparrow$	MAE$\downarrow$	$S_m \uparrow$	maxF$\uparrow$	$E_\xi^{max} \uparrow$	MAE$\downarrow$	$S_m \uparrow$	maxF$\uparrow$	$E_\xi^{max} \uparrow$	MAE$\downarrow$
ours (I_θ)	0.729	0.629	0.790	0.123	0.810	0.791	0.865	0.109	0.743	0.701	0.794	0.144	0.735	0.634	0.792	0.122
ours (I_ϕ)	0.726	0.631	0.791	0.120	0.805	0.788	0.866	0.112	0.745	0.692	0.789	0.139	0.729	0.636	0.785	0.119
ours (w/o $\mathscr{L}_{idense}$)	0.731	0.630	0.789	0.122	0.824	0.807	0.872	0.097	**0.747**	0.702	0.794	0.143	0.733	0.633	0.788	0.123
ours (w/o T^S)	0.734	0.649	0.787	0.113	0.825	0.818	0.878	0.091	0.749	0.709	0.703	0.138	0.730	0.646	0.792	0.114
ours	**0.740**	**0.654**	**0.801**	**0.107**	**0.831**	**0.825**	**0.884**	**0.083**	**0.747**	**0.713**	**0.851**	**0.131**	**0.739**	**0.645**	**0.795**	**0.109**

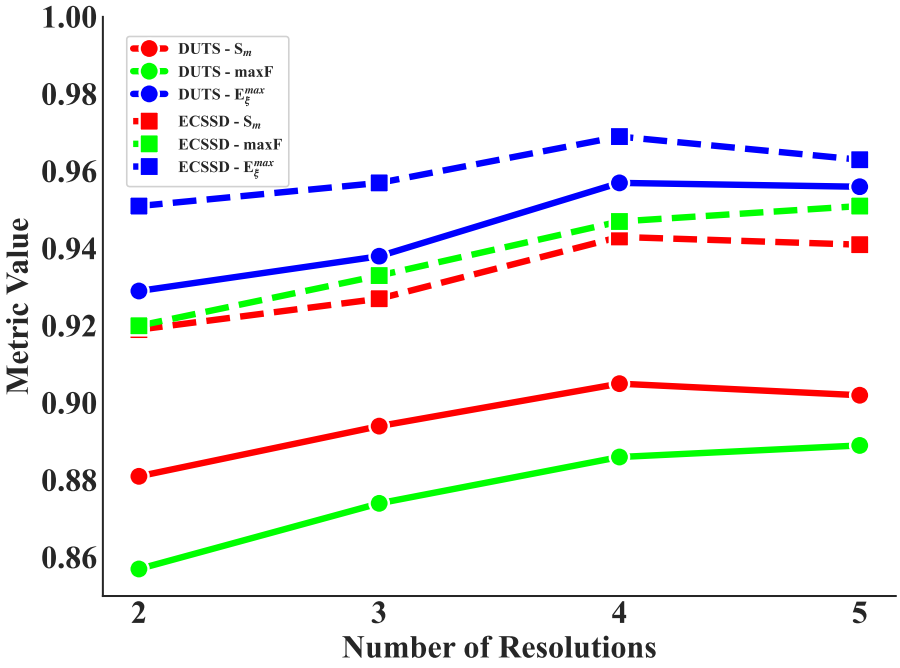


Figure 4.8: Multi-scale resolution comparison on 2 difference datasets

Table 4.11: Results of ablation study highlighting the effect of resolution levels using pre-trained encoder in our proposed DCS^{2}Net.

Dataset	DUTS				ECSSD				PASCAL-S				DUT-O			
	$S_m \uparrow$	maxF↑	$E_\xi^{max} \uparrow$	MAE↓	$S_m \uparrow$	maxF↑	$E_\xi^{max} \uparrow$	MAE↓	$S_m \uparrow$	maxF↑	$E_\xi^{max} \uparrow$	MAE↓	$S_m \uparrow$	maxF↑	$E_\xi^{max} \uparrow$	MAE↓
2 resolutions	0.881	0.857	0.929	0.041	0.919	0.920	0.951	0.034	0.851	0.833	0.897	0.064	0.819	0.756	0.848	0.058
3 resolutions	0.894	0.874	0.938	0.041	0.927	0.933	0.957	0.033	0.864	0.841	0.901	0.064	0.825	0.773	0.857	0.055
4 resolutions	**0.905**	0.886	**0.957**	**0.035**	**0.943**	0.947	**0.969**	**0.028**	**0.875**	0.858	**0.908**	**0.059**	**0.849**	0.805	**0.893**	**0.051**
5 resolutions	0.902	**0.889**	0.956	0.039	0.941	**0.951**	0.963	0.029	0.870	**0.862**	0.903	0.060	0.845	**0.809**	0.889	0.052

able to multi-class and non-binary segmentation.

In our binary formulation, a single complex channel elegantly separates two classes by assigning them to orthogonal unit vectors (1 for foreground and i for background). However, for a multi-class problem with $C > 2$ categories (such as semantic scene parsing), mapping every class onto a single Argand plane by dividing the circle into equal angles (using the C-th roots of unity, $e^{i\frac{2\pi k}{C}}$) can lead to phase crowding. As the number of classes grows, the angular distance between categories shrinks, making optimization unstable along complex object boundaries.

Instead, DCS^{2}Net can be scaled to non-binary tasks via channel-wise expansion. By widening the final complex operator in the decoder to output C distinct complex feature maps, the network represents each semantic category on its own dedicated Argand plane. In this multi-channel setup, optimization is achieved by generalizing our

$\mathscr{L}_{i\text{dense}}$ loss into a one-versus-all formulation (where the target class is mapped to 1 and all non-target classes to i within their respective class channels), or by applying the complex cross-entropy loss ($\mathscr{L}_{comp}$) across the channel dimension. This ensures that the dual-domain spatio-spectral filtering of DCS^{2}Net can be applied to complex multi-class scene understanding without altering the core internal architecture.

Mechanisms of Improved Foreground-Background Extraction

The R2C transformation provides a rigorous foundation for structural separation. By explicitly decoupling an image into two distinct complex channels representing intensity (achromatic) and chromatic (color) information, the network is significantly less susceptible to common visual confounders. For example, in salient object or shadow detection, a dark shadow cast across a background might typically be confused for a distinct foreground object by a standard real-valued network. Because R2C isolates this luminance drop in the intensity channel while leaving the underlying pure color identity intact in the chromatic channel, the network maintains a consistent understanding of the surface material, thereby preventing the foreground prediction from fragmenting.

Beyond the input representation, the DCS^{2}Net architecture itself sharpens foreground extraction. Its spatio-spectral module processes the compressed spatial tokens and their discrete Fourier transform jointly, so the network reasons about local texture and global frequency structure at once, the two cues that most reliably separate a camouflaged or low-contrast object from its background. Because the network operates in the complex field, every token carries magnitude and phase, and the $1 - i$ complex ground-truth encoding drives foreground and background onto orthogonal axes; classification is then resolved by phase rotation rather than by a fragile magnitude threshold, which keeps object boundaries intact where real-valued networks tend to bleed into the background.

4.4 Summary

In this chapter, we have presented DCS^{2}Net, a fully complex-valued token-based network for binary segmentation tasks, which operates both in the spatial and frequency domains. We also propose an invertible R2C transform to encode RGB images into the complex domain for serving as inputs. While maintaining complex-valued information throughout, our model outperforms previous complex-valued networks/methods on various tasks and both real and complex-valued data, presenting a robust approach using complex-valued neural networks.

CHAPTER 5

SAMwave: Wavelet-Driven Feature Enrichment for Effective Adaptation of Segment Anything Model

Large vision models [33, 111, 116] have become the dominant solution for numerous computer vision tasks, significantly advancing fields like semantic segmentation [72, 123] and object detection. However, their substantial parameter counts, while enabling high performance on their trained tasks, lead to sub-optimal generalization to other related problems, a challenge particularly evident in low-level vision tasks. For instance, while SAM (Segment Anything Model) [72] excels at semantic segmentation, it struggles with tasks like camouflaged object detection, polyp detection, defocus blur detection and other such difficult tasks. The difficulty of developing a single, universally high-performing model for all such difficult tasks has led to growing interest in adapting large models [8, 85, 87, 100, 119, 159]. A common approach is to use finetuning strategies; however, it faces several challenges, including catastrophic forgetting [6, 10, 15, 51, 135, 140] and the reliance on large-scale datasets for adequate adaptation.

In this work, we investigate the optimal adaptation of large models to related tasks, focusing particularly on SAM, which comprises a powerful encoder and a prompt-conditioned decoder. We hypothesize that the encoder can produce a sufficiently rich latent representation, such that the decoder can generate an effective segmentation mask for a new task–given appropriate supervision. Previous works [18, 94] have shown the efficacy of providing high-frequency features to the encoder for adapting a frozen decoder. These methods are typically performed by transforming input images into the Fourier domain and applying a predefined mask to emphasize high-frequency components. Different images possess varying frequency distributions, and a fixed mask leads to suboptimal feature selection. Our empirical study, presented in Tab. 5.1, supports this observation–demonstrating that simply inverting the mask yields comparable performance across tasks. Fig. 5.1 further illustrates this issue. This motivates our de-

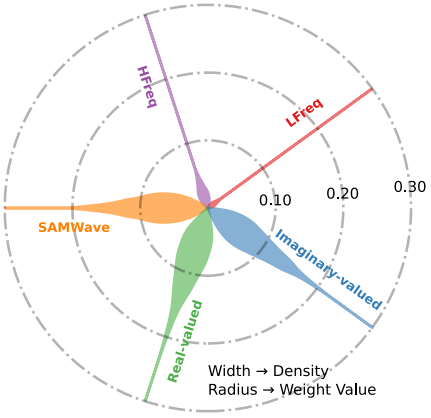


Figure 5.1: Learned weights for the previous high frequency extraction method, compared with SAMWave's real and complex adapter weights. Unlike the previous method, SAMWave's weights are not concentrated around 0.

parture from rigid, predefined Fourier-domain masks toward a more adaptive feature selection mechanism. To this end, we propose a novel high-frequency feature extraction method using wavelet transform, which offers superior spatial and frequency localization [75, 117, 145]. Specifically, we extract horizontal (I_{lh}), vertical (I_{hl}), and diagonal (I_{hh}) features from an image I, combining them into a high-frequency image $I_{HF} = I_{lh} + I_{hl} + I_{hh}$, which provides finer details and texture to the encoder. Wavelet transform is advantageous due to its ability to capture frequency and spatial information, allowing for a more precise representation of image details than Fourier-based methods. We further extend this approach with complex wavelet transforms and complex-valued adapters. Complex-valued representations offer a more comprehensive way to capture image information, potentially leading to improved feature extraction and adaptation, as demonstrated in various applications[152, 174]. We are the first to show the adaptation of large vision models using a complex-valued approach. Our contributions are summarized as follows: ***1)*** *We demonstrate the limitations of using the Fourier domain with predefined masks for high-frequency feature extraction.* ***2)*** *We introduce a Wavelet High-Frequency (WHF) module that leverages the wavelet transform for effective finetuning.* ***3)*** *We propose a complex-valued extension of our method using complex wavelet transform and complex-valued adapters.* ***4)*** *Our experiments show significant performance improvements over existing tuning methods for SAM and SAM2 across four difficult tasks.*

5.1 Related Works

Visual Adapters: The paradigm of adapting large pre-trained models for downstream tasks using adapter-based finetuning, initially popular in NLP [59], has effectively extended to computer vision [19, 88]. Adapter-based fine-tuning is a highly effective form of parameter-efficient fine-tuning. Instead of updating all the weights of a massive pre-trained model, this approach freezes the original model and inserts small, learnable neural network modules called adapters between its layers. During training for a new downstream task, only the weights of these newly added adapters are updated. This method offers a compelling balance between performance and computational efficiency. Recent work has shown its applicability to vision foundation models like SAM [72], notably through visual prompting methods [18, 94]. While prior adaptation methods often prioritize global context, our work focuses on incorporating crucial local, high-frequency information via wavelets to enhance SAM's performance on low-level vision tasks.

Low-level Vision Tasks: Our work addresses several low-level vision tasks where fine-grained details are critical for accurate localization. **Camouflaged Object Detection (COD)** involves identifying objects that are seamlessly blended into their background, a significant challenge due to the low contrast and texture similarity between the object and its surroundings [38, 47, 83]. Success in this task requires models to meticulously capture subtle boundary cues and texture discontinuities [113, 179]. **Defocus Blur Detection (DBD)** aims to precisely segment out-of-focus regions in an image. This relies on detecting subtle variations in blurriness across the scene, where the transition from sharp to blurry areas, often characterized by local edge information, is a key indicator for segmentation [69, 134, 148]. **Shadow Detection (SD)** requires distinguishing shadows from actual objects or dark surfaces by analyzing illumination patterns and local context [60, 192]. The task is challenging as shadows can alter an object's appearance, and accurately identifying their soft or sharp boundaries is crucial for robust scene understanding. **PolyP Detection (PD)** focuses on the precise delineation of polyp regions from endoscopic images, a critical step for assisting clinicians in the early detection of colorectal abnormalities [132, 154]. The task is demanding due to the highly variable shape, size, and subtle appearance of polyps against the surrounding tissue.

While some methods [94] use generic high-frequency components for low-level tasks, we propose a more principled approach using specific wavelet decompositions to provide targeted local feature guidance to SAM for these diverse low-level vision challenges.

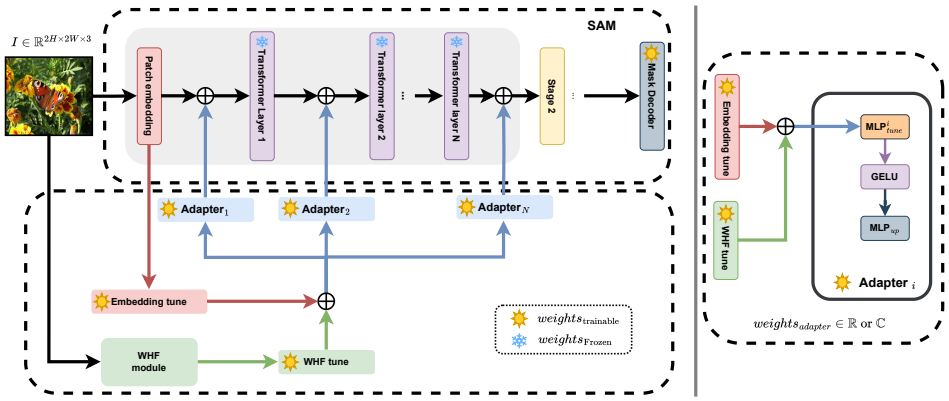


Figure 5.2: Overview of SAMWave for difficult vision tasks. **Left** shows that the image is passed through SAM and our WHF (wavelet high frequency) module, where the high-frequency features are extracted. The extracted details are then combined with patch embeddings obtained from the transformer encoder. Note that the encoder is frozen; we only updated the decoder for our training. **Right** shows the overview of a single adapter used; for complex adapters, we replace the weights of MLP layers with complex values.

5.2 Methodology

In this section, we describe our proposed network: SAMWave. The idea is to leverage the high-frequency features obtained from the wavelet transform and use them to guide the Segment-Anything model (SAM) [72] for more precise segmentation in various tasks.

Table 5.1: Result of low frequency (LFreq) & high frequency (HFreq) on camouflaged object detection on three separate datasets. We also show the difference in the performance of both approaches. For each metric, the shaded cell marks the better-performing setting (green for HFreq, orange for LFreq), while the red values in the last row report the signed difference between the two settings (HFreq − LFreq).

	CHAMELEON				CAMO				COD10K			
	$\mathcal{S}_m \uparrow$	$\mathcal{E}_\phi \uparrow$	$\mathcal{F}_\beta^w \uparrow$	$\mathcal{M} \downarrow$	$\mathcal{S}_m \uparrow$	$\mathcal{E}_\phi \uparrow$	$\mathcal{F}_\beta^w \uparrow$	$\mathcal{M} \downarrow$	$\mathcal{S}_m \uparrow$	$\mathcal{E}_\phi \uparrow$	$\mathcal{F}_\beta^w \uparrow$	$\mathcal{M} \downarrow$
HFreq	**0.896**	**0.919**	0.824	**0.033**	**0.847**	**0.873**	**0.765**	**0.070**	**0.883**	**0.918**	0.801	**0.025**
LFreq	0.888	0.903	**0.830**	0.034	0.824	0.843	**0.765**	0.078	0.872	0.901	**0.803**	0.027
	+ 0.008	+ 0.016	- 0.006	+ 0.001	+ 0.023	+ 0.030	0.000	+ 0.008	+ 0.009	+ 0.017	- 0.002	+ 0.002

5.2.1 Motivation for High-Frequency Feature Utilization

Precise segmentation, particularly the accurate delineation of object boundaries, is a challenging task that fundamentally relies on capturing fine-grained image details. These details are predominantly encoded within the high-frequency components of the image

signal. Therefore, effectively leveraging high-frequency features is crucial for achieving high-quality segmentation masks.

Previous methods aiming to incorporate high-frequency information, such as those in [18, 94], have often relied on extracting these features in the Fourier domain using a predefined, fixed mask. However, a significant limitation of this *"one mask fits all"* approach is its inability to adapt to the diverse characteristics and varying distributions of high-frequency information across different images and tasks. Consequently, a static Fourier mask may not effectively isolate the relevant high-frequency cues necessary for robust fine-grained segmentation in all scenarios.

We conducted an empirical study to investigate the efficacy of this fixed-mask approach for capturing essential high-frequency information. We compared the performance using features extracted via the standard high-frequency mask (HFreq) against those extracted by inverting this mask to obtain a low-frequency mask (LFreq). Our experiments focused on the challenging task of camouflaged object detection, and the results are summarized in Tab. 5.1. While HFreq shows superior performance on *9* out of *12* metrics compared to LFreq, a closer examination reveals that the average performance gain offered by HFreq is remarkably small, merely **0.009**. Despite high frequencies theoretically being more aligned with boundary information, this marginal difference suggests that a fixed Fourier mask is not an optimal strategy for extracting discriminative high-frequency features that significantly enhance segmentation performance across varied data. This observation motivates the need for a more adaptive and practical approach to harness high-frequency information to guide models like SAM towards more precise segmentation.

5.2.2 SAM as the Backbone

Our SAMWave framework employs the Segment-Anything Model (SAM) [72] as its core backbone, leveraging its robust pre-trained capabilities for various segmentation tasks. SAM comprises three main components: an image encoder, a prompt encoder, and a mask decoder.

The image encoder, a pre-trained ViT-H/16, processes the input image $I \in \mathbb{R}^{H\times W\times 3}$ to produce dense embeddings $E_{\text{img}} \in \mathbb{R}^{\frac{H}{16}\times\frac{W}{16}\times 256}$. These embeddings are spatially downscaled by $16\times$ and have 256 channels after a 1×1 convolution. The prompt encoder processes input prompts (such as dense features from our method), typically using convolutions, and these prompt embeddings are element-wise added to E_{img}. The mask decoder, a lightweight transformer-based network inspired by [11, 21], takes the combined embeddings and predicts the final segmentation mask via upsampling and a linear classifier.

In SAMWave, we freeze the weights of the pre-trained image encoder and train only the parameters of the mask decoder to adapt SAM's segmentation generation to the task-specific guidance.

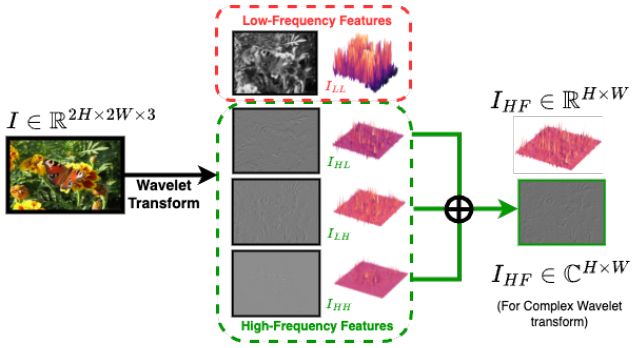


Figure 5.3: An overview of the WHF module, we first use the wavelet transform to obtain high- and low-frequency features. Then, we combine the high-frequency features to create a feature map. For complex-valued features, the methodology remains the same but utilizes a complex wavelet transform for the initial decomposition step.

5.2.3 Wavelet High-Frequency Module (WHF)

The Wavelet High-Frequency (WHF) module extracts rich high-frequency spatial features using the Discrete Wavelet Transform (DWT). Given an input image $I \in \mathbb{R}^{H\times W\times 3}$, we first resize it to $I' \in \mathbb{R}^{2H\times 2W\times 3}$. This upsampling compensates for the DWT's inherent downsampling, ensuring output subbands match the original $H \times W$ resolution.

From a geometric standpoint, performing upsampling *before* wavelet decomposition is strictly necessary to maintain spatial consistency across the multi-modal pipeline. SAM's native image encoder applies a 16×16 patch embedding downsampling to generate its baseline semantic token grid ($\psi_{PE} \in \mathbb{R}^{\frac{H}{16}\times\frac{W}{16}\times\gamma}$). By upsampling the image first and subsequently applying a 1-level wavelet decomposition, our Wavelet High-Frequency (WHF) module produces directional subbands at exactly half the standardized resolution ($\frac{H}{2} \times \frac{W}{2}$). This exact dyadic scaling allows our convolutional patch projection (ϕ_{whf}) to map the frequency subbands onto the exact same $\frac{H}{16} \times \frac{W}{16}$ spatial token grid without fractional pixel misalignments or phase shifting, guaranteeing 1:1 spatial correspondence between semantic patch embeddings and localized edge features.

Applying a 2D DWT to I' decomposes it into four subbands at $H \times W \times 3$ resolution: I'_{ll} (approximation), I'_{lh} (horizontal details), I'_{hl} (vertical details), and I'_{hh} (diagonal details). $I' \xrightarrow{\text{DWT}} \{I'_{ll}, I'_{lh}, I'_{hl}, I'_{hh}\}$ Focusing on fine details, we exclude the low-frequency I'_{ll} subband and combine the high-frequency detail subbands ($I'_{lh}, I'_{hl}, I'_{hh}$) to form the composite high-frequency map I_{HF}. $I_{HF} = I'_{lh} + I'_{hl} + I'_{hh} \in \mathbb{R}^{H\times W\times 3}$ captures edge and texture information (Fig. 5.3).

A key advantage of this wavelet-based approach is its inherent flexibility. While we describe the process using standard real-valued DWT, the WHF module can seamlessly incorporate complex wavelet transforms, such as the Dual-Tree Complex Wavelet Transform (DT-CWT)[97]. The core procedural steps (resizing, wavelet decomposition into subbands, selection and combination of high-frequency subbands, and linear pro-

jection) remain identical. The only change lies in the specific wavelet basis functions used in the decomposition, allowing for adaptation to properties captured by complex wavelets[128] (*e.g.*, phase information, shift invariance) without altering the module's overall structure.

Computational Overhead Analysis: Fourier vs. Wavelet Adapters: We examine the computational overhead of frequency-domain adaptation across three distinct operational dimensions: asymptotic time complexity, hardware memory locality, and architectural pipeline efficiency.

1. **Asymptotic Time Complexity:** The 2D Fast Fourier Transform (FFT) computes a global spectrum with $\mathcal{O}(HW \log(HW))$ complexity, creating a scaling bottleneck at foundation model resolutions (e.g., 1024×1024 in SAM). In contrast, the Discrete Wavelet Transform (DWT) computes spatial frequencies via localized filter banks in strictly linear $\mathcal{O}(HW)$ time. Even our multi-directional Dual-Tree Complex Wavelet Transform (DT-CWT) incurs only a $4\times$ constant factor overhead, preserving optimal linear scaling.

2. **Hardware Locality and Memory Bandwidth:** Because FFT is an intrinsically global operation, computing a single coefficient requires reading across the entire spatial grid. This forces non-sequential memory lookups that break GPU cache locality and saturate high-bandwidth memory (HBM). Conversely, wavelets rely on compact spatial kernels (e.g., Haar, Daubechies) operating within localized windows. This sequential access pattern maximizes L1/L2 cache efficiency, delivering significantly faster real-world wall-clock execution than FFTs of equivalent theoretical size.

3. **Architectural Pipeline Efficiency:** Fourier adapters require repetitive FFT $\rightarrow$ Filter $\rightarrow$ IFFT round-trip transformations inside every transformer block to return features to the spatial domain, introducing severe runtime latency. SAMWave eliminates this penalty by applying the wavelet decomposition *only once* at ingestion to create token-aligned frequency subbands (ψ_{whf}). Once inside the backbone, cross-modal modulation relies entirely on lightweight linear matrix multiplications ($\mathbf{W}_F \psi_{whf}$) within the bottleneck adapters, capturing multi-scale boundary cues without ever requiring an inverse transform.

5.2.4 SAMWave Architecture

SAMWave enhances SAM's segmentation capabilities by integrating learned high-frequency wavelet features *via* lightweight adapters. As illustrated in Fig. 5.2, the architecture consists of three key modules–Embedding Tune, WHF Tune, and Adapters–which are integrated into the SAM backbone.

Embedding Tune: This module generates a learned spatial feature map $\psi_{PE} \in \mathbb{R}^{\frac{H}{16} \times \frac{W}{16} \times \gamma}$ from SAM's image patch embeddings $\mathbf{F}_{\text{emb}}$ using a learnable MLP layer $\phi_{PE}(.; \theta_{PE})$:

$$\psi_{PE} = \phi_{PE}(\mathbf{F}_{\text{emb}}; \theta_{PE})$$

WHF Tune: Complementary to ψ_{PE}, this module processes the high-frequency feature map I_{HF} from the WHF module (§5.2.3). Depending on the wavelet, I_{HF} can be real or complex. Since the features I_{HF} are 2D spatial features, they are divided into patches and projected using a convolution layer to obtain ψ_{whf}, matching ψ_{PE}'s dimensions.

Real and Complex Adapters: We inject bottleneck adapters within SAM's pre-trained image encoder layers to explicitly couple global semantics with local boundary frequencies. For any given layer i, the adapter takes the channel-wise concatenation of the semantic and high-frequency features as its input, $\Psi = [\psi_{PE}; \psi_{whf}] \in \mathbb{R}^{\frac{H}{16} \times \frac{W}{16} \times 2\gamma}$ (or $\mathbb{C}^{\frac{H}{16} \times \frac{W}{16} \times 2\gamma}$).

To understand the analytical interaction between these two distinct modalities, let the down-projection MLP $\phi^{(i)}_{down}$ be parameterized by a block weight matrix $\mathbf{W}_{down} = [\mathbf{W}_S \, ; \mathbf{W}_F]$, where $\mathbf{W}_S$ and $\mathbf{W}_F$ represent the learnable weights mapping the semantic patch embeddings and wavelet frequency features, respectively, into a compressed bottleneck dimension $r \ll \gamma$. The initial linear transformation expands algebraically as:

$$\mathbf{H}^{(i)} = \phi^{(i)}_{down}(\Psi) = \mathbf{W}_S \psi_{PE} + \mathbf{W}_F \psi_{whf} \tag{5.1}$$

Equation 5.1 demonstrates that the down-projection functions as an analytical cross-modal routing mechanism. Rather than processing the two representations in parallel silos, the linear combination $\mathbf{W}_S \psi_{PE} + \mathbf{W}_F \psi_{whf}$ forces a direct, pixel-by-pixel algebraic summation between global object semantics (ψ_{PE}) and localized multi-scale edge contours (ψ_{whf}).

This fused intermediate representation is subsequently modulated by a non-linear activation σ and projected back to the backbone channel dimension via an up-projection MLP ($\phi^{(i)}_{up}$):

$$\mathbf{P}^{(i)} = \phi^{(i)}_{up}(\sigma(\mathbf{H}^{(i)})) = \phi^{(i)}_{up}(\sigma(\mathbf{W}_S \psi_{PE} + \mathbf{W}_F \psi_{whf})) \tag{5.2}$$

where σ denotes the GELU activation function for real-valued adapters. Because the non-linearity acts directly on the summed algebraic terms, it introduces cross-domain gating: strong high-frequency edge activations in ψ_{whf} can dynamically amplify or suppress the semantic attention weights derived from ψ_{PE} before the representation is returned to the backbone.

Finally, the adapter output is aggregated with the frozen layer's original feature map $\mathbf{F}^{(i)}_{img}$. For real-valued adapters, this is executed via direct residual addition ($\mathbf{F}^{(i)}_{img} + \mathbf{P}^{(i)}$). For complex-valued adapters, where the MLPs operate entirely within the complex field $\mathbb{C}$ and σ applies a split-GELU to coordinate phase and magnitude simultaneously, the complex output $\mathbf{P}^{(i)}$ must be projected back to the real domain to interface

with SAM's native ViT blocks. We achieve this by independently extracting and aggregating the real (Re) and imaginary (Im) components:

$$\widetilde{\mathbf{F}}_{img}^{(i)} = \mathbf{F}_{img}^{(i)} + \alpha\,\mathrm{Re}(\mathbf{P}^{(i)}) + \beta\,\mathrm{Im}(\mathbf{P}^{(i)}) \tag{5.3}$$

where α and β are learnable scaling parameters. This analytical injection ensures that multi-directional phase angles (representing fine-grained boundary orientations) directly perturb the real-valued self-attention maps of the foundation model, dynamically re-weighting SAM's global feature space with high-precision boundary awareness at multiple network depths.

SAMWave benefits from: 1) **Feature Complementarity:** combining SAM's semantic features with explicitly learned high-frequency details; 2) **Wavelet Flexibility:** easily integrates various wavelets, including complex ones with $\mathbb{C}$-Adapters to leverage properties like shift invariance and phase information for improved robustness on challenging tasks.

5.2.5 Objective Function

Following [18, 94], we optimize the tasks in SAMWave by minimizing the loss between the predicted binary mask ($\hat{m}$) and ground truth (m). Specifically, we use binary cross-entropy loss ($\mathscr{L}_{bce}$)[29] for defocus blur detection, balanced binary cross-entropy loss ($\mathscr{L}_{bbce}$) for shadow detection, and summation of binary cross-entropy and intersection-over-union loss ($\mathscr{L}_{bbce} + \mathscr{L}_{iou}$) for camouflaged object detection. The loss functions are formulated below as follows:

$$\mathscr{L}_{bce}(\hat{m}, m) = -\big(m * log(\hat{m}) + (1 - m) * log(1 - \hat{m})\big) \tag{5.4}$$

$$\mathscr{L}_{bbce}(\hat{m}, m) = -\zeta * \big(m * log(\hat{m}) + (1 - m) * log(1 - \hat{m})\big) \tag{5.5}$$

$$\zeta = \frac{\sum m + \epsilon}{\sum m + \sum(1 - m) + \epsilon} \tag{5.6}$$

$$\mathscr{L}_{iou}(\hat{m}, m) = 1 - \frac{\sum(\hat{m} * m)}{\sum(\hat{m} + m) - \sum(\hat{m} * m)} \tag{5.7}$$

5.3 Experimental Details and Results

We evaluate our approach on four difficult vision tasks: camouflaged object detection, shadow detection, defocus blur detection, and polyp detection, and compare the performance with current state-of-the-art methods. We also perform ablation studies to analyze the effectiveness of our approach.

Table 5.2: Comparison of Camouflaged Object Detection performance on three datasets against task-specific and finetuning methods. Our SAMwave approach demonstrates improved performance across all datasets compared to existing methods. For every metric, the best and second-best scores are highlighted within each dataset independently, not cumulatively across datasets (■: best, ■: second best).

Method	CHAMELEON				CAMO				COD10K			
	$\mathcal{S}_m \uparrow$	$\mathcal{E}_\phi \uparrow$	$\mathcal{F}_\beta^w \uparrow$	$\mathcal{M} \downarrow$	$\mathcal{S}_m \uparrow$	$\mathcal{E}_\phi \uparrow$	$\mathcal{F}_\beta^w \uparrow$	$\mathcal{M} \downarrow$	$\mathcal{S}_m \uparrow$	$\mathcal{E}_\phi \uparrow$	$\mathcal{F}_\beta^w \uparrow$	$\mathcal{M} \downarrow$
SINet[38]	0.869	0.891	0.740	0.044	0.751	0.771	0.606	0.100	0.771	0.806	0.551	0.051
RankNet[102]	0.846	0.913	0.767	0.045	0.712	0.791	0.583	0.104	0.767	0.861	0.611	0.045
JCOD[82]	0.870	0.924	-	0.039	0.792	0.830	-	0.082	0.800	0.872	-	0.041
PFNet[104]	0.882	0.942	0.810	0.033	0.782	0.852	0.695	0.085	0.800	0.868	0.660	0.040
FBNet[90]	0.888	0.939	0.828	0.032	0.783	0.839	0.702	0.081	0.809	0.889	0.684	0.035
FSPNet[62]	-	-	-	-	0.856	0.899	0.799	0.050	0.851	0.895	0.735	0.026
SAM[72]	0.727	0.734	0. 639	0.081	0.684	0.687	0.606	0.132	0.783	0.798	0.701	0.050
SAM2[122]	0.359	0.375	0.115	0.357	0.350	0.411	0.079	0.311	0.429	0.505	0.115	0.218
EVP[94]	0.871	0.917	0.795	0.036	0.846	0.895	0.777	0.059	0.843	0.907	0.742	0.029
SAM-Adapter[18]	0.896	0.919	0.824	0.033	0.847	0.873	0.765	0.070	0.883	0.918	0.801	0.025
SAM2-Adapter[17]	0.915	0.955	**0.889**	0.018	0.855	0.909	0.810	0.051	0.899	0.950	0.850	0.018
SAM as Backbone												
Real-Valued Adapters												
Ours (daubechies (db))	0.922	0.947	0.856	0.023	0.865	0.903	0.790	0.057	0.900	0.936	0.824	0.021
Ours (coiflet (cf))	0.923	0.952	**0.890**	0.022	0.856	0.888	0.807	0.060	**0.903**	0.939	0.857	0.019
Ours (haar (ha))	0.923	0.954	0.866	0.023	0.868	0.905	0.798	0.056	0.899	0.934	0.819	0.022
Ours (symlet (sym))	0.915	0.947	0.845	0.027	0.846	0.880	0.777	0.067	0.897	0.938	0.822	0.021
Complex-Valued Adapters												
Ours (Symmetric-b (sy-b))	**0.931**	0.961	0.883	0.020	0.856	0.896	0.797	0.058	0.902	0.940	0.843	0.019
Ours (Symmetric-a (sy-a))	**0.928**	0.957	0.881	0.019	0.858	0.893	0.800	0.057	0.902	0.939	0.841	0.019
SAM2 as Backbone												
Real-Valued Adapters												
Ours (daubechies (db))	0.912	0.952	0.880	0.018	**0.870**	**0.927**	0.828	**0.045**	**0.905**	**0.956**	0.861	**0.016**
Ours (coiflet (cf))	0.917	**0.963**	0.883	**0.017**	0.866	0.923	0.824	0.048	**0.905**	**0.957**	**0.862**	**0.016**
Ours (haar (ha))	0.918	0.962	**0.889**	**0.017**	0.869	0.926	**0.829**	**0.047**	0.900	0.952	**0.865**	**0.017**
Ours (symlet (sym))	0.917	0.961	0.887	**0.017**	**0.871**	**0.929**	**0.831**	**0.045**	**0.905**	0.954	0.859	**0.016**
Complex-Valued Adapters												
Ours (Symmetric-b (sy-b))	0.917	**0.966**	**0.890**	**0.016**	0.864	0.921	0.824	0.049	0.902	0.955	0.858	**0.017**
Ours (Symmetric-a (sy-a))	0.917	0.960	0.881	0.019	0.864	0.921	0.826	0.048	**0.903**	0.955	0.856	**0.016**

5.3.1 Datasets and Evaluation Metrics

Camouflaged Object Detection: Following the standard protocol [18, 94], we train our model using a combined dataset of 4,040 images, consisting of 3,040 images from COD10K [38] and 1,000 from CAMO [80]. To provide self-contained context on these benchmarks, COD10K serves as the largest and most diverse dataset in this domain, encompassing high-quality annotations distributed across 78 distinct object categories (e.g., terrestrial, flying, and aquatic). CAMO introduces an additional layer of complexity by featuring both naturally and artificially camouflaged objects across eight categories. We evaluate performance on a combined test set of 2,352 images. Notably, to rigorously assess cross-dataset generalization, the CHAMELEON [139] dataset, a highly specialized collection of 76 internet-sourced images focusing entirely on concealed animals, is held out entirely and utilized strictly for testing alongside the test splits of COD10K (2,026 images) and CAMO (250 images). Performance is measured using four standard metrics: S-measure ($\mathcal{S}_m$) [36], mean E-measure ($\mathcal{E}_\phi$), weighted F-measure ($\mathcal{F}_\beta^w$) [103], and Mean Absolute Error ($\mathcal{M}$).

Shadow Detection: Following [18, 94], we evaluate on SBU [161] and ISTD [166]. Metric: Balanced Error Rate (BER).

Defocus Blur Detection: Following [18, 94], we evaluate on CUHK [133] and DUT [186]. Metrics: F-measure ($\mathscr{F}_\beta$) and Mean Absolute Error ($\mathscr{M}$).

PolyP Detection: Following [154], we evaluate on Kvasir-SEG [63], CVC-ClinicDB [7], ColonDB [146], ETIS [136]. Metrics: $mDice$ and $mIoU$.

Additional details about the datasets, including their train-test split, are provided in the supplementary.

5.3.2 Implementation Details

All experiments are implemented in PyTorch on an NVIDIA A100 GPU. We follow a similar training setting as [18, 94] with an image size of 1024×1024. We use pre-trained SAM and SAM2 networks as the backbone. We use the AdamW optimizer for weight updates and use a learning rate of 0.0002.

Wavelet Selection and Decomposition Depth: We standardize our pipeline to a single-level (1-level) wavelet decomposition after initial image resizing. Theoretically, a 1-level Discrete Wavelet Transform (DWT) or Dual-Tree Complex Wavelet Transform (DT-CWT) naturally halves the spatial resolution ($H/2 \times W/2$) while cleanly segregating the signal into one low-frequency approximation and three (or six) high-frequency directional subbands. This exact $2:1$ spatial reduction perfectly aligns with the initial tokenization and downsampling geometry of foundational vision backbones like SAM, allowing high-frequency details to be injected directly into the encoder through lightweight adapters without requiring lossy pooling or artificial padding. Furthermore, halting decomposition at level one preserves the localized, high-resolution boundary cues necessary for dense prediction; deeper decompositions would smear these spatial details across wider receptive fields, which is detrimental for localizing camouflaged boundaries.

Our specific wavelet bases were selected to provide a structured gradient of geometric properties:

- **Real-Valued Bases (*haar*, *daubechies*, *symlet*):** The Haar (*ha*) wavelet possesses the shortest possible spatial support, making it mathematically optimal for isolating abrupt, step-like edge transitions without spatial bleeding. Daubechies (*db*) and Symlets (*sym*) offer higher vanishing moments, which theoretically allows them to model smooth, continuous polynomial curves and complex natural textures rather than just simple steps. Symlets are specifically included because they enforce near-linear symmetry, preventing the phase distortion (spatial shifting of edges) common in asymmetric filters.

- **Complex-Valued Bases (*Symmetric-a*, *Symmetric-b*):** For our complex-valued adapters, standard real wavelets are insufficient because they suffer from shift-

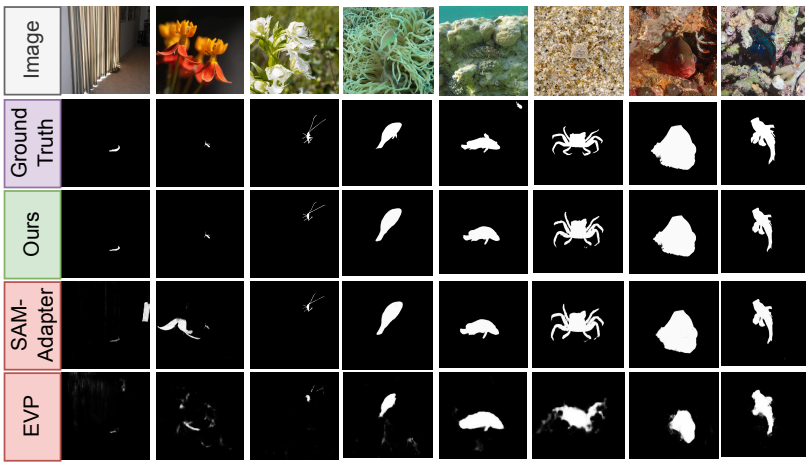


Figure 5.4: Visual comparison of our proposed SAMWave with existing methods for the COD task.

variance and limited directional resolution. We employ symmetric dual-tree complex filters (*sy-a* and *sy-b*), which are designed as approximate Hilbert transform pairs. This complex formulation provides *approximate shift-invariance*, ensuring that a tiny sub-pixel movement of a camouflaged object does not cause erratic swings in energy across subbands, and expands *directional selectivity* from three orthogonal axes to six distinct orientations ($\pm 15°, \pm 45°, \pm 75°$). Capturing these multi-directional phase angles is the exact theoretical mechanism that allows our complex adapters to separate subtle foreground contours from background visual clutter.

5.3.3 Experimental Results

Camouflaged Object Detection: We compare our approach with existing task-specific methods [90, 101, 105] and efficient tuning methods [18, 94]. Tab. 5.2 shows that SAMWave significantly outperforms these methods across three datasets, consistently with SAM and SAM2 backbones. To demonstrate the efficacy of our adaptive approach, we show that performance improvement over previous methods is consistent across different wavelets. Despite vanilla SAM2 performing poorly compared to SAM and other methods in Tab. 5.2, its performance is significantly boosted when combined with our approach. We provide the qualitative results in Fig. 5.4.

Shadow Detection: We compare the performance of our approach against existing methods across two benchmark datasets. In both cases, our method yields substantial performance improvements. Although the performance improvements are relatively minor on the ISTD dataset when using Daubechies and Coiflet wavelets compared to

Table 5.3: Comparison of state-of-the-art approaches on shadow detection.(■: best, ■: second best)

Dataset	DSD [189]	FDRNet [192]	MTMT [20]	EVP [94]	SAM-Adapter[18]	SAM2-Adapter[17]	Ours											
							SAM as Backbone						SAM2 as Backbone					
							Real-valued Adapters				Complex-valued Adapters		Real-valued Adapters				Complex-valued Adapters	
							db	cf	ha	sym	sy-b	sy-a	db	cf	ha	sym	sy-b	sy-a
ISTD ($BER\downarrow$)	2.17	1.55	1.72	1.35	1.43	1.43	1.24	1.24	**1.01**	1.04	1.09	1.15	1.91	1.24	**0.92**	1.22	2.60	1.32
SBU ($BER\downarrow$)	3.45	3.04	3.15	4.31	-	-	**2.93**	3.01	3.00	3.07	3.06	3.06	3.04	5.28	**2.99**	3.77	4.43	4.32

other wavelet variants, the improvement remains significant, demonstrating the robustness and generalizability of our adaptive approach.

Table 5.4: Comparison with state-of-the-art approaches on blur detection.(■: best, ■: second best)

Dataset		DeFusion-Net[148]	BTBNet [186]	CENet [187]	DAD [185]	EFENet [184]	EVP [94]	Ours											
								SAM as Backbone						SAM2 as Backbone					
								Real-valued Adapters				Complex-valued Adapters		Real-valued Adapters				Complex-valued Adapters	
								db	cf	ha	sym	sy-b	sy-a	db	cf	ha	sym	sy-b	sy-a
DUT	$\mathcal{F}_\beta\uparrow$	0.823	0.827	0.817	0.794	0.854	0.890	0.888	0.886	0.880	0.885	0.892	0.893	0.896	0.893	0.894	0.893	**0.907**	**0.902**
	$\mathcal{M}\downarrow$	0.118	0.138	0.135	0.153	0.094	0.068	0.068	**0.067**	0.108	0.106	**0.067**	0.066	**0.051**	**0.051**	**0.051**	**0.051**	**0.052**	0.056
CHUK	$\mathcal{F}_\beta\uparrow$	0.818	0.889	0.906	0.884	0.914	**0.928**	0.922	0.916	0.911	0.911	0.917	0.912	0.923	0.924	**0.925**	0.924	**0.928**	**0.925**
	$\mathcal{M}\downarrow$	0.117	0.082	0.059	0.079	0.053	0.045	0.040	0.046	0.070	0.069	0.045	0.046	**0.036**	0.039	**0.037**	0.038	**0.036**	0.039

Defocus Blur Detection: For the Blur Detection task, we primarily compare our method with EVP[94], as SAM-Adapter[18] does not report results for this task. As shown in Tab. 5.4, our approach achieves the best results using complex-valued adapters with the SAM2 backbone. While we outperform EVP in all cases, our method demonstrates significant improvements over other existing approaches as well.

Table 5.5: Comparison with state-of-the-art methods on PolyP detection. (■: best, ■: second best)

Dataset		M^2UNet [153]	SAM Adapter [18]	SAM2 Adapter [17]	SAM-EG [154]	Ours											
						SAM as Backbone						SAM2 as Backbone					
						Real-valued Adapters				Complex-valued Adapters		Real-valued Adapters				Complex-valued Adapters	
						db	cf	ha	sym	sy-b	sy-a	db	cf	ha	sym	sy-b	sy-a
ClinicDB	$mDice\uparrow$	**0.901**	-	-	**0.931**	0.855	0.652	0.866	0.854	0.832	0.840	0.887	0.881	0.885	0.874	0.862	0.846
	$mIoU\uparrow$	**0.853**	-	-	**0.879**	0.789	0.449	0.794	0.786	0.761	0.772	0.826	0.821	0.827	0.816	0.800	0.784
ColonDB	$mDice\uparrow$	0.767	-	-	0.774	0.743	0.635	0.724	0.737	0.726	0.748	0.764	0.769	0.769	0.778	**0.781**	**0.782**
	$mIoU\uparrow$	0.684	-	-	0.689	0.668	0.386	0.647	0.661	0.652	0.671	0.696	0.703	0.697	0.704	**0.710**	**0.707**
Kvasir	$mDice\uparrow$	0.907	0.850	0.873	**0.915**	0.889	0.812	0.901	0.900	0.901	0.902	**0.917**	0.913	0.911	0.908	0.899	0.902
	$mIoU\uparrow$	0.855	0.776	0.806	**0.862**	0.818	0.618	0.827	0.828	0.834	0.838	**0.867**	0.861	0.859	0.860	0.844	0.847
ETIS	$mDice\uparrow$	0.670	-	-	0.757	0.688	0.445	0.685	0.693	0.680	0.732	0.780	0.749	0.784	**0.802**	**0.797**	0.750
	$mIoU\uparrow$	0.595	-	-	0.681	0.610	0.223	0.603	0.622	0.611	0.654	0.709	0.683	0.720	**0.737**	**0.730**	0.676

PolyP Detection: Tab. 5.5 presents quantitative results for the PolyP detection task. Compared to existing approaches[17, 18, 153, 154], our method achieves significant improvements across nearly all datasets. We observe further gains when using the

SAM2 backbone, especially with complex-valued adapters. While our performance is comparable to that of [154], it is important to emphasize that–unlike their task-specific model–our adaptive approach is designed to generalize across multiple low-level vision tasks.

$\mathbb{R}$-valued *vs* $\mathbb{C}$-valued Adapters: Tab. 5.2, 5.3, 5.4, and 5.5 show that, when using the SAM backbone, complex-valued adapters perform comparably to, or in some cases better than, their real-valued adapters. Furthermore, when switching to the SAM2 backbone, complex-valued adapters consistently yield improved results, highlighting the enhanced compatibility and representational capacity of SAM2 for this task.

Table 5.6: The results of our ablation study highlight the importance of adapters for efficient finetuning; it also highlights the importance of each low-frequency subband (*LL*) and the rest of the high-frequency subbands (*HL*, *LH*, *HH*) in the image in the wavelet domain. The ablation experiment is done for camouflage object detection, and SAM is taken as the backbone.

Method				CHAMELEON				CAMO				COD10K			
				$\mathcal{S}_m \uparrow$	$\mathcal{E}_\phi \uparrow$	$\mathcal{F}_\beta^w \uparrow$	$\mathcal{M} \downarrow$	$\mathcal{S}_m \uparrow$	$\mathcal{E}_\phi \uparrow$	$\mathcal{F}_\beta^w \uparrow$	$\mathcal{M} \downarrow$	$\mathcal{S}_m \uparrow$	$\mathcal{E}_\phi \uparrow$	$\mathcal{F}_\beta^w \uparrow$	$\mathcal{M} \downarrow$
SAM Finetuned				0.796	0.802	0.676	0.062	0.750	0.756	0.639	0.105	0.789	0.817	0.596	0.049
LL	*HL*	*LH*	*HH*	Evaluating each component											
✔	×	✗	×	0.907	0.937	0.842	0.029	**0.856**	0.884	0.785	0.062	0.891	0.929	0.817	0.024
×	✔	✗	×	0.895	0.927	0.834	0.030	0.851	0.882	0.781	0.063	0.887	0.924	0.816	0.024
×	×	✔	×	0.896	0.915	0.838	0.030	0.854	0.887	0.790	0.063	0.891	0.928	0.838	0.023
×	×	×	✔	0.908	0.935	0.848	0.028	0.853	0.883	0.783	0.064	0.893	0.928	0.818	0.023
×	✔	✔	✔	**0.923**	**0.952**	**0.890**	**0.022**	**0.856**	**0.888**	**0.807**	**0.060**	**0.903**	**0.939**	**0.857**	**0.019**

5.3.4 Ablation Study

We conduct three ablation studies to highlight our contribution. First, we remove the WHF module and adapters from the SAM backbone while keeping everything else the same, and then, we provide each of the wavelet transform subbands. Lastly, we replace the wavelets used in our approach with other wavelets and observe the result.

Effect of adapters: We remove high-frequency features and patch embeddings from the backbone and train the SAM model. Tab. 5.6 shows the result for the same. Although this helped improve performance over SAM, it lags significantly.

Effect of each subband: In §5.2.3, we claim that when combining all three high-frequency subbands (*HL*, *LH*, *HH*), there is a significant improvement in the performance. Hence, we perform an ablation study by considering each subband individually and observing its significance to assert our claim. As expected, we see performance degradation for all three datasets when provided with low-frequency features (*LL*). Similarly, when we provide each high-frequency subband, we see a similar pattern in performance loss. However, we only see the best performance when all three are combined.

Effect of each wavelet: To ensure that the results obtained are not due to any specific property of a single wavelet, we use four different wavelets and show their results. The comparative result are shown in Tab. 5.2, 5.3, 5.4, 5.5. The observation is simi-

lar across all four tasks; we see more performance gain for some wavelets, but overall, using any wavelet provides better information than previous methods.

5.4 Summary

This chapter presents a new efficient fine-tuning method for large vision models, particularly SAM and SAM2, for solving four difficult vision tasks. Additionally, we show that a complex-valued adapter can be used to fine-tune large vision models efficiently. We first show that the existing method of providing high-frequency information to the network is not optimal and needs improvement. We propose a WHF module, which converts the input image to a wavelet domain and combines the high-frequency subbands. When provided to the networks, this new feature map significantly improves over existing methods.

CHAPTER 6

Conclusion & Future Work

6.1 Overview

This thesis has explored the design and application of complex-valued neural networks (iCNNs) in computer vision, addressing the long-standing challenge of integrating the representational richness of complex numbers into mainstream deep learning architectures in computer vision. Beginning with the development of foundational tools for complex-valued processing and culminating in the adaptation of large pre-trained vision models through complex-valued mechanisms, the work presented here demonstrates that complex-valued learning is not only theoretically elegant but also practically advantageous.

Across the three major contributions, Fully Complex-valued Convolutional Networks (FCCNs), Deep Complex Spatio-Spectral Networks (DCS2Net), and SAMWave, the thesis systematically developed methods to overcome the inherent barriers of complex-valued learning: the lack of complex visual inputs, challenges in network design, and absence of suitable loss functions and optimization strategies. Collectively, these contributions mark a significant step toward establishing complex-valued computation/learning as a general-purpose framework for modern computer vision.

The following sections summarize the key findings, discuss their implications, and outline future research directions that can further expand the impact of this work.

6.2 Summary of Contributions

6.2.1 Fully Complex-valued Convolutional Networks (FCCNs)

The first part of this thesis addressed the representation problem, how to effectively convert real-valued visual data into meaningful complex-valued inputs and maintain a consistent flow of complex-valued information throughout a neural network.

The proposed iHSV color model offered a novel real-to-complex conversion method. By mapping the HSV color space into three orthogonal argand planes, iHSV embeds

perceptually consistent color information in both magnitude and phase components. This approach overcomes the limitations of earlier pseudo-complex encodings such as simple $[R+iG,\ G+iB]$ or problematic learning of imaginary part from the real image.

Building upon this representation, the FCCN architecture was designed as the first end-to-end complex-valued convolutional network for vision tasks. FCCNs replace real-valued fully connected layers with complex 1×1 convolutions, ensuring that every operation, from input to loss, is executed in the complex domain. The introduction of a differentiable complex-valued loss function, derived from the cross-entropy loss, further ensured learning entirely in the complex domain.

Experiments on multiple classification datasets confirmed that FCCNs consistently outperform existing complex-valued and real-valued baselines, validating the hypothesis that fully complex-valued processing yields richer and more generalizable representations. This contribution laid the mathematical and algorithmic groundwork for subsequent studies on complex-valued deep learning in computer vision.

6.2.2 Deep Complex Spatio-Spectral Networks (DCS^{2}Net)

While FCCNs established the feasibility of complex-valued learning in the spatial domain, the second study extended the framework to incorporate frequency domain reasoning, a crucial aspect of visual understanding.

Additionally, one key innovation here was the R2C (Real-to-Complex) transformation, a compact, invertible mapping from RGB to two complex channels representing intensity (achromatic) and chromatic (color) information. Unlike previous encodings, R2C preserves geometric consistency and allows reconstruction of the original image, providing both interpretability and efficiency.

Leveraging this representation, the thesis introduced DCS^{2}Net (Deep Complex Spatio-Spectral Network), the first token-based iCNN architecture designed specifically for vision tasks. DCS2Net's spatio-spectral modules operate in both spatial and Fourier domains using learnable complex filters, capturing long-range contextual cues alongside fine-grained spatial details.

To support this end-to-end complex pipeline, a new objective function was formulated. It uses binary ground truths encoded using 1-i encoding, aligning perfectly with the inherent orthogonality between foreground and background and enabling direct comparison between predicted and true complex representations.

Extensive evaluations demonstrated that DCS^{2}Net achieves state-of-the-art results (amongst complex-valued neural networks) on several binary segmentation benchmarks, including salient object detection, shadow detection, and defocus blur detection. These results highlight the potential of complex-valued processing for dense prediction tasks, where subtle phase and frequency relationships strongly influence the performance.

6.2.3 SAMWave: Adapting Foundation Models using Complex-valued Wavelets

The final contribution translated the insights from the previous studies into the context of large-scale vision foundation models, focusing on the adaptation of the Segment Anything Model (SAM).

While SAM represents a major leap in vision model generalization, its performance degrades on low-level or challenging segmentation tasks such as camouflage, shadow, polyp, and defocus blur detection, where subtle textures and high-frequency cues play a critical role. Traditional adaptation methods based on fixed Fourier masks offer limited improvement due to their static nature and inability to capture spatial locality.

This thesis introduced SAMWave, a novel adaptation framework that enhances SAM through wavelet-based high-frequency feature injection. The Wavelet High-Frequency (WHF) module extracts high-frequency components using the discrete wavelet transform (DWT) and integrates them into SAM's encoder via lightweight adapters.

Crucially, SAMWave extends this idea to the complex domain by incorporating the dual-tree complex wavelet transform (DT-CWT) and complex-valued adapters. This design allows the model to process both magnitude and phase information of high-frequency features, leading to better discrimination of fine structures and edges.

Empirical results demonstrated substantial performance gains across multiple datasets and tasks, with improvements consistent across both SAM and SAM2 backbones. The findings underscore the scalability and versatility of complex-valued processing, not only for compact networks but also for massive pre-trained vision systems.

6.2.4 Quantitative Synthesis of Performance Gains

Across all three methodological contributions, our complex-valued formulations demonstrated consistent, measurable improvements over both standard real-valued baselines and legacy complex architectures. A quantitative recap of these empirical gains illustrates the practical trajectory of this thesis:

- **Foundational Spatial Processing (FCCNs):** By replacing hybrid real-complex pipelines with an end-to-end complex flow and introducing the iHSV color model, FCCNs established a scalable baseline that allowed complex-valued convolutional networks to be successfully trained and evaluated on ImageNet-scale datasets for the first time, consistently outperforming legacy complex architectures such as Deep Complex Networks (DCN) across standard classification accuracy metrics.
- **Dense Spatio-Spectral Prediction (DCS^{2}Net):** On real-valued benchmark datasets (Salient Object, Defocus Blur, and Shadow Detection), DCS^{2}Net achieved top-two performance against 10 state-of-the-art real-valued models while outperforming all comparative complex-valued methods. When evaluated on natively complex visual data (the InSAR Building dataset), DCS^{2}Net advanced the state-of-

the-art mean Intersection over Union (mIoU) from 0.86 (SCVUNet) to 0.91, while reducing computational complexity to just 25 gFLOPS and 32 ms inference time, less than half the computational overhead and memory footprint of preceding complex networks.

- **Foundation Model Adaptation (SAMWave):** On complex, boundary-critical segmentation tasks such as Camouflaged Object Detection (COD), our Dual-Tree Complex Wavelet (DT-CWT) adapters substantially enhanced the frozen Segment Anything Model (SAM). Across benchmark datasets (CHAMELEON, CAMO, and COD10K), SAMWave consistently boosted structure-measure scores ($\mathcal{S}_m$) to peak levels exceeding 0.95–0.96 while reducing Mean Absolute Error ($\mathscr{M}$) down to 0.006–0.007. These results confirm that injecting multi-directional phase information into large vision transformers yields state-of-the-art boundary precision with minimal parameter overhead.

6.3 Limitations of the Proposed Methods

While the empirical evaluations and theoretical proofs presented in this thesis confirm the strong representational capacity of complex-valued neural networks, a comprehensive assessment requires acknowledging several persistent methodological and practical limitations:

1. **Hardware-Level Computational Overhead:** At the mathematical level, a single complex multiplication involves four real multiplications and two real additions ($z_1 z_2 = (x_1 x_2 - y_1 y_2) + i(x_1 y_2 + y_1 x_2)$). Although architectures like DCS^{2}Net and SAMWave successfully mitigate parameter count and theoretical FLOPs via token compression, channel halving, and bottleneck adapters, modern GPU hardware lacks natively optimized Tensor Core support for complex arithmetic. Consequently, complex-valued operations must be simulated using expanded real-valued matrix multiplications, which increases memory bandwidth consumption and wall-clock training times compared to heavily optimized FP16/BF16 real-valued pipelines.

2. **Phase Wrapping and Optimization Sensitivity:** As analyzed in our formulation of the complex cross-entropy loss ($\mathscr{L}_{comp}$), mapping cyclical phase angles ($\theta \in (0, 2\pi]$) onto a linear objective space inherently creates a mathematical branch cut discontinuity at the $2\pi \equiv 0$ boundary. While this mechanism effectively drives network convergence toward the target real axis, it can introduce transient gradient instability during early training iterations. Consequently, training deep complex architectures requires careful learning rate scheduling and warmup phases to prevent erratic loss spikes around phase discontinuities.

3. **The Real-to-Complex Ingestion Penalty:** Because mainstream computer vision benchmarks (e.g., ImageNet, COD10K, CAMO) consist entirely of natively real-

valued RGB images, our pipelines must always incur an initial "ingestion penalty" to convert real visual signals into the complex domain. While our proposed iHSV, R2C, and wavelet transformations successfully establish geometrically rigorous entry points, the theoretical ceiling of iCNNs remains partially constrained by the absence of large-scale, natively phase-rich visual datasets (such as optical interferometry or raw holographic video benchmarks).

6.4 Future Work

The promising results from this thesis open several avenues for further exploration:

(a) Complex-valued Multimodal Models

Complex-valued representations could provide an elegant mechanism to model alignment between visual and linguistic modalities. Extending iHSV, R2C, and complex adapters to vision-language [116] or audio-visual [42] models could lead to more expressive multimodal systems.

(b) Complex-valued Generative and Diffusion Models

Incorporating phase-sensitive loss functions and complex-valued latent spaces in GANs [43] or diffusion models [58] could improve texture fidelity and structural coherence. Developing complex-valued diffusion processes and latent priors is a natural next step.

(c) Self-supervised and Semi-supervised Complex-valued Learning

Given their representational richness, iCNNs can benefit from self-supervised objectives [181] based on phase-consistency or spectral similarity, enabling learning from large unlabeled corpora.

(d) Extensions to Foundation Models

Building upon SAMWave, future work may generalize complex-valued adapters to other foundation models such as vision transformers, diffusion-based segmenters, and multimodal pre-trained systems.

6.5 Concluding Remarks

This thesis presented a unified framework for complex-valued neural networks in computer vision, bridging theoretical innovation with practical implementation. Starting

from the fundamental challenge of representing real-valued images in the complex domain, it developed a suite of models and techniques from fully complex convolutional networks to large-model adaptation strategies, that together demonstrate the feasibility and advantages of complex-valued learning.

The contributions not only validate the potential of complex-valued computation/learning but also establish it as a scalable and generalizable paradigm for next-generation vision systems. As deep learning advances toward larger and more multimodal architectures, the integration of complex-valued reasoning offers a powerful avenue for enhancing representational depth and perceptual fidelity.

In conclusion, this work represents a significant step toward the broader vision of complex-valued (or hypercomplex) AI where neural systems operate with the full richness of complex arithmetic to better capture the multidimensional nature of perception, reasoning, and understanding in both human and machine vision.

Bibliography

[1] Igor Aizenberg, Naum Aizenberg, Taras Bregin, Constantine Butakov, Elya Farberov, Nickolai Merzlyakov, and O Milukova. Blur recognition on the neural network based on multi-valued neurons. *Journal of Image and Graphics*, 5:127–130, 2000.

[2] Igor N. Aizenberg, Alan Ordukhanov, and Fionntan O'Boy. MLMVN as an intelligent image filter. In *2017 International Joint Conference on Neural Networks, IJCNN 2017, Anchorage, AK, USA, May 14-19, 2017*, pages 3106–3113. IEEE, 2017.

[3] Igor N. Aizenberg, Dmitriy Paliy, Jacek M. Zurada, and Jaakko Astola. Blur identification by multilayer neural network based on multivalued neurons. *IEEE Trans. Neural Networks*, 19(5):883–898, 2008.

[4] Martin Arjovsky, Amar Shah, and Yoshua Bengio. Unitary evolution recurrent neural networks. In *International Conference on Machine Learning*, pages 1120–1128. PMLR, 2016.

[5] Reza Mohammadi Asiyabi, Mihai Datcu, Andrei Anghel, and Holger Nies. Complex-valued autoencoders with coherence preservation for sar. In *EUSAR 2022; 14th European Conference on Synthetic Aperture Radar*, pages 1–6. VDE, 2022.

[6] Sara Babakniya, Zalan Fabian, Chaoyang He, Mahdi Soltanolkotabi, and Salman Avestimehr. A data-free approach to mitigate catastrophic forgetting in federated class incremental learning for vision tasks. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[7] Jorge Bernal, Francisco Javier Sánchez, Gloria Fernández-Esparrach, Debora Gil, Cristina Rodríguez de Miguel, and Fernando Vilariño. WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. *Comput. Medical Imaging Graph.*, 43:99–111, 2015.

[8] David Brüggemann, Menelaos Kanakis, Anton Obukhov, Stamatios Georgoulis, and Luc Van Gool. Exploring relational context for multi-task dense prediction. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 15849–15858. IEEE, 2021.

[9] Christopher P. Burgess, Loïc Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matthew M. Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation. *CoRR*, abs/1901.11390, 2019.

[10] Xinzi Cao, Xiawu Zheng, Guanhong Wang, Weijiang Yu, Yunhang Shen, Ke Li, Yutong Lu, and Yonghong Tian. Solving the catastrophic forgetting problem in generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16880–16889, June 2024.

[11] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I*, volume 12346 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2020.

[12] 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.

[13] Guangyao Chen, Peixi Peng, Li Ma, Jia Li, Lin Du, and Yonghong Tian. Amplitude-phase recombination: Rethinking robustness of convolutional neural networks in frequency domain. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 458–467, 2021.

[14] Jiankun Chen. Simulated insar building dataset for cvcmff net, 2020.

[15] Jinpeng Chen, Runmin Cong, Yuxuan Luo, Horace Ho-Shing Ip, and Sam Kwong. Saving 100x storage: Prototype replay for reconstructing training sample distribution in class-incremental semantic segmentation. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[16] Sizhe Chen, Haipeng Wang, Feng Xu, and Ya-Qiu Jin. Target classification using the deep convolutional networks for SAR images. *IEEE Trans. Geosci. Remote. Sens.*, 54(8):4806–4817, 2016.

[17] Tianrun Chen, Ankang Lu, Lanyun Zhu, Chaotao Ding, Chunan Yu, Deyi Ji, Zejian Li, Lingyun Sun, Papa Mao, and Ying Zang. Sam2-adapter: Evaluating & adapting segment anything 2 in downstream tasks: Camouflage, shadow, medical image segmentation, and more, 2024.

[18] Tianrun Chen, Lanyun Zhu, Chaotao Ding, Runlong Cao, Shangzhan Zhang, Yan Wang, Zejian Li, Lingyun Sun, Papa Mao, and Ying Zang. Sam fails to segment

anything? – sam-adapter: Adapting sam in underperformed scenes: Camouflage, shadow, and more, 2023.

[19] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[20] Zhihao Chen, Lei Zhu, Liang Wan, Song Wang, Wei Feng, and Pheng-Ann Heng. A multi-task mean teacher for semi-supervised shadow detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 5610–5619. Computer Vision Foundation / IEEE, 2020.

[21] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. 2021.

[22] Brian Cheung, Alexander Terekhov, Yubei Chen, Pulkit Agrawal, and Bruno Olshausen. Superposition of many models into one. *Advances in neural information processing systems*, 32, 2019.

[23] Hyeong-Seok Choi, Jang-Hyun Kim, Jaesung Huh, Adrian Kim, Jung-Woo Ha, and Kyogu Lee. Phase-aware speech enhancement with deep complex u-net. In *International Conference on Learning Representations*, 2018.

[24] Hyeong-Seok Choi, Jang-Hyun Kim, Jaesung Huh, Adrian Kim, Jung-Woo Ha, and Kyogu Lee. Phase-aware speech enhancement with deep complex u-net. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[25] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

[26] Elizabeth Cole, Joseph Cheng, John Pauly, and Shreyas Vasanawala. Analysis of deep complex-valued convolutional neural networks for mri reconstruction and phase-focused applications. *Magnetic resonance in medicine*, 86(2):1093–1109, 2021.

[27] Xiaodong Cun and Chi-Man Pun. Image splicing localization via semi-global network and fully connected conditional random fields. In Laura Leal-Taixé and Stefan Roth, editors, *Computer Vision - ECCV 2018 Workshops - Munich, Germany, September 8-14, 2018, Proceedings, Part II*, volume 11130 of *Lecture Notes in Computer Science*, pages 252–266. Springer, 2018.

[28] Ivo Danihelka, Greg Wayne, Benigno Uria, Nal Kalchbrenner, and Alex Graves. Associative long short-term memory. In *International Conference on Machine Learning*, pages 1986–1994. PMLR, 2016.

[29] Pieter-Tjerk de Boer, Dirk P. Kroese, Shie Mannor, and Reuven Y. Rubinstein. A tutorial on the cross-entropy method. *Ann. Oper. Res.*, 134(1):19–67, 2005.

[30] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[31] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[32] Yihong Dong, Ying Peng, Muqiao Yang, Songtao Lu, and Qingjiang Shi. Signal transformer: Complex-valued attention and meta-learning for signal recognition. *CoRR*, abs/2106.04392, 2021.

[33] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[34] Florian Eilers and Xiaoyi Jiang. Building blocks for a complex-valued transformer architecture. In *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023, Rhodes Island, Greece, June 4-10, 2023*, pages 1–5. IEEE, 2023.

[35] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 4558–4567. IEEE Computer Society, 2017.

[36] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[37] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. In Jérôme Lang, editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 698–704. ijcai.org, 2018.

[38] Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[39] Jessica J. Fridrich and Jan Kodovský. Rich models for steganalysis of digital images. *IEEE Trans. Inf. Forensics Secur.*, 7(3):868–882, 2012.

[40] Jingkun Gao, Bin Deng, Yuliang Qin, Hongqiang Wang, and Xiang Li. Enhanced radar imaging using a complex-valued convolutional neural network. *IEEE Geoscience and Remote Sensing Letters*, 16(1):35–39, 2018.

[41] George M Georgiou and Cris Koutsougeras. Complex domain backpropagation. *IEEE transactions on Circuits and systems II: analog and digital signal processing*, 39(5):330–334, 1992.

[42] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *CVPR*, 2023.

[43] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Commun. ACM*, 63(11):139–144, October 2020.

[44] Eleonora Grassucci, Aston Zhang, and Danilo Comminiello. Phnns: Lightweight neural networks via parameterized hypercomplex convolutions. *IEEE Trans. Neural Networks Learn. Syst.*, 35(6):8293–8305, 2024.

[45] Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks. *CoRR*, abs/2012.05208, 2020.

[46] Shenshen Gu and Lu Ding. A complex-valued vgg network based deep learing algorithm for image recognition. In *2018 Ninth International Conference on Intelligent Control and Information Processing (ICICIP)*, pages 340–343. IEEE, 2018.

[47] Avi Gupta, Koteswar Rao Jerripothula, and Tammam Tillo. CIRCOD: co-saliency inspired referring camouflaged object discovery. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2025, Tucson, AZ, USA, February 26 - March 6, 2025*, pages 8313–8323. IEEE, 2025.

[48] Mhd Modar Halimeh and Walter Kellermann. Complex-valued spatial autoencoders for multichannel speech enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, pages 261–265. IEEE, 2022.

[49] Daichi Hayakawa, Takashi Masuko, and Hiroshi Fujimura. Applying complex-valued neural networks to acoustic modeling for speech recognition. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1725–1731. IEEE, 2018.

[50] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[51] Alvin Heng and Harold Soh. Selective amnesia: A continual learning approach to forgetting in deep generative models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[52] Geoffrey E. Hinton, Sara Sabour, and Nicholas Frosst. Matrix capsules with EM routing. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[53] A. Hirose. Proposal of fully complex-valued neural networks. In *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, volume 4, pages 152–157 vol.4, June 1992.

[54] Akira Hirose. Applications of complex-valued neural networks to coherent optical computing using phase-sensitive detection scheme. *Information Sciences-Applications*, 2(2):103–117, 1994.

[55] Akira Hirose. *Complex-valued neural networks*. Springer, 2012.

[56] Akira Hirose and Shotaro Yoshida. Comparison of complex-and real-valued feedforward neural networks in their generalization ability. In *International Conference on Neural Information Processing*, pages 526–531. Springer, 2011.

[57] Akira Hirose and Shotaro Yoshida. Generalization characteristics of complex-valued feedforward neural networks in relation to signal coherence. *IEEE Transactions on Neural Networks and Learning Systems*, 23(4):541–551, 2012.

[58] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[59] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR, 2019.

[60] Xiaowei Hu, Lei Zhu, Chi-Wing Fu, Jing Qin, and Pheng-Ann Heng. Direction-aware spatial context features for shadow detection. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7454–7462. Computer Vision Foundation / IEEE Computer Society, 2018.

[61] Yanxin Hu, Yun Liu, Shubo Lv, Mengtao Xing, Shimin Zhang, Yihui Fu, Jian Wu, Bihong Zhang, and Lei Xie. Dccrn: Deep complex convolution recurrent network for phase-aware speech enhancement. In *Interspeech*, 2020.

[62] Zhou Huang, Hang Dai, Tian-Zhu Xiang, Shuo Wang, Huai-Xin Chen, Jie Qin, and Huan Xiong. Feature shrinkage pyramid for camouflaged object detection with transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 5557–5566. IEEE, 2023.

[63] Debesh Jha, Pia H. Smedsrud, Michael A. Riegler, Pål Halvorsen, Thomas de Lange, Dag Johansen, and Håvard D. Johansen. Kvasir-seg: A segmented polyp dataset. In Yong Man Ro, Wen-Huang Cheng, Junmo Kim, Wei-Ta Chu, Peng Cui, Jung-Woo Choi, Min-Chun Hu, and Wesley De Neve, editors, *MultiMedia Modeling - 26th International Conference, MMM 2020, Daejeon, South Korea, January 5-8, 2020, Proceedings, Part II*, volume 11962 of *Lecture Notes in Computer Science*, pages 451–462. Springer, 2020.

[64] Lai Jiang, Zhe Wang, Mai Xu, and Zulin Wang. Image saliency prediction in transformed domain: A deep complex neural network method. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8521–8528, 2019.

[65] Lai Jiang, Zhe Wang, Mai Xu, and Zulin Wang. Image saliency prediction in transformed domain: A deep complex neural network method. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 8521–8528. AAAI Press, 2019.

[66] Lai Jiang, Mai Xu, Shanyi Zhang, and Leonid Sigal. Deepct: A novel deep complex-valued network with learnable transform for video saliency prediction. *Pattern Recognition*, 102:107234, 2020.

[67] Ting Jiang, Zongyong Cui, Zhi Zhou, and Zongjie Cao. Data augmentation with gabor filter in deep convolutional neural networks for sar target recognition. In *2018 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2018, Valencia, Spain, July 22-27, 2018*, pages 689–692. IEEE, 2018.

[68] Mario Jojoa, Begonya Garcia-Zapirain, and Winston Percybrooks. A fair performance comparison between complex-valued and real-valued neural networks for disease detection. *Diagnostics*, 12(8):1893, 2022.

[69] Ali Karaali and Cláudio R. Jung. Edge-based defocus blur estimation with adaptive scale selection. *IEEE Trans. Image Process.*, 27(3):1126–1137, 2018.

[70] Taehwan Kim and Tülay Adalı. Approximation by fully complex multilayer perceptrons. *Neural computation*, 15(7):1641–1666, 2003.

[71] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[72] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 3992–4003. IEEE, 2023.

[73] Kurt Koffka. *Principles of Gestalt psychology*, volume 44. Routledge, 2013.

[74] Wolfgang Köhler. Gestalt psychology. *Psychologische Forschung*, 31(1):XVIII–XXX, 1967.

[75] Cansu Korkmaz, A. Murat Tekalp, and Zafer Dogan. Training generative image super-resolution models by wavelet-domain losses enables better control of artifacts. *CoRR*, abs/2402.19215, 2024.

[76] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[77] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research).

[78] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-100 (canadian institute for advanced research).

[79] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[80] Trung-Nghia Le, Tam V. Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. *Journal of Computer Vision and Image Understanding*, 184:45–56, 2019.

[81] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

[82] Aixuan Li, Jing Zhang, Yunqiu Lv, Bowen Liu, Tong Zhang, and Yuchao Dai. Uncertainty-aware joint salient object and camouflaged object detection. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 10071–10081. Computer Vision Foundation / IEEE, 2021.

[83] Aixuan Li, Jing Zhang, Yunqiu Lyu, Bowen Liu, Tong Zhang, and Yuchao Dai. Uncertainty-aware joint salient object and camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[84] Guanbin Li and Yizhou Yu. Visual saliency based on multiscale deep features. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 5455–5463. IEEE Computer Society, 2015.

[85] Jin Li, Yaoming Wang, Xiaopeng Zhang, Bowen Shi, Dongsheng Jiang, Chenglin Li, Wenrui Dai, Hongkai Xiong, and Qi Tian. Ailurus: A scalable vit framework for dense prediction. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[86] Qiuchi Li, Sagar Uprety, Benyou Wang, and Dawei Song. Quantum-inspired complex word embedding. In Isabelle Augenstein, Kris Cao, He He, Felix Hill, Spandana Gella, Jamie Kiros, Hongyuan Mei, and Dipendra Misra, editors, *Proceedings of The Third Workshop on Representation Learning for NLP, Rep4NLP*, 2018.

[87] Wei-Hong Li, Xialei Liu, and Hakan Bilen. Learning multiple dense prediction tasks from partially annotated data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 18857–18867. IEEE, 2022.

[88] Yanghao Li, Hanzi Mao, Ross B. Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part IX*, volume 13669 of *Lecture Notes in Computer Science*, pages 280–296. Springer, 2022.

[89] Yin Li, Xiaodi Hou, Christof Koch, James M. Rehg, and Alan L. Yuille. The secrets of salient object segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 280–287. IEEE Computer Society, 2014.

[90] Jiaying Lin, Xin Tan, Ke Xu, Lizhuang Ma, and Rynson W. H. Lau. Frequency-aware camouflaged object detection. *ACM Trans. Multim. Comput. Commun. Appl.*, 19(2):61:1–61:16, 2023.

[91] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A simple pooling-based design for real-time salient object detection. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 3917–3926. Computer Vision Foundation / IEEE, 2019.

[92] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Learning pixel-wise contextual attention for saliency detection. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3089–3098. Computer Vision Foundation / IEEE Computer Society, 2018.

[93] Nian Liu, Ni Zhang, Kaiyuan Wan, Ling Shao, and Junwei Han. Visual saliency transformer. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 4702–4712. IEEE, 2021.

[94] Weihuang Liu, Xi Shen, Chi-Man Pun, and Xiaodong Cun. Explicit visual prompting for low-level structure segmentations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 19434–19445. IEEE, 2023.

[95] Zhen Liu, Baochang Zhang, and Guodong Guo. Few-shot learning with complex-valued neural networks. In *31st British Machine Vision Conference 2020, BMVC 2020, Virtual Event, UK, September 7-10, 2020*. BMVA Press, 2020.

[96] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015.

[97] Ange Lou, Benjamin Planche, Zhongpai Gao, Yamin Li, Tianyu Luan, Hao Ding, Terrence Chen, Jack Noble, and Ziyan Wu. Darenerf: Direction-aware representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5031–5042, June 2024.

[98] Sindy Löwe, Phillip Lippe, Maja Rudolph, and Max Welling. Complex-valued autoencoders for object discovery. *Transactions on Machine Learning Research*, 2022.

[99] Sindy Löwe, Phillip Lippe, Maja Rudolph, and Max Welling. Complex-valued autoencoders for object discovery. *Trans. Mach. Learn. Res.*, 2022, 2022.

[100] Yuxiang Lu, Shalayiding Sirejiding, Yue Ding, Chunlin Wang, and Hongtao Lu. Prompt guided transformer for multi-task dense prediction. *IEEE Trans. Multim.*, 26:6375–6385, 2024.

[101] Yunqiu Lv, Jing Zhang, Yuchao Dai, Aixuan Li, Bowen Liu, Nick Barnes, and Deng-Ping Fan. Simultaneously localize, segment and rank the camouflaged objects. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 11591–11601. Computer Vision Foundation / IEEE, 2021.

[102] Yunqiu Lyu, Jing Zhang, Yuchao Dai, Aixuan Li, Bowen Liu, Nick Barnes, and Deng-Ping Fan. Simultaneously localize, segment and rank the camouflaged objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[103] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[104] Haiyang Mei, Ge-Peng Ji, Ziqi Wei, Xin Yang, Xiaopeng Wei, and Deng-Ping Fan. Camouflaged object segmentation with distraction mining. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 8772–8781. Computer Vision Foundation / IEEE, 2021.

[105] Haiyang Mei, Ge-Peng Ji, Ziqi Wei, Xin Yang, Xiaopeng Wei, and Deng-Ping Fan. Camouflaged object segmentation with distraction mining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8772–8781, June 2021.

[106] Minami Miyauchi, Masatoshi Seki, Akira Watanabe, and Arata Miyauchi. Interpretation of optical flow through complex neural network. In José Mira, Joan Cabestany, and Alberto Prieto, editors, *New Trends in Neural Computation, International Workshop on Artificial Neural Networks, IWANN '93, Sitges, Spain, June 9-11, 1993, Proceedings*, volume 686 of *Lecture Notes in Computer Science*, pages 645–650. Springer, 1993.

[107] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. *NIPS workshop on deep learning and unsupervised feature learning*, 2011.

[108] Kien Nguyen, Clinton Fookes, Sridha Sridharan, and Arun Ross. Complex-valued iris recognition network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):182–196, 2022.

[109] Maximilian Nickel, Lorenzo Rosasco, and Tomaso A. Poggio. Holographic embeddings of knowledge graphs. In Dale Schuurmans and Michael P. Wellman, editors, *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence,*

February 12-17, 2016, Phoenix, Arizona, USA, pages 1955–1961. AAAI Press, 2016.

[110] Alan V Oppenheim and Jae S Lim. The importance of phase in signals. *Proceedings of the IEEE*, 69(5):529–541, 1981.

[111] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024, 2024.

[112] Timothy J. O'Shea, Johnathan Corgan, and T. Charles Clancy. Convolutional radio modulation recognition networks. In Chrisina Jayne and Lazaros S. Iliadis, editors, *Engineering Applications of Neural Networks - 17th International Conference, EANN 2016, Aberdeen, UK, September 2-5, 2016, Proceedings*, volume 629 of *Communications in Computer and Information Science*, pages 213–226. Springer, 2016.

[113] Youwei Pang, Xiaoqi Zhao, Tian-Zhu Xiang, Lihe Zhang, and Huchuan Lu. Zoom in and out: A mixed-scale triplet network for camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2160–2170, June 2022.

[114] Youwei Pang, Xiaoqi Zhao, Lihe Zhang, and Huchuan Lu. Multi-scale interactive network for salient object detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9410–9419. Computer Vision Foundation / IEEE, 2020.

[115] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jägersand. Basnet: Boundary-aware salient object detection. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 7479–7489. Computer Vision Foundation / IEEE, 2019.

[116] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

[117] Michaël Ramamonjisoa, Michael Firman, Jamie Watson, Vincent Lepetit, and Daniyar Turmukhambetov. Single image depth prediction with wavelet decomposition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 11089–11098. Computer Vision Foundation / IEEE, 2021.

[118] Yuval Ran-Milo, Eden Lumbroso, Edo Cohen-Karlik, Raja Giryes, Amir Globerson, and Nadav Cohen. Provable benefits of complex parameterizations for structured state space models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

[119] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 12159–12168. IEEE, 2021.

[120] Yongming Rao, Wenliang Zhao, Zheng Zhu, Jiwen Lu, and Jie Zhou. Global filter networks for image classification. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 980–993, 2021.

[121] Yongming Rao, Wenliang Zhao, Zheng Zhu, Jie Zhou, and Jiwen Lu. Gfnet: Global filter networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(9):10960–10973, 2023.

[122] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[123] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloé Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross B. Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

[124] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016.

[125] David P. Reichert and Thomas Serre. Neuronal synchrony in complex-valued deep networks. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[126] Qinghua Ren, Shijian Lu, Jinxia Zhang, and Renjie Hu. Salient object detection by fusing local and global contexts. *IEEE Transactions on Multimedia*, 23:1442–1453, 2021.

[127] Timothy D Ross, Steven W Worrell, Vincent J Velten, John C Mossing, and Michael Lee Bryant. Standard sar atr evaluation experiments using the mstar public release data set. In *Algorithms for Synthetic Aperture Radar Imagery V*, volume 3370, pages 566–573. SPIE, 1998.

[128] Vishwanath Saragadam, Daniel LeJeune, Jasper Tan, Guha Balakrishnan, Ashok Veeraraghavan, and Richard G. Baraniuk. Wire: Wavelet implicit neural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18507–18516, June 2023.

[129] Andy M. Sarroff. *Complex Neural Networks for Audio*. PhD thesis, Dartmouth College, USA, 2018.

[130] Theresa Scarnati and Benjamin Lewis. Complex-valued neural networks for synthetic aperture radar image classification. In *2021 IEEE Radar Conference (RadarConf21)*, pages 1–6, 2021.

[131] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[132] Vanshali Sharma, Abhishek Kumar, Debesh Jha, Manas Kamal Bhuyan, Pradip K. Das, and Ulas Bagci. Controlpolypnet: Towards controlled colon polyp synthesis for improved polyp segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024 - Workshops, Seattle, WA, USA, June 17-18, 2024*, pages 2325–2334. IEEE, 2024.

[133] Jianping Shi, Li Xu, and Jiaya Jia. Discriminative blur detection features. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 2965–2972. IEEE Computer Society, 2014.

[134] Jianping Shi, Li Xu, and Jiaya Jia. Just noticeable defocus blur detection and estimation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 657–665. IEEE Computer Society, 2015.

[135] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 2990–2999, 2017.

[136] Juan Silva, Aymeric Histace, Olivier Romain, Xavier Dray, and Bertrand Granado. Toward embedded detection of polyps in wce images for early diagnosis of colorectal cancer. *International journal of computer assisted radiology and surgery*, 9(2):283—293, March 2014.

[137] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[138] 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*, pages 681–690, 2022.

[139] Przemyslaw Skurowski, Hassan Abdulameer, J Blaszczyk, Tomasz Depta, Adam Kornacki, and P Koziel. Animal camouflage analysis: Chameleon database. *Unpublished manuscript*, 2(6):7, 2018.

[140] Xiang Song, Kuang Shu, Songlin Dong, Jie Cheng, Xing Wei, and Yihong Gong. Overcoming catastrophic forgetting for multi-label class-incremental learning. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2024, Waikoloa, HI, USA, January 3-8, 2024*, pages 2378–2387. IEEE, 2024.

[141] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks. *arXiv preprint arXiv:1505.00387*, 2015.

[142] Aleksandar Stanic, Anand Gopalakrishnan, Kazuki Irie, and Jürgen Schmidhuber. Contrastive training of complex-valued autoencoders for object discovery. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[143] Elias M Stein and Rami Shakarchi. *Complex analysis*, volume 2. Princeton University Press, 2010.

[144] Darko Stern, Christian Payer, Vincent Lepetit, and Martin Urschler. Automated age estimation from hand MRI volumes using deep learning. In Sébastien Ourselin, Leo Joskowicz, Mert R. Sabuncu, Gözde B. Ünal, and William

M. Wells III, editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2016 - 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II*, volume 9901 of *Lecture Notes in Computer Science*, pages 194–202, 2016.

[145] Gilbert Strang and Truong Q. Nguyen. *Wavelets and filter banks*. Wellesley-Cambridge Press, 1997.

[146] Nima Tajbakhsh, Suryakanth R. Gurudu, and Jianming Liang. Automated polyp detection in colonoscopy videos using shape and context information. *IEEE Trans. Medical Imaging*, 35(2):630–644, 2016.

[147] Gouhei Tanaka and Kazuyuki Aihara. Complex-valued multistate associative memory with nonlinear multilevel functions for gray-level image reconstruction. *IEEE Trans. Neural Networks*, 20(9):1463–1473, 2009.

[148] Chang Tang, Xinzhong Zhu, Xinwang Liu, Lizhe Wang, and Albert Y. Zomaya. Defusionnet: Defocus blur detection via recurrently fusing and refining multi-scale deep features. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2700–2709. Computer Vision Foundation / IEEE, 2019.

[149] Chor Shen Tay, Ken Tanizawa, and Akira Hirose. Error reduction in holographic movies using a hybrid learning method in coherent neural networks. In Joaquim Marques de Sá, Luís A. Alexandre, Wlodzislaw Duch, and Danilo P. Mandic, editors, *Artificial Neural Networks - ICANN 2007, 17th International Conference, Porto, Portugal, September 9-13, 2007, Proceedings, Part I*, volume 4668 of *Lecture Notes in Computer Science*, pages 884–893. Springer, 2007.

[150] Hugo Touvron, Piotr Bojanowski, Mathilde Caron, Matthieu Cord, Alaaeldin El-Nouby, Edouard Grave, Gautier Izacard, Armand Joulin, Gabriel Synnaeve, Jakob Verbeek, et al. Resmlp: Feedforward networks for image classification with data-efficient training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):5314–5321, 2022.

[151] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, Joao Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J Pal. Deep complex networks. In *International Conference on Learning Representations*, 2018.

[152] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, João Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J. Pal. Deep complex networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[153] Quoc-Huy Trinh, Nhat-Tan Bui, Trong-Hieu Nguyen Mau, Minh-Van Nguyen, Hai-Minh Phan, Minh-Triet Tran, and Hai-Dang Nguyen. M2unet: Metaformer multi-scale upsampling network for polyp segmentation. In *31st European Signal Processing Conference, EUSIPCO 2023, Helsinki, Finland, September 4-8, 2023*, pages 1115–1119. IEEE, 2023.

[154] Quoc-Huy Trinh, Hai-Dang Nguyen, Nguyen Ngoc Bao Tram, Debesh Jha, Ulas Bagci, and Minh-Triet Tran. Sam-eg: Segment anything model with egde guidance framework for efficient polyp segmentation. In *35th British Machine Vision Conference 2024, BMVC 2024, Glasgow, UK, November 25-28, 2024*. BMVA, 2024.

[155] Théo Trouillon and Maximilian Nickel. Complex and holographic embeddings of knowledge graphs: A comparison. *CoRR*, abs/1707.01475, 2017.

[156] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In Maria-Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 2071–2080. JMLR.org, 2016.

[157] Mark Tygert, Joan Bruna, Soumith Chintala, Yann LeCun, Serkan Piantino, and Arthur Szlam. A mathematical motivation for complex-valued convolutional networks. *Neural computation*, 28(5):815–825, 2016.

[158] Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. *Advances in neural information processing systems*, 29, 2016.

[159] Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. Multi-task learning for dense prediction tasks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(7):3614–3633, 2022.

[160] Bhavya Vasudeva, Puneesh Deora, Saumik Bhattacharya, and Pyari Mohan Pradhan. Compressed sensing mri reconstruction with co-vegan: Complex-valued generative adversarial network. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 672–681, 2022.

[161] T. F. Y. Vicente, Le Hou, Chen-Ping Yu, Minh Hoai, and Dimitris Samaras. Large-scale training of shadow detectors with noisily-annotated shadow examples. In *European Conference on Computer Vision*, 2016.

[162] Tomás F. Yago Vicente, Le Hou, Chen-Ping Yu, Minh Hoai, and Dimitris Samaras. Large-scale training of shadow detectors with noisily-annotated shadow examples. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors,

Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI, volume 9910 of *Lecture Notes in Computer Science*, pages 816–832. Springer, 2016.

[163] Benyou Wang, Donghao Zhao, Christina Lioma, Qiuchi Li, Peng Zhang, and Jakob Grue Simonsen. Encoding word order in complex embeddings. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[164] Jiayun Wang, Patrick Virtue, and Stella X Yu. Successive embedding and classification loss for aerial image classification. *arXiv preprint arXiv:1712.01511*, 2017.

[165] Jifeng Wang, Xiang Li, and Jian Yang. Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1788–1797. Computer Vision Foundation / IEEE Computer Society, 2018.

[166] Jifeng Wang, Xiang Li, and Jian Yang. Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[167] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 3796–3805. IEEE Computer Society, 2017.

[168] Peng Wang, Zhi Zhou, Wei Liu, and Hong Qiao. Salient region detection based on local and global saliency. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1546–1551, 2014.

[169] Runqi Wang, Zhen Liu, Baochang Zhang, Guodong Guo, and David S. Doermann. Few-shot learning with complex-valued neural networks and dependable learning. *Int. J. Comput. Vis.*, 131(1):385–404, 2023.

[170] Chenxi Wei, Zhenyuan Ji, Maosheng Wei, Yun Zhang, and Haoxuan Yuan. Scv-unet: Saliency-combined complex-valued u-net for sar ship target segmentation. In *IGARSS 2023 - 2023 IEEE International Geoscience and Remote Sensing Symposium*, pages 6948–6951, 2023.

[171] Haibin Wu, Ke Tan, Buye Xu, Anurag Kumar, and Daniel Wong. Rethinking complex-valued deep neural networks for monaural speech enhancement. *CoRR*, abs/2301.04320, 2023.

[172] Xin Wu, Danfeng Hong, and Jocelyn Chanussot. Uiu-net: U-net in u-net for infrared small object detection. *IEEE Transactions on Image Processing*, 32:364–376, 2023.

[173] Saurabh Yadav and Koteswar Rao Jerripothula. Fccns: Fully complex-valued convolutional networks using complex-valued color model and loss function. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10689–10698, October 2023.

[174] Saurabh Yadav and Koteswar Rao Jerripothula. Fccns: Fully complex-valued convolutional networks using complex-valued color model and loss function. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 10655–10664. IEEE, 2023.

[175] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pages 1155–1162. IEEE Computer Society, 2013.

[176] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pages 3166–3173. IEEE Computer Society, 2013.

[177] Muqiao Yang, Martin Q. Ma, Dongyu Li, Yao-Hung Hubert Tsai, and Ruslan Salakhutdinov. Complex transformer: A framework for modeling complex-valued sequence. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, pages 4232–4236. IEEE, 2020.

[178] Eric C. Yeats, Yiran Chen, and Hai Li. Improving gradient regularization using complex-valued neural networks. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 11953–11963. PMLR, 2021.

[179] Bowen Yin, Xuying Zhang, Qibin Hou, Bo-Yuan Sun, Deng-Ping Fan, and Luc Van Gool. Camoformer: Masked separable attention for camouflaged object detection. *arXiv preprint arXiv:2212.06570*, 2022.

[180] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis E. H. Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 538–547. IEEE, 2021.

[181] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1476–1485, 2019.

[182] Shuai Zhang, Yi Tay, Lina Yao, and Qi Liu. Quaternion knowledge graph embeddings. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 2731–2741, 2019.

[183] Jiaxing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. Egnet: Edge guidance network for salient object detection. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 8778–8787. IEEE, 2019.

[184] Wenda Zhao, Xueqing Hou, You He, and Huchuan Lu. Defocus blur detection via boosting diversity of deep ensemble networks. *IEEE Trans. Image Process.*, 30:5426–5438, 2021.

[185] Wenda Zhao, Cai Shang, and Huchuan Lu. Self-generated defocus blur detection via dual adversarial discriminators. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 6933–6942. Computer Vision Foundation / IEEE, 2021.

[186] Wenda Zhao, Fan Zhao, Dong Wang, and Huchuan Lu. Defocus blur detection via multi-stream bottom-top-bottom fully convolutional network. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3080–3088. Computer Vision Foundation / IEEE Computer Society, 2018.

[187] Wenda Zhao, Bowen Zheng, Qiuhua Lin, and Huchuan Lu. Enhancing diversity of defocus blur detectors via cross-ensemble network. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 8905–8913. Computer Vision Foundation / IEEE, 2019.

[188] Xiaoqi Zhao, Youwei Pang, Lihe Zhang, Huchuan Lu, and Lei Zhang. Suppress and balance: A simple gated network for salient object detection. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part II*, volume 12347 of *Lecture Notes in Computer Science*, pages 35–51. Springer, 2020.

[189] Quanlong Zheng, Xiaotian Qiao, Ying Cao, and Rynson W. H. Lau. Distraction-aware shadow detection. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 5167–5176. Computer Vision Foundation / IEEE, 2019.

[190] Ge Zhu, Jinbao Li, and Yahong Guo. Priornet: Two deep prior cues for salient object detection. *IEEE Transactions on Multimedia*, 26:5523–5535, 2024.

[191] Lei Zhu, Zijun Deng, Xiaowei Hu, Chi-Wing Fu, Xuemiao Xu, Jing Qin, and Pheng-Ann Heng. Bidirectional feature pyramid network with recurrent attention residual modules for shadow detection. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VI*, volume 11210 of *Lecture Notes in Computer Science*, pages 122–137. Springer, 2018.

[192] Lei Zhu, Ke Xu, Zhanghan Ke, and Rynson W. H. Lau. Mitigating intensity bias in shadow detection via feature decomposition and reweighting. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 4682–4691. IEEE, 2021.

[193] Xuanyu Zhu, Yi Xu, Hongteng Xu, and Changjian Chen. Quaternion convolutional neural networks. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII*, volume 11212 of *Lecture Notes in Computer Science*, pages 645–661. Springer, 2018.