Quaternion-Enhanced Neural Networks: A New Paradigm for Audio Processing Efficiency

A THESIS

submitted by

ARYAN CHAUDHARY

in partial fulfilment of the requirements
for the award of the degree of

MASTER OF TECHNOLOGY

COMPUTER SCIENCE ENGINEERING
INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI

NEW DELHI- 110020

May 2024

THESIS CERTIFICATE

This is to certify that the thesis titled **Quaternion-Enhanced Neural Networks: A New Paradigm for Audio Processing Efficiency**, submitted by **Aryan Chaudhary**, to the INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY, DELHI, for the award of the degree of **Master of Technology**, in Computer Science Engineering with specialization in AI, is a bonafide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Vinayak Abrol
Thesis Supervisor
Assistant Professor
Dept. of Computer Science Engineering
IIIT Delhi, 110020

Place: New Delhi
Date: June 13, 2024

ACKNOWLEDGEMENTS

I am immensely grateful to the Computer Science Engineering Department and the Infosys Centre for AI (CAI) at IIIT Delhi. Their generous provision of computational resources and financial support has been crucial to my research.

I extend my deepest thanks to my lab, Cross-Caps Laboratory (I doubt I'll ever have a better desk to sit and work at), and my advisor, Dr. Vinayak Abrol, whose guidance, not only as an advisor but also as a mentor, has been invaluable. Dr. Abrol has been a mentor like no other, supporting me with not just academic guidance but also ensuring that all logistical aspects of my research trips, travels, and workshop registrations were taken care of. His boundless enthusiasm for mathematical constructs and the foundational aspects of speech analysis along with his unparalled work ethic and vast knowledge has been a constant source of inspiration and motivation for me. I could not have asked for any better.

I also extend my gratitude to Dr. Arshdeep Singh and Professor Mark D. Plumbley from the CVSSP lab (Centre for Vision, Speech and Signal Processing), University of Surrey, London, for their collaboration. Their insights have significantly enriched this work. I would like to acknowledge the effort put by them to collaborate even with timezone differences and also providing the HPC cluster for the neccessary compute required.

My participation in the WASPAA 2023, NY, USA (IEEE Workshop on Applications of Signal Processing to Audio and Acoustics) was an honor, and I am very thankful for the WASPAA grant that supported my presentation there. I would also mention the sheer support given by, Dr. Minje Kim, WASPAA General Chairs, to make my travel possible. This work has been also supported by the SERB Startup Research Grant Government of India, and Infosys Foundation via the Infosys Centre for AI, IIIT Delhi. My role as a postgraduate researcher employed under CAI supported by this grant has been a great helping hand in my academic pursuit.

I am also thankful to Vishal Kumar, a Ph.D. candidate in CrossCap Lab, whose help

and insights have been very beneficial in my research. I am especially grateful to my friends who convinced me to opt for thesis and then supported me through my countless rants and frustrations as I navigated complex challenges in my research. Balancing the rigorous coursework and thesis work at IIITD was particularly tough in the beginning, but their support and care were crucial to maintaining my health and managing my workload effectively.

I am also thankful to my parents and my sister, whose unwavering support has been my cornerstone in my master's journey.

Thank you all once again for your invaluable support and belief in my work.

ABSTRACT

KEYWORDS: Quaternion Neural Networks; Audio Processing; Keyword Spotting; Audio Tagging; Speech Synthesis; Vocoder Models; Model Compression; Loss-Landscape Visualisation; Speech and Audio Applications

This thesis explores the integration of quaternion algebra into neural network architectures to enhance their efficiency for diverse audio processing tasks. Quaternion-based transformations are employed to achieve structural compression to reduce model size and computational demands. Further, this is achieved while retaining the task's high accuracy and reliability and enhancing the model's learning capabilities. This thesis presents three main studies: the first focuses on applying quaternion models for on-device keyword spotting, demonstrating their ability to match the performance of state-of-the-art models with a fraction of the computational footprint; the second investigates the combined use of quaternion transformations and pruning techniques in convolutional neural networks for audio tagging, achieving substantial reductions in computational demands and memory usage; the third explores the use of quaternion algebra in speech synthesis through vocoder models, which enables high-quality speech generation with significantly reduced parameter sizes and computational overhead. The proposed quaternion models demonstrate substantial reductions in parameter count and computational load across these applications, making them suitable for deployment on resource-limited devices. Experimental validations on standard datasets highlight the effectiveness and versatility of these models. Together, these studies underscore the potential of quaternion-based models in advancing real-world applications on edge devices. All these studies achieve or set the state-of-the-art performance in their respective domains.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

ADAM	Adaptive Moment Estimation Optimizer
AI	Artificial Intelligence
ANNs	Artificial Neural Networks
ASRU	Automatic Speech Recognition and Understanding Workshop
BCNet	Broadcast Residual Network
BCResNet	Broadcast Residual Network
CLAP	Contrastive Language-Audio Pretraining
CNN	Convolutional Neural Network
CNN14	CNN model with 14 layers
CVSSP Lab	Centre for Vision, Speech and Signal Processing
DL	Deep Learning
DNN	Deep Neural Network
DSC	Depthwise Separable Convolutions
E-QCNN	Efficient Quaternion Convolutional Neural Network
EIE	Efficient Inference Engine
FAD	Fréchet Audio Distance
FNN	Feedforward Neural Network
GANs	Generative Adversarial Networks
HMM	Hidden Markov Model
ICASSP	IEEE International Conference on Acoustics, Speech, and Signal Processing
ICLR	International Conference on Learning Representations
ICML	International Conference on Machine Learning
IJCNN	International Joint Conference on Neural Networks
IIITD	Indraprastha Institute Of Information Technology, Delhi
IoT	Internet of Things
KD	Knowledge Distillation
KWS	Keyword Spotting
LR	Learning Rate

MACs	Multiplication-Accumulation Operations
mAPs	Mean Average Precision Scores
MOS	Mean Opinion Score
MPD	Multi Period Discriminator
MSD	Multi-Scale Discriminator
NLP	Natural Language Processing
PANNs	Pre-trained Audio Neural Networks
PESQ	Perceptual Evaluation of Speech Quality
QBCResNet	Quaternion Broadcast Residual Network
QCNN	Quaternion Convolutional Neural Network
QGAN	Quaternion Generative Adversarial Network
QMPD	Quaternion Multi-Period Discriminator
QMSD	Quaternion Multi-Scale Discriminator
QMRF	Quaternion Multi-Receptive Field Fusion
QNN	Quaternion Neural Network
QFNN	Quaternion Feedforward Neural Network
ReLU	Rectified Linear Unit
ResNet	Residual Network
RNNs	Recurrent Neural Networks
RTFM	Read The Fine Manual
SOTA	State Of The Art
STFT	Short Time Fourier Transform
STOI	Short-Time Objective Intelligibility
TF	Time-Frequency
TLDR	Too Long; Didn't Read
WASSPA	Workshop on Applications of Signal Processing to Audio and Acoustics

NOTATION

$\bar{\sigma}$	Activation Function
σ	Split Activation Function
γ_{nm}^{l}	Quaternion output at layer l and position (n, m)
O_{nm}^{l}	Pre-activation quaternion output at layer l and position (n, m)
w_{nm}^{l-1}	Quaternion-valued weight filter at layer $l-1$ for position (n, m)
W	Combined weight matrix for the real, $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ components
$\mathbf{q}$	Quaternion vector representing input features, $\mathbf{q} = \mathbf{a} + \mathbf{bi} + \mathbf{cj} + \mathbf{dk}$
$W_i, i = 1 - 4$	Real and imaginary components of Quaternion filter/weight matrix
$M(4, R)$	4 × 4 matrix ring
$\otimes$	Hamiltonian product
$M(2, C)$	2 × 2 complex matrix
i, j, k	Imaginary units
$Q(f, t)$	Quaternion LogMel-Spectrogram for frequency f and time frame t
G	Weight Vector's magnitude used to update weights
V	Direction vector used alongside G for weight updates
δ	Log spectral norm measure of the weight matrix
L_{Adv}	Least square adversarial loss
L_{Mel}	Perception-based Mel loss
L_{FM}	Feature matching loss
ℓ_1	Manhattan distance based norm
γ	Exponential-decay rate
$s(n)$, $\hat{s}(n)$	Original and Processed/Synthesized speech signal respectively
$S(\omega), \hat{S}(\omega)$	Fourier transforms of the original and synthesized signals, respectively
f	Function representing the perceptual model
μ_r, μ_s	Means of vectors reference and synthesized audio, respectively
Σ_r, Σ_s	Covariance Matrix of reference and synthesized audio, respectively
Tr	Trace of a matrix
θ^*	Central point in the Optimization Space
δ, η	Direction vectors considered from θ^*
ϕ	Function that transforms a waveform into the corresponding Mel-spectrogram
D_Q	Quaternion Discriminator
G_Q	Quaternion Generator
p	Pruning percentage
$\sigma(A)$	Spectral Norm of the matrix A
R	Real number domain
H	Hypercomplex number domain

Most of the notations used are specified here. In terms of clash each and every notation has been defined where they have been used.

CHAPTER 1

INTRODUCTION

1.1 General Overview

Neural networks, a subset of machine learning, have fundamentally transformed how we interact with technology. Originating from the desire to mimic the human brain's architecture and functionality, neural networks have evolved through the years to become a cornerstone of deep learning. This evolution has been marked by the development from simple perceptrons to complex architectures like convolutional neural networks (CNNs) and recurrent neural networks (RNNs), which are capable of processing spatial and sequential data, respectively. Deep learning, which involves neural networks with multiple layers (hence "deep"), has significantly enhanced the capabilities of these models, leading to breakthroughs across various domains such as image recognition, natural language processing, and autonomous driving. The ability of deep learning to extract patterns from large datasets and make intelligent predictions has not only advanced scientific research but also transformed industries by driving innovations that were once thought impossible.

In this rapidly evolving field of artificial intelligence, neural networks have also prominently emerged as transformative tools across a broad spectrum of audio processing applications. Audio processing is a dynamic field within signal processing that focuses on the analysis and synthesis of audio signals. With the advent of digital media and smart devices, the importance of audio processing has been magnified, catering to a plethora of applications ranging from simple noise reduction in call quality keyword spotting, speaker recognition, speech recognition, voice conversion, speech enhancement, and audio tagging, to complex tasks like music synthesis, speech synthesis, and voice-assisted technologies. Traditional neural network architectures, while effective, often suffer from significant computational and memory overheads, making them less ideal for use in environments with limited resources. Furthermore, Speech signals are intricate, composed of a variety of sine and cosine waves. Feature extraction from these

signals typically employs methods such as Fourier transforms, wavelet transforms, and frequency banks. These methods produce high-dimensional data and hence there arises a need to capture relationships across multiple dimensions accurately.

Recent advancements in quaternion algebra-based Neural Networks present a promising avenue to address these challenges. Quaternion numbers, consisting of one real and three imaginary components, allow the compact representation of multidimensional data and enable significant reductions in the complexity and size of neural models without sacrificing performance. This thesis explores the integration of quaternion algebra into neural network architectures, aiming to harness these benefits to enhance model efficiency and performance in the audio domain.

1.2 Setting up the Flow

In the forthcoming sections, we define the objectives of the thesis and develop a series of research questions to identify technological gaps and set up our hypotheses. We then explore current use cases of Quaternion Neural Networks (QNNs) to understand their value proposition. Subsequently, we delve into the motivations behind employing QNNs from various perspectives and set up the case studies conducted in the thesis.

The next chapter, titled "Quaternion Network Preliminaries," equips the reader with all the ammunition required to understand and develop QNNs. We introduce the foundational concepts of Quaternion Algebra and its practical implementation in constructing neural network architectures. This theoretical groundwork sets the stage for the detailed exploration of specific use cases.

Once we have established the problem statement and are equipped with sufficient knowledge, we commence with case studies (chapters 3, 4, and 5), each detailed in its own chapter. Each chapter independently defines its problem statement, sets up the experiments, and describes the dataset, model architecture, and all technical requirements. Results are discussed and outcomes are analyzed independently in each chapter. Through these chapters, we tackle independent problems using novel approaches and attempt to understand the results using various techniques or mathematical frameworks.

In the final chapter, we comprehensively cover all research questions, objectives,

and hypotheses of our work. We outline the central themes of all three case studies and use their unique contributions to validate our central hypothesis. We also discuss the limitations and future scope of our work and conclude our thoughts on the entire project.

1.3 Objectives of the Thesis

1.3.1 The Research Gap

While quaternion neural networks (QNNs) have been explored within the realms of computer graphics and robotics, their application in the field of audio processing remains largely untapped. Current research predominantly focuses on traditional real-valued neural networks, which, despite their widespread use and success, often struggle with the complexities inherent in multidimensional audio data. This oversight presents a significant research gap: the potential of quaternion algebra to enhance audio processing has not been fully explored or utilized. The novelty of applying quaternion algebra to audio processing lies in its theoretical capability to manage complex, interrelated data dimensions more naturally and efficiently than scalar-based approaches. This application could lead to groundbreaking improvements in how audio data is processed, yet it lacks comprehensive empirical studies and methodological developments specific to audio applications. By addressing this gap, this research aims to pioneer the integration of QNNs into audio processing, setting a foundation for future explorations and technological advancements in this domain.

1.3.2 Research Questions and Hypotheses

The exploration of quaternion neural networks (QNNs) in audio processing is guided by several pivotal research questions that aim to uncover the potential benefits and limitations of this approach. These questions are designed to direct the investigation toward meaningful, measurable outcomes that can validate the utility of QNNs over traditional neural network architectures. The primary research questions are:

Can quaternion neural networks provide better performance and efficiency in

audio processing tasks compared to traditional real-valued networks? This question seeks to determine whether the unique properties of quaternion algebra contribute to enhanced performance metrics such as accuracy, speed, and reliability in audio processing tasks like keyword spotting, audio tagging, and speech synthesis.

How do quaternion neural networks handle the multidimensional nature of audio data in comparison to scalar-based neural networks? This question addresses the theoretical aspect of quaternion application, specifically its ability to naturally encode and process multidimensional data structures, which is critical for complex audio signal processing.

What are the computational benefits of using quaternion neural networks in terms of parameter efficiency? Focused on the computational advantages, this question examines whether the reduced parameter space capabilities of QNNs translate into tangible efficiencies in real-world applications.

To what extent can quaternion neural networks be integrated into existing audio processing systems, and what are the challenges associated with such integration? This question explores the practical aspects of implementing QNNs within current technological frameworks, assessing the compatibility and adaptation required to leverage quaternion benefits in established systems.

Can Quaternion Neural Networks scale up to accommodate larger networks, more extensive training data, and a higher cardinality of classes? This question investigates whether QNNs can maintain their computational efficiency and performance advantages as the network architecture becomes larger (wider and deeper), the training data sets grow more extensive, and the number of classes increases. It addresses the critical aspect of whether the unique properties of quaternion algebra continue to provide benefits in much larger scales, which is essential for applications in fields such as large-scale image and video processing, comprehensive language models, and extensive sensor networks.

Can the theoretical advantages of quaternion space be supported in practicality with explainable AI experiments? This research question aims to bridge the gap between theoretical mathematical advantages and practical, observable outcomes within the realm of quaternion neural networks. Explainable AI (XAI) provides methods and

techniques to make the operation of AI models transparent and understandable to humans. This question is crucial as it addresses the interpretability and transparency of quaternion-based models, which is a significant aspect when assessing the adaptability and trustworthiness of new neural network architectures.

One Central Hypothesis

"Quaternion Neural Networks are suitable and effective for the audio domain, capable of enhancing performance, computational efficiency, and scalability compared to traditional real-valued neural networks, while also being explainable and integrable into existing systems."

1.4 Quaternion: A little literature Survery

Current research shows the use of Quaternion Neural Networks in various domains including but not limited to image processing Parcollet *et al.* (2019); Yin *et al.* (2019); Kusamichi *et al.* (2004), text processing Tay *et al.* (2019*b*), Generative networks Grassucci *et al.* (2022), Graph networks Nguyen *et al.* (2021), RNNs Parcollet *et al.* (2018*a*), 3d space transformations Kumar *et al.* (2020), facial expression recognition Zhou *et al.* (2021), color image compression Luo *et al.* (2010), robot control Cui *et al.* (2013), PolSAR land classification Shang and Hirose (2013), banknote classification Huang and Gai (2020), speech emotion recognition Muppidi and Radfar (2021), multi-distant speech recognition system Qiu *et al.* (2020), spoken language understanding Parcollet *et al.* (2016), automatic speech recognition system Parcollet *et al.* (2018*b*), etc.

1.5 Motivation: Why Quaternion?

The current generation of neural networks faces three interconnected challenges: they possess a large footprint, exhibit a high degree of freedom, and struggle to understand the latent semantic relationships among different data dimensions, particularly when processing signal-processing based data. This high degree of freedom often leads to overparameterization, compounding the complexity and inefficiency of the models.

This thesis addresses these critical issues by proposing enhancements to neural network architectures that tackle each of these problems. Experimental evidence provided herein supports our claims, demonstrating significant improvements in neural network performance across these three key areas.

1.5.1 Paradigm I: The Model Size

Traditional deep learning models, particularly those employed in audio processing tasks such as keyword spotting, audio tagging, and speech synthesis, typically require vast amounts of computational resources and large memory footprints. This presents a significant challenge for on-device AI applications where power consumption, memory use, and processing speed are limited by hardware capabilities. Furthermore, the increasing demand for real-time processing and the need for privacy-preserving on-device computations push for advancements in model efficiency and compactness. One of the main motivations for this thesis stems from the need to explore and validate quaternion algebra as a viable approach to reduce the computational and memory overhead of neural networks while maintaining, or even enhancing their performance.

Quaternions Neural Networks as an alternative for Real valued neural networks offer 75 percent lesser parameters. This happens due to the representation of QNNs. To represent a 4D signal, QNNs will require one node input to one node output while the real-valued networks will take 4 node input to 4 node output. Here in QNNs one node input to one node output will have 4 parameters while for real-valued networks it will be 4x4=16 parameters. Therefore here we see the fourfold memory saving. Hence QNNs not only are more native to represent multidimensional signals but they also drastically reduce the network footprint. Furthermore, a section of this research is driven by the hypothesis that quaternion-based transformations, coupled with techniques like pruning, can create highly efficient neural architectures without compromising the accuracy required for practical applications. The ultimate goal is to enable the broader adoption of AI technologies in everyday devices, making intelligent systems more accessible and sustainable in the long term. In Chapter 3 we show through experiments the overparameterization issues in conventional neural networks and give insights into the learning behavior of the Networks.

1.5.2 Paradigm II: The Multidimensional Madness

In the recent few years in Speech and Audio Processing, researchers have shifted from HMM, GMM-based models to Deep Neural Networks to solve various tasks including but not limited to speaker identification, speaker verification, speech emotion, speaker diarization, speaker detection, Keyword spotting, speech synthesis, etc. Most of these tasks require the audio signals to be presented in a time or frequency domain. Speech signals are complex signals and are made up of varying sines and cosines. Extracting features from these signals involves the use of Fourier transforms, wavelet transforms, frequency banks, etc. All these are high dimensional in nature and possess interdimensional information. An efficient AI model will be able to encode these internal latent interactions within the data and should be able to learn the global features as well. When this data becomes multichannel it is possible that the data has some internal intricacies which can be learned by a model. This often happens in the case of audio signals. A more effective approach to representing the information contained in an acoustic signal sequence is to view each time-frame as a unified entity consisting of three closely interconnected elements, rather than a collection of one-dimensional elements that may or may not be related to each other, as in conventional real-valued neural networks. This means that by treating each time-frame as a whole entity with its own unique properties, we can better capture the complex relationships between the elements within the sequence. This approach is particularly useful in processing acoustic signals, as it allows for a more efficient and accurate representation of the data. In contrast, traditional real-valued neural networks may struggle to capture the intricacies of acoustic signals due to their unidimensional nature. The current models miss this opportunity to learn the internal correlation within various channels of the data. Most present-day models learn the features independently from 3D/4D data. These models are not natural extensions to the 3D/4D signals.

It is required that a more adequate way to represent multidimensional input features which is more innate to capture the internal relations should be investigated. Real-valued neural network architectures can be expanded to include the quaternion domain in order to take advantage of its capabilities. This involves incorporating quaternion-valued neurons and operations into the neural network architecture. By doing so, the network can handle more complex data structures, such as hypercomplex numbers and

multidimensional arrays, which will form the basis for most of the signal processing applications.

In the upcoming sections, we show that, For example, multidimensional log Mel features and their derivatives, which help in integrating the interrelated semantic information and elevate the observable frequencies in the acoustic signal, can be used in a QNNss more effectively than real-valued networks.

1.5.3 Paradigm III: The Issues with Degree Of Freedom

The Artificial Neural Networks given by McCulloch and Pitts McCulloch and Pitts (1943) were based on biological neurons. The modern-day ANNs try to mimic the human brain neuron model. When considering biological neurons as a motivation for artificial neurons, it is important to note that their activation energies can have varying pulse configurations and intervals between pulses. This variability suggests the potential benefits of using a quaternionic signal to represent neuronal activity, which contains both amplitude and phase information as well as correlations among its components. This approach has shown promise in accurately capturing the complex dynamics of biological neurons and may lead to more effective modeling of neural activity and improved understanding of neural function. Hence it becomes the biggest motivation to use a Quaternon number system based neural network as it is closer to mimicking the brain as compared to traditional ANNs.

Quaternions have become an active field of research due to their properties coming from Quaternion algebra and the way they are represented in terms of Neural Network architecture. In Quaternionic NNs, internal relationships between the channels are captured better than the real-valued networks due to the Hamilton product. The large degree of freedom of Real valued networks makes this problem more difficult. To alleviate this difficulty it is reasonable to use the implicit assumptions of the input features in the model. This will enhance the learning capability of the Network and will also require fewer parameters to converge. Hence the high degree of freedom that causes the variability in the network can be solved by using Quaternions. Why so? Interestingly, in QNNs, the Hamilton Product restricts the degree of freedom and the parameters are lesser due to its representation capacity. Hence it solves this issue. We

show loss landscape experiments which show the effects of overparameterization and degree of freedom in the learning behavior of the Network.

1.6 A Tripartite Case Study of Speech Applications

We have selected three distinct audio applications as case studies to comprehensively address our research questions. These applications have been carefully chosen to ensure a broad coverage of scenarios, enabling us to confidently validate our central hypothesis. Below is an overview of each case study, detailing the experiments and methodologies we have implemented to test the effectiveness of Quaternion Neural Networks in the audio domain.

First Study: Key Word Spotting

The first study presented in this thesis investigates quaternion neural models applied to on-device keyword spotting. It demonstrates that these models can match the performance of state-of-the-art alternatives while requiring only a fraction of the computational resources. This exploration highlights the potential of quaternion models to replace traditional convolutional neural networks (CNNs) in environments where computational efficiency is critical. At the time of writing, we hold SOTA for the KWS task on the Google Speech Command dataset.

Second Study: Audio Tagging

The second study expands the application of quaternion transformations to convolutional neural networks for audio tagging. By integrating quaternion algebra with pruning techniques, this study achieves substantial reductions in computational demands and memory usage. This dual approach not only maintains high model accuracy but also enhances the scalability and feasibility of deploying advanced audio tagging systems on edge devices. At the time of writing, we have given the smallest model for audio tagging on the Audioset dataset with competitive accuracy.

Third Study: Speech Synthesis

The third study delves into the realm of speech synthesis using vocoder models adapted to the quaternion framework. This part of the research demonstrates how quaternion-based vocoders can generate high-quality speech while utilizing significantly fewer parameters than traditional models. The findings indicate that quaternion vocoders provide a robust method for efficient speech generation, making them particularly suitable for applications in mobile and embedded systems. At the time of writing, we have given the smallest vocoder for the Hindi language trained on the OPenSLR dataset.

The Converging Point

This integrated approach not only broadens the utility of quaternion networks in audio processing but also sets a precedent for their use in other domains requiring efficient computational solutions. Through detailed experimental evaluations, this work substantiates the hypothesis that quaternion-based models are not merely theoretical constructs but practical tools for advancing real-world applications on edge devices. The implications of these findings extend beyond audio processing, offering pathways to innovation in areas as diverse as telecommunications, healthcare, and consumer electronics, where the efficient processing of complex data streams is paramount.

This thesis aims to contribute to the body of knowledge in efficient neural architecture design with better signal representation and learning capacity, providing a foundation for future research and development in the field of deep learning and signal processing.

CHAPTER 2

Quaternion Network Preliminaries

2.1 Quaternion Algebra

Quaternions are hypercomplex numbers extending the complex domain Parcollet *et al.* (2017) using a real and three imaginary components as

$$Q = r + x\mathbf{i} + y\mathbf{j} + z\mathbf{k};$$

$$r, x, y, z \in \mathbb{R}; \quad \mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1.$$

In quaternion models, the standard dot product is replaced with the Hamilton product. Quaternions facilitate swift operations analogous to those performed with matrices, thanks to the existence of a lossless one-to-one mapping for both their addition and multiplication processes while maintaining the structural properties of these operations Kumar *et al.* (2020). For instance, the Hamiltonian product $Q_1 \otimes Q_2$ can be expressed in terms of a 4×4 matrix ring $\mathbf{M}(4, \mathbb{R})$ as

For example, the Hamiltonian product ($\otimes$) between two quaternions $Q_1 = r_1 + x_1\mathbf{i} + y_1\mathbf{j} + z_1\mathbf{k},$ and $Q_2 = r_2 + x_2\mathbf{i} + y_2\mathbf{j} + z_2\mathbf{k}$ can be efficiently expressed in terms of a 4×4 matrix ring $\mathbf{M}(4, \mathbb{R})$ as (or in layman terms, in terms of the products of the basis elements as)

$$\begin{aligned} Q_1 \otimes Q_2 =& (r_1 r_2 - x_1 x_2 - y_1 y_2 - z_1 z_2) + \\ & (r_1 x_2 + x_1 r_2 + y_1 z_2 - z_1 y_2)\boldsymbol{i} + \\ & (r_1 y_2 - x_1 z_2 + y_1 r_2 + z_1 x_2)\boldsymbol{j} + \\ & (r_1 z_2 + x_1 y_2 - y_1 x_2 + z_1 r_2)\boldsymbol{k}. \end{aligned}$$

It is also possible to express operations using 2×2 complex matrix $\mathbf{M}(2, \mathbb{C})$ Gaudet and Maida (2018*a*).

2.2 Implementing Quaternion Model Conversion

One can design a quaternion variant of a conventional neural model by replacing 1) regular matrix-vector multiplications with the Hamiltonian product and 2) activation function with a split activation function. In this work, we are interested mostly in CNNs, and thus we describe the conversion process for the same below. We also cover the conversion process for the FNN.

2.2.1 Quaternion FNN

Conversion of a FNN to a Quaternion FNN requires the number of channels to be divisible by 4. In particular, we split the channels into four groups, each representing the real components and the $\mathbf{i}, \mathbf{j}, \mathbf{k}$ imaginary components. Instead of Dot Production, in the quaternion domain feed-forward layer we use the Hamiltonian product. This product is of a quaternion weight matrix $\mathbf{W} = \mathbf{W_1} + \mathbf{W_2}\mathbf{i} + \mathbf{W_3}\mathbf{j} + \mathbf{W_4}\mathbf{k}$ by a quaternion input vector $\mathbf{q} = \mathbf{a} + \mathbf{b}\mathbf{i} + \mathbf{c}\mathbf{j} + \mathbf{d}\mathbf{k}$ as shown above, where $\mathbf{W}$ is combined weight matrix for the real, i, j, and k components. Note that as a result of operations with a 4×4 matrix ring, QCNN results in $4\times$ parameter saving compared to conventional feed-forward networks.

In layman's terms, Quaternion operations lead to $4\times$ parameter savings in a deep neural network (see Figure 2.1 for an illustration). A dense neural network layer with n inputs and outputs results in a total n^2 number of parameters. In contrast, the quaternion domain encapsulates n real input units into $\frac{n}{4}$ units in the hypercomplex domain and represents each parameter in the quaternion neural network in the hypercomplex domain. Thus, quaternion transformation yields ($\frac{n}{4} \times \frac{n}{4} \times 4 = \frac{n^2}{4}$) parameter, reducing the total number of parameters by 4 times.

Assume γ^l_{nm} and O^l_{nm} be the quaternion output and the pre-activation quaternion output at layer l and at the indexes (n, m) of the output. The quaternion feed-forward network is mathematically expressed as Gaudet and Maida (2018*a*):

$$\gamma^l = \sigma(O^l)$$

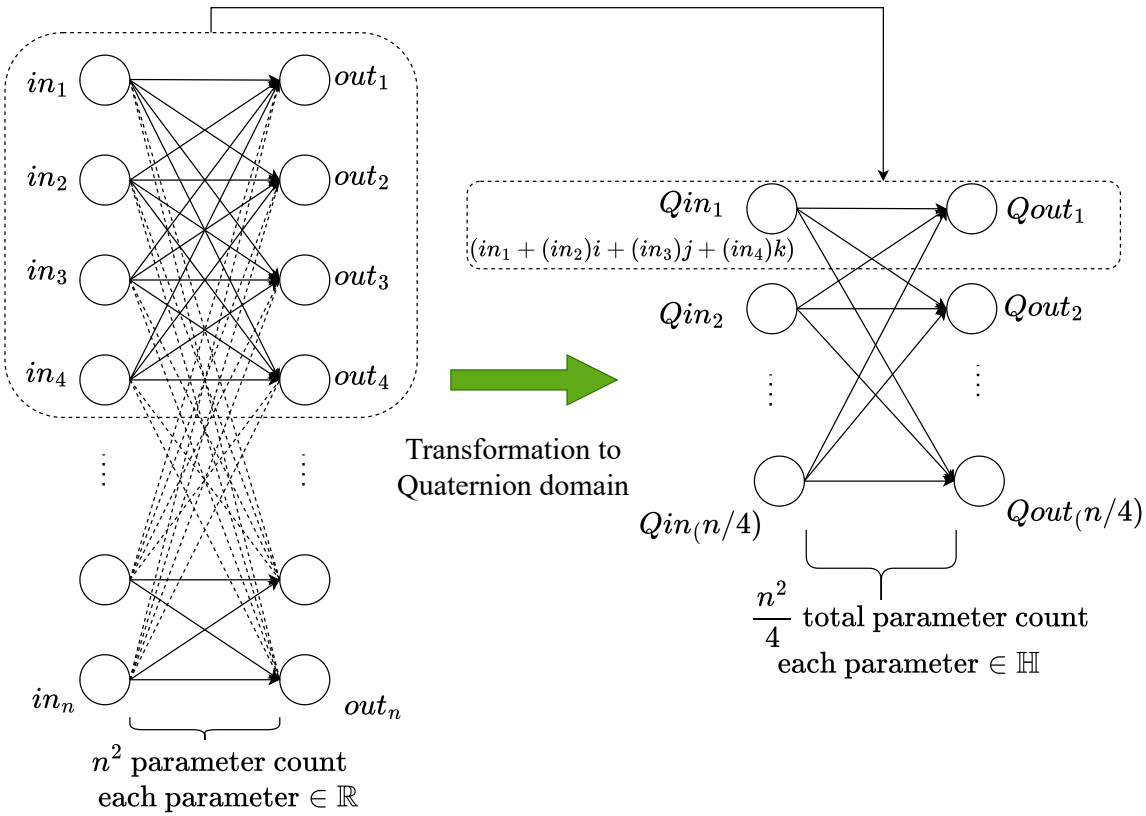


Figure 2.1: Transformation of a conventional neural network in quaternion domain.

$$O^l = \sum_{i=1}^{N} w_i^l \otimes q_i^{l-1}$$

where w^l is the quaternion-valued weight matrix for a map of some size $p \times q$, q_i^{l-1} is Quaternion input from the previous layer of size $q \times$p, σ is a quaternion split activation (defined in next section).

2.2.2 Split Activation Function

In quaternion neural networks, the activation function is an essential component that introduces non-linearity into the learning process. However, traditional activation functions are designed for real numbers and do not directly apply to the quaternion domain, which comprises four interlinked components: one real and three imaginary parts.

To address this, quaternion neural networks utilize what is known as a "split activation function." This type of activation function separately applies a standard, real-valued activation function to each component of the quaternion. The split activation function thus preserves the structure of the quaternion by treating each component

independently, allowing the network to effectively capture the complexities of multi-dimensional data.

The mathematical representation of a split activation function defined in terms of a standard activation function $\bar{\sigma}$ as

$$\sigma(\mathbf{q}) = \bar{\sigma}(\mathbf{a}) + \bar{\sigma}(\mathbf{b})\mathbf{i} + \bar{\sigma}(\mathbf{c})\mathbf{j} + \bar{\sigma}(\mathbf{d})\mathbf{k};$$

where $\bar{\sigma}$ represents a real-valued activation function such as ReLU, sigmoid, or tanh, applied to each individual component.

The adoption of split activation functions in quaternion neural networks is crucial for maintaining the integrity of the quaternion's algebraic properties through layers of transformations, enabling the network to effectively learn from and make predictions based on quaternion-valued data.

2.2.3 Quaternion CNN

Conversion of a CNN to a Quaternion CNN also requires the number of channels to be divisible by 4. Once again, we split the channels into four groups, each representing the real components and the $\mathbf{i}, \mathbf{j}, \mathbf{k}$ imaginary components. This is similar to group convolution, with the constraint that the number of channels should be divisible by 4. However, unlike multiple individual kernels per group, quaternion kernels are shared, i.e., they interact with each group to produce the final output. Convolution in the quaternion domain (QConv) is done by the Hamiltonian product of a quaternion filter matrix $\mathbf{W} = \mathbf{W_1} + \mathbf{W_2}\mathbf{i} + \mathbf{W_3}\mathbf{j} + \mathbf{W_4}\mathbf{k}$ by a quaternion vector $\mathbf{q} = \mathbf{a} + \mathbf{b}\mathbf{i} + \mathbf{c}\mathbf{j} + \mathbf{d}\mathbf{k}$ as shown above, where $\mathbf{W}$ is combined weight matrix for the real, i, j, and k components. Note that as a result of operations with a 4×4 matrix ring, QCNN also results in $4\times$ parameter saving compared to regular convolution. Assume γ^l_{nm} and O^l_{nm} be the quaternion output and the pre-activation quaternion output at layer l and at the indexes (n, m) of the output. The quaternion convolution process is mathematically expressed as Gaudet and Maida (2018*a*):

$$\gamma^l_{nm} = \sigma(O^l_{nm})$$

$$O^{l}_{nm} = \sum_{n'=1}^{K-1} \sum_{m'=1}^{K-1} w^{l-1}_{n'm'} \otimes \gamma^{l-1}_{(n+n')(m+m')},$$

where $w_{n'm'}$ is the quaternion-valued weight filter for a map of size $K \times K$, σ is a quaternion split activation defined above.

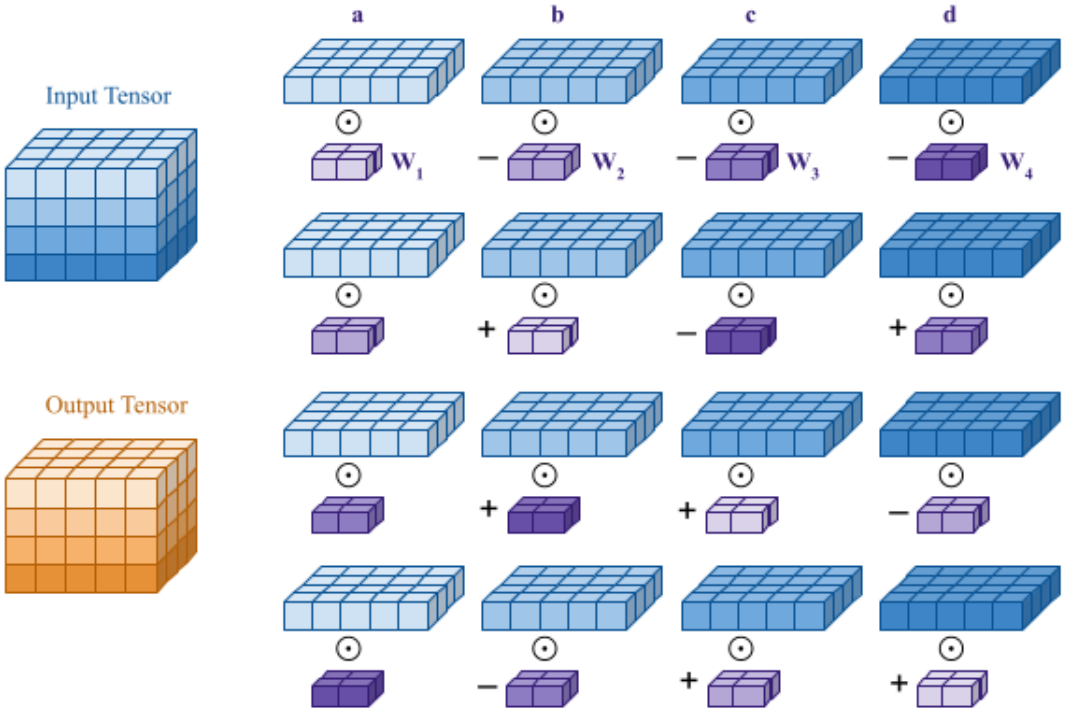


Figure 2.2: Illustration of the quaternion convolution operation using a single quaternion weight $\mathbf{W}$. Best seen in color.

Depthwise Convolutions

In some of the pipelines that we explored during all three studies, we encountered Depthwise separable convolution layers. In regular convolution, each channel of the kernel is multiplied by the corresponding input channel. In contrast, each axis of a quaternion kernel interacts with each axis of the input channel. This is equivalent (though not exactly similar) to Depthwise separable convolution (DSC), where first Depthwise convolution is applied over channels followed by a pointwise convolution to get a linear interaction of the outputs. Thus for models with a DSC layer, one needs to assume an equivalent regular convolutional layer and replace it with its quaternion counterpart. In this case, there will not be any parameter saving, and both the DSC and

QConv layer will have approximately the same number of parameters[1].

2.2.4 Acoustic Quaternions Representations

In all three studies, if required, the given time-domain speech/audio signal is first transformed into a log Mel-spectrogram. Then we employ Logmel-spectrogram-based acoustic quaternion $Q(f,t)$ defined for each frequency f and time frame t. This quaternion acoustic representation is composed of the energy $\psi(f,t)$ within the Mel-filter band at frequency f, along with its first-order (velocity), second-order (acceleration), and third-order (jerk) time derivatives, as described in Chaudhary and Abrol (2023):

$$Q(f,t) = \psi(f,t) + \frac{\partial \psi(f,t)}{\partial t}\mathbf{i} + \frac{\partial^2 \psi(f,t)}{\partial^2 t}\mathbf{j} + \frac{\partial^3 \psi(f,t)}{\partial^3 t}\mathbf{k}$$

This allows QCNN to capture the spatial relationships across different temporal perspectives of the same frequency. One can also build similar quaternions by considering other Time-Frequency representations such as magnitude spectrum, cepstrum, LP spectra, phase spectra, etc.

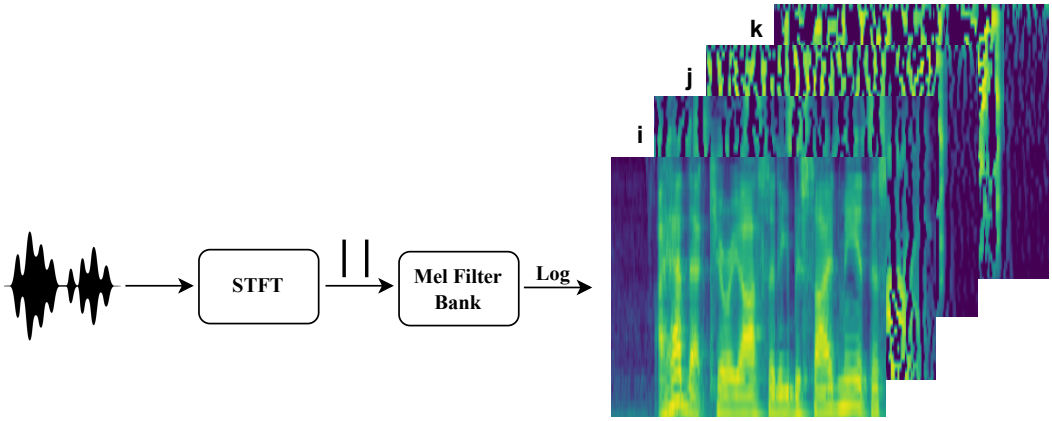


Figure 2.3: Quaternionic Mel-spectrogram

2.2.5 Proposed Quaternion Weight Norm

In chapter 5 we propose Weight normalization in the Quaternion domain as it was required for the speech synthesis training pipeline. Weight normalization Salimans and Kingma (2016) is a training technique that enhances the training stability and efficiency

[1]Both will have an approximately equal number of parameters when the number of channels is exactly divisible by 4.

of a DNN. In particular, this method decouples each weight vector's magnitude G and direction V to update weights as $W = G * (V/||V||)$ that has been shown to achieve quicker convergence. Any layer in a quaternion model consists of 4 groups of channels. Consequently, similar to the split activation, the norm of the weight matrix is also distributed across these four groups as:

$$W = \frac{G_r}{||V_r||}V_r + \frac{G_i}{||V_i||}V_i\mathbf{i} + \frac{G_j}{||V_j||}V_j\mathbf{j} + \frac{G_k}{||V_k||}V_k\mathbf{k}$$

2.2.6 Quaternion Spectral Norm

In chapter 5 we have used the Quaternion version of Spectral normalization (SN) which is the standard technique used essentially for GANs to stabilize training by maintaining a Lipschitz-continuous discriminator Farenick and Pidkowich (2003). In particular, the issue of gradient explosion is addressed by bounding the spectral norm of weights. In the context of QGANs spectral normalization can be applied in two ways namely 1) normalize each submatrix of the quaternion weight matrix independently and 2) normalize the full effective Hamiltonian weight matrix. In this work, we used the first design:

$$W = \frac{W_r}{\sigma(W_r)} + \frac{W_i}{\sigma(W_i)}\mathbf{i} + \frac{W_j}{\sigma(W_j)}\mathbf{j} + \frac{W_k}{\sigma(W_k)}\mathbf{k}$$

CHAPTER 3

Study I: Towards on-device Key Word Spotting

3.1 TLDR

On-device keyword spotting (KWS) is an essential component for wake-up and user interaction on smart edge devices. Existing low-footprint models are mainly based on 2D and 1D convolutions, where the former is better at capturing invariances while the latter enables faster inference times. In this work, we explore Quaternion neural models as an alternative for effective acoustic modeling for the KWS task. Quaternion models can embed various facets of input features within the multiple dimensions of the quaternion space. This leads to smaller & efficient models as compared to their conventional counterparts. We demonstrate this using quaternion versions of the popular KWS models on the Google Command V2 dataset, where our models achieve comparable performance to existing ones. In addition, we also provide an extensive analysis of the learning behavior in the quaternion network to motivate their use in other speech/audio tasks.

3.2 Introduction

Over the last decade, Deep neural networks (DNNs) have continued to demonstrate excellent generalization performance in a wide range of applications, including speech and audio domainsSainath and Parada (2015); Graves *et al.* (2013); Abrol and Sharma (2020). In parallel, innovations in approaches for designing efficient architectures and hardware accelerators have prompted commercial interest in deploying such systems on edge devices. This redefined user experience/interaction with smart devices firstly requires low-latency keyword spotting (KWS) or wake word detection that activates the devices such as 'Alexa & HomePod'. Recent state-of-the-art (SOTA) low-footprint KWS models Majumdar and Ginsburg (2020); Kim *et al.* (2021*a*) have achieved quite remarkable success on public benchmark datasets. These models are mainly based on

2D and 1D convolutions trained over spectral representations e.g., MelSpectrograms; where the former design is better at capturing invariances while the latter enables faster inference times. In addition, some of the works have also proposed recurrent and transformer models for streaming settings Rybakov *et al.* (2020). However, any low-footprint model suffers from poor expressivity & generalization, and such models need constant fine-tuning/adaptation depending on the environment they are deployed, often via training with augmented data.

In this work, we explore Quaternion Kumar *et al.* (2023) neural models as an alternative for effective acoustic modeling for the KWS task. Quaternion models, being an extension of complex Kumar *et al.* (2020) DNN can embed various facets of input features such as multiple channels, spectrograms with delta & acceleration coefficients or magnitude & phase representations, simultaneously within the multiple dimensions of the quaternion space Parcollet *et al.* (2019). In contrast to conventional real-valued DNNs, quaternion models can better learn the complex relationships within various dimensions. Note that conventional multi-channel models are not naturally designed for 3D/4D inputs since they process all the channels as a single tensor, thus ignoring internal correlation among various channels. In quaternion models, the standard dot product is replaced with the Hamilton product, which offers much higher expressivity with very few parameters that are shared across channels Parcollet *et al.* (2019). These models have recently gained traction in speech/audio tasks such as speech recognition Parcollet *et al.* (2018*b*); Qiu *et al.* (2020), event detection Comminiello *et al.* (2019) & speech enhancement.

For the KWS task, we first explore ways to represent the data in the Quaternion domain using popular time-frequency (TF) representations. We then demonstrate the effectiveness of quaternion models using quaternion versions of the popular KWS models on the public Google Command V2 dataset. The aim here is not to propose a state-of-the-art neural architecture but to demonstrate the modeling power of quaternion models.

3.3 Related Work

The Google Speech Command dataset Warden (2018) has become the de facto benchmark for evaluating KWS systems. Since its launch in 2018, various new neural models

have been proposed claiming performance gains over each other. Popular ones include 2D/1D convolutional neural networks (CNN) Sainath and Parada (2015), deep residual networks Wei *et al.* (2022) , recurrent networks Graves *et al.* (2013), RNNTransducer with attention de Andrade *et al.* (2018), and transformer Berg *et al.* (2021). Most of the early models were not efficient in terms of the number of parameters and operations. Recent low-footprint models are mainly CNNs with depthwise separable modules replacing the conventional dense convolutions along the temporal or channel dimension Zhang *et al.* (2010). Two such popular state-of-the-art models are 1D time-channel separable convolution-based MatchboxNet Majumdar and Ginsburg (2020) and 1D/2D convolution-based broadcast residual network (BCNet) Kim *et al.* (2021*a*). Both these models achieve superior performance in extremely low parameter budgets compared to existing models. In addition, there have been attempts to improve the inference time and memory requirements of models as a trade-off with accuracy by using techniques such as pruning, quantization, or via dedicated acceleratorsMordido *et al.* (2021).

3.4 Experiment Setup

This section provides a system description, experimental protocol, and the dataset used in the experimental study.

3.4.1 Dataset

All experiments in this work are performed on the Google Speech Command V2 dataset Warden (2018). The dataset contains 105,829 second-long utterances sampled at 16KHz of 35 words from 2618 speakers. The dataset provides a benchmark for designing and evaluating compact models capable of identifying individual words from a limited vocabulary in the presence of background noise or irrelevant speech. We use the standard train/validation/test set and report the accuracy averaged over 10 trials on the test set.

3.4.2 Model Architecture

In this work, we experimented with SOTA 1D & 2D-convolution based Residual network (ResNet-18) He *et al.* (2016), 1D-DSC based MatchboxNet Majumdar and Ginsburg (2020) refer figure 3.1, 1D/2D-DSC based BCResNet Kim *et al.* (2021*a*) refer figures3.2 and 3.3. We have experimented with different configurations of these models leading to different numbers of model parameters. We have also trained them with inputs having only Mel-spectrogram (1 input channel), Mel-spectrograms with 3-time derivatives (4 input channels), or acoustic quaternion. In order to experiment with different model sizes the number of channels per layer is adjusted to achieve a desired parameter budget without changing the depth of the network.

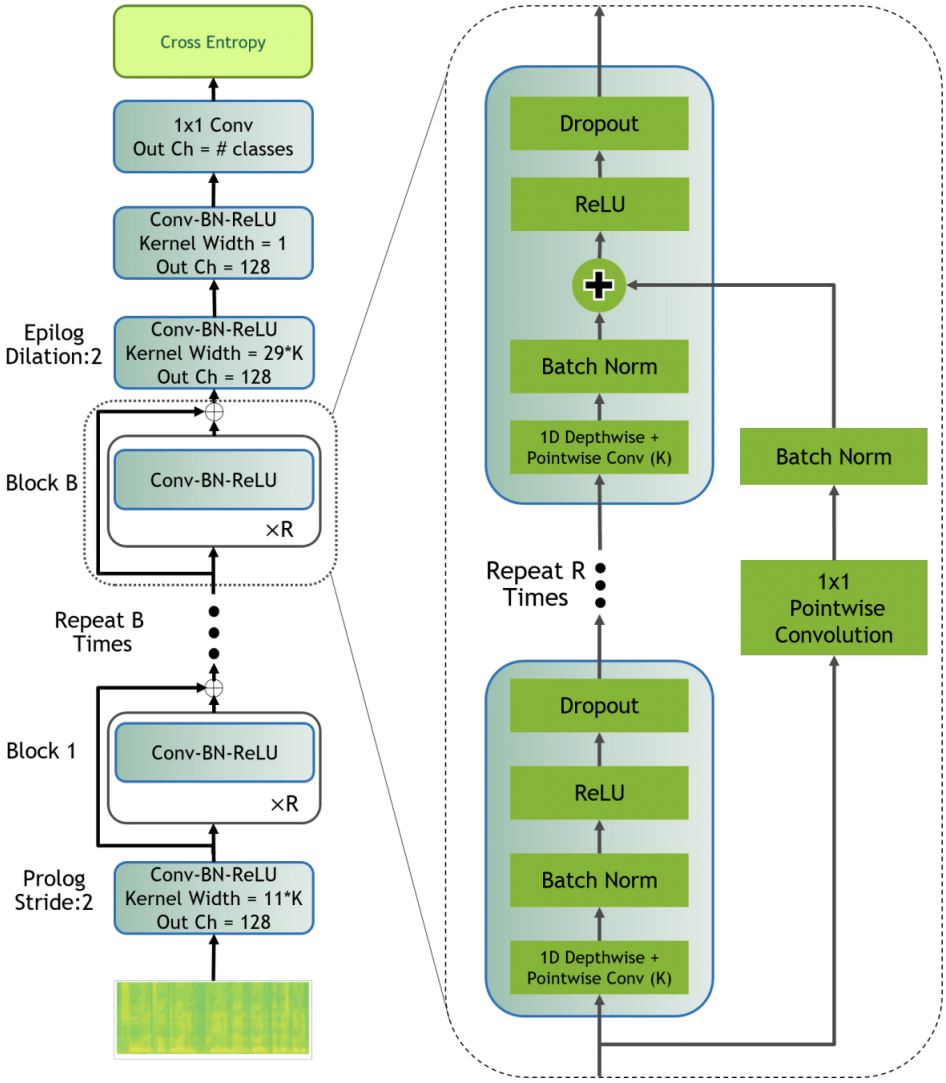


Figure 3.1: MatchboxNet BxRxC model: B - number of blocks, R - number of sub-blocks, C - the number of channels. Diagram Majumdar and Ginsburg (2020)

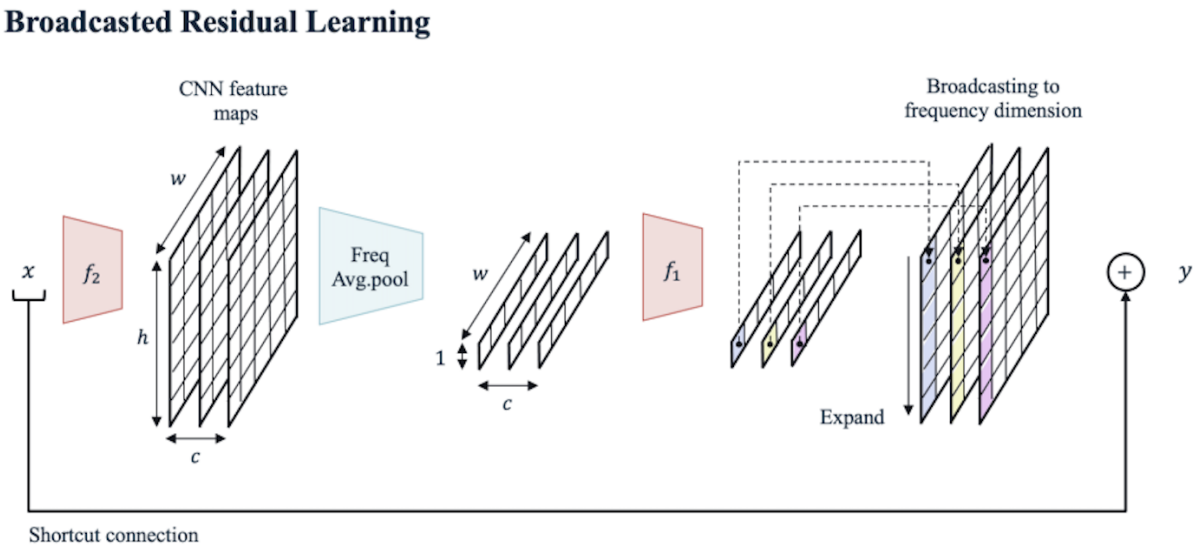


Figure 3.2: From Kim *et al.* (2021*a*):Broadcasted Residual Learning, where x E Rexhxw with number of channels c.

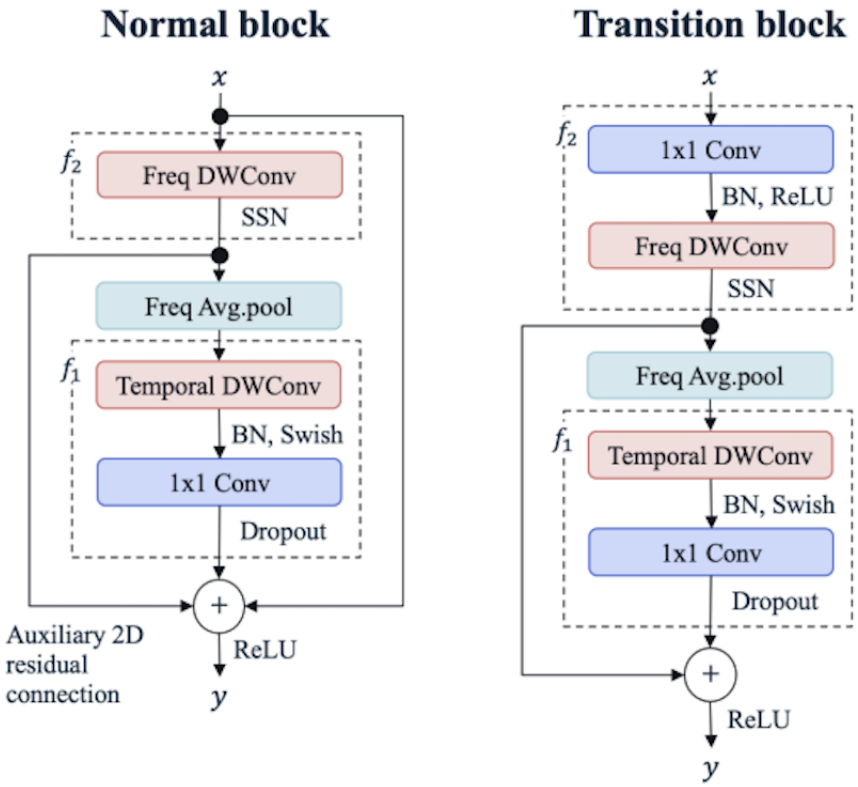


Figure 3.3: From Kim *et al.* (2021*a*):BC- ResBlock. The BC-ResNet block contains a frequency-depthwise convolution with a SubSpectralNorm. Then the feature is averaged by frequency followed by temporal-depthwise separable convolution. The temporal feature is broadcasted to 2D features at residual connection. In a transition block, we have an additional 1x1 convolution on the front to change the number of channels without identity

3.4.3 Model Training

Each model is trained for 200 epochs with cross-entropy loss, ADAM optimizer, learning rate (LR) of 1e-4, exponential-decay (γ = .95) LR scheduler, batch size set to 256 using a single Nvidia RTX3090 GPU. Inputs to the model are 40-dimensional log Mel-spectrograms (or its quaternion) extracted with a window of size 30ms and a shift of 10ms. Similar to Kim *et al.* (2021*b*), the data augmentation strategy included time-shift, additive background noise, and SpecAugment. For a fair comparison or benchmarking, we retrained all models (existing & proposed) with exactly the same experimental setup.

Table 3.1: Comparison of accuracies for ResNet Class of Models. Accuracy (%) is averaged over 5 trials (95% confidence interval). The number inside {} denotes the number of input channels.

Model	**Param**	**Feature**	**Accuracy**
ResNet18-2D {1}	12.6M	MelSpec	96.52
ResNet18-2D {4}	12.6M	MelSpec + derivatives	94.45
EfficientNet-2D {4}	4M	MelSpec + derivatives	96.48
QResnet18-2D {1}	3.2M	QMelSpec	96.77
QResnet18-1D {1}	1M	QMelSpec	96.41

Table 3.2: Comparison of accuracies for MatchboxNet Class of Models. Accuracy (%) is averaged over 10 trials (95% confidence interval). The number inside {} denotes the number of input channels.

Model	**Param**	**Feature**	**Accuracy**
MatchboxNet {1}	80K	MelSpec	96.91
MatchboxNet {4}	80K	MelSpec + derivatives	95.34
MatchboxNet {1}	140K	MelSpec	97.37
QMatchboxNet {1}	30K	QMelSpec	96.20
QMatchboxNet {1}	80K	QMelSpec	96.69
QMatchboxNet {1}	140K	QMelSpec	97.41

3.5 Results and Analysis

3.5.1 Experiments with ResNet18

The results of these experiments are reported in Table 3.1, where we trained a 2D-convolution based ResNet18 and its quaternion version (QResNet18). It can be observed that QResNet18 achieves slightly higher performance than ResNet18 at approx 4$\times$ reduction in model parameters. Also, note the decrease in performance when

Table 3.3: Comparison of accuracies for BCResNet Class of Models. Accuracy (%) is averaged over 10 trials (95% confidence interval). The number inside {} denotes the number of input channels.

Model	**Param**	**Feature**	**Accuracy**
BCResNet {1}	10K	MelSpec	96.90
BCResNet {1}	360K	MelSpec	98.70
BCResNet {4}	360K	MelSpec + derivatives	96.69%
QBCResNet {1}	11K	QMelSpec	97.11
QBCResNet {1}	320K	QMelSpec	98.62
No Augumentation			
BCResNet {1}	11K	MelSpec	89.34
BCResNet {4}	11K	MelSpec + derivatives	87.23
QBCResNet {1}	11K	QMelSpec	92.42

ResNet18 is trained with 4-channel input, i.e., the model cannot model the complex relationships in the multichannel input. Using a complex architecture (with better expressivity & generalization) such as EfficientNet helps recover some of that performance back. However, searching for such an optimal task-specific architecture might be difficult in practice. QResNet18 on the other hand, can effectively model the multiview information in quaternion input. In fact, the 1D-QResNet18 with 1D kernels resulting in just 1M parameters achieves comparable performance to 2D-QResNet18. This demonstrates the expressive power of QCNNs at medium-to-low parameter regimes.

3.5.2 Experiments with MatchboxNet

The results of these experiments are reported in Table 5.3. MatchboxNet is a SOTA KWS model based on 1D-DSC that achieves the best performance of 97.37% with just 140K parameters. The more popular MatchboxNet, in practice with 80k parameters, also performs reasonably well. However, again note that the model fails to exploit the information in multi-channel input, leading to an absolute 1.57% reduction in performance. Its quaternion version (QMatchboxNet) not only achieves comparable performance but is robust to degradation in an extremely low parameter regime of only 30K parameters.

3.5.3 Experiments with BCResNet

In this experiment, we have considered BCResNet and its equivalent-sized quaternion version (QBCResNet). BCResNet utilizes the advantage of 2D & 1D-DSC layers simultaneously, thus achieving good performance in an extremely low parameter regime of only 10K parameters. To the best of the author's knowledge, BCResNet, with 320K parameters, has the best accuracy of 98.7% so far. However, compared to MatchboxNet, BCResNet has a high inference time due to a significantly higher number of arithmetic operations (refer to Kim *et al.* (2021*b*) for more details). Table 3.2 shows a performance comparison of BCResNet and QBCResNet. As expected and reported in the previous Sections 3.5.1 & 3.5.2, we observe similar trends where QBCResNet performs comparably to BCResNet for similar parameter budgets. In addition, we also demonstrate how crucial data augmentation is at an extremely low-parameter regime. For this, we retrained equally sized both BCResNet and QBCResNet models without any time or spectral augmentation. It can be observed performance of BCResNet drops significantly by 4% as compared to QBCResNet.

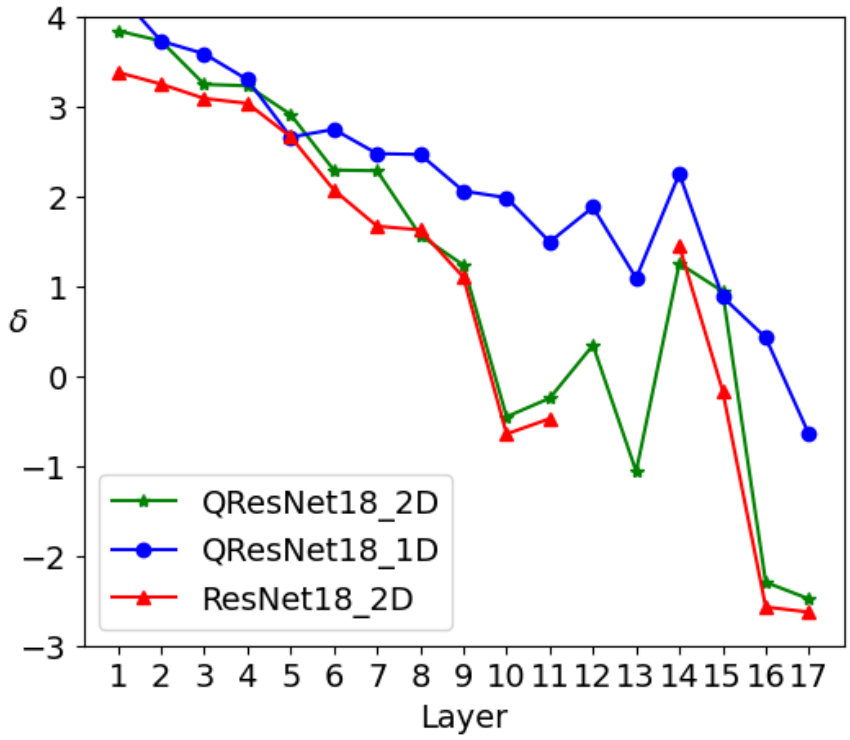


Figure 3.4: LogSpectralNorm of convolutional layers of ResNet and QResNet models. Missing values in a few layers of ResNet18 are due to scale collapse.

3.6 Data-Independent Model Analysis

To study and quantify the quality of a model, we compare ResNet18 and QResNet18 using a data-dependent approach proposed in Martin *et al.* (2021). In particular, for each layer, we compute the Log spectral norm measure of the weight matrix $\mathbf{W}$ as

$$\delta = \log \|\mathbf{W}\|_\infty = .5 \log \lambda_{max}$$

where λ_{max} is the maximum eigenvalue of $\mathbf{W}$. Figure 3.4 shows the δ measure computed for the convolutional layers for both models. Classification models by design are non-homeomorphic maps since the layer-by-layer latent space has to collapse the space of inputs (for each class) to a single point (class label). Hence, we see that the measure δ on average monotonically decays from a higher value in the initial layers to a small one in the penultimate layers. It can be observed that both ResNet and its quaternion version exhibit similar behavior, which confirms that they are trained well Martin *et al.* (2021). However, ResNet, compared to QResNet18, is highly over-parametrized and thus suffers from Scale Collapse, i.e., a few layers have unusually small spectral norms. To quantify this, we trained both ResNet18 and QResNet18 with different initialization seeds and measured if scale collapse occurred in the model. We observe that for the KWStask considered in this work, QResNet18 never exhibits scale collapse, while the over-parametrized ResNet18 exhibits this for approximately 5% of the time. This behavior is unsurprising given that quaternion convolution learns from an intermix of information in the multiview inputs leading to stable and robust models. Our future work will extend the exploration to data-dependent analysis via tools such as those presented in Muckenhirn *et al.* (2019); Gupta and Abrol (2022).

3.7 Conclusion

In this chapter, we present Quaternion neural networks for robust KWS on devices with limited computational and memory resources. We demonstrated how quaternion models are able to learn complex relationships from multiple views of the input data and consistently better than multichannel input counterparts in the case of conventional models. Experiments on the Google Speech Commands v2 dataset show quaternion models

achieve comparable performance with significantly fewer parameters than models with similar accuracy.

CHAPTER 4

Study II: Pruned QCNNs for efficient Audio Tagging

4.1 TLDR

This study presents a novel approach to make convolutional neural networks (CNNs) efficient by reducing their computational cost and memory footprint. Even though large-scale CNNs show state-of-the-art performance in many tasks, high computational costs and the requirement of a large memory footprint make them resource-hungry. Therefore, deploying large-scale CNNs on resource-constrained devices poses significant challenges. To address this challenge, we propose to use quaternion CNNs, where quaternion algebra enables the memory footprint to be reduced. Furthermore, we investigate methods to reduce the memory footprint and computational cost further through pruning the quaternion CNNs. Experimental evaluation of the audio tagging task involving the classification of 527 audio events from AudioSet shows that the quaternion algebra and pruning reduce memory footprint by 90% and computational cost by 70% compared to the original CNN model while maintaining similar performance.

4.2 Introduction

The everyday acoustic environment contains a diverse array of sounds, from traffic and construction to subtle noises like keys jangling, music playing, or conversations Virtanen *et al.* (2018). This rich soundscape holds vast information useful for surveillance, healthcare, and improving environments in workplaces or cities Abrol and Sharma (2020). Techniques for recognizing these sound activities, known as audio tagging, involve capturing environmental sounds with microphones and identifying different activities Gemmeke *et al.* (2017).

In recent years, commercial interest in artificial intelligence (AI) integration into edge devices for personalized experiences and privacy solutions Su *et al.* (2022) has

reshaped user interactions, emphasizing the demand for low-latency, memory-efficient AI models Cheng *et al.* (2018). Although recent advancements in deep learning (DL), especially convolutional neural networks (CNNs), have shown remarkable performance in tasks like audio tagging. However, large-scale CNN models present challenges due to their resource-intensive nature, requiring substantial computational power and memory resources. For example, cutting-edge CNN frameworks like pre-trained audio neural networks (PANNs) Kong *et al.* (2020*b*) have 81M parameters, occupying 312MB of memory, and require over 20G computations to process 10 seconds of audio. The high computational complexity results in high inference latency and energy inefficiency, especially considering the large memory footprint required by CNNs. Consequently, deploying such large-scale CNNs on mobile phones or IoT devices presents substantial challenges due to their limited computational and memory capabilities.

The most popular method for enhancing CNN architecture efficiency is model compression Cheng *et al.* (2018). Typically, model compression involves simplifying or removing elements of CNN models by filter pruning methods, where few learned filters contributing least to performance are pruned away from CNNs Liang *et al.* (2021). An alternative method is to use knowledge distillation (KD) from large-scale CNNs to design low-complexity, smaller memory footprint student models. However, selecting an optimal student network involves intricate engineering, and network size tuning and demands substantial amounts of unlabeled data and entails significant training costs Lagunas *et al.* (2021); Xia *et al.* (2022).

In this work, we explore Quaternion convolutional neural networks (QCNNs) Zhu *et al.* (2018) in combination with architectural refinement through filter pruning Li *et al.* (2017) as an alternative for effective acoustic modeling for the audio-tagging task. While convolutional operation in CNNs operates with real-valued data and convolutional filters, QCNNs operate with quaternion data and quaternion filters, which are hypercomplex numbers consisting of a real part and three imaginary parts Kumar *et al.* (2023). Unlike multi-channel models that ignore internal correlation among channels, QCNNs can embed various facets of input features along with spatial transformations and can better learn the complex relationships within them Qiu *et al.* (2020). QCNNs employ the Hamilton product instead of the standard dot product, which results in generalization across different orientations, scales, and translations more effectively with very few parameters that are shared across channels Parcollet *et al.* (2019). An illustra-

tion of transforming a conventional neural network to a quaternion domain that results in reducing total parameter count is shown in Figure 2.1. More details about the quaternion transformation are explained in Section 4.4. We have established that QCNN models have recently gained traction in speech/audio tasks such as speech recognition Parcollet *et al.* (2018*b*), event detection Comminiello *et al.* (2019), and keyword-spotting Chaudhary and Abrol (2023). However, a QCNN layer requires more computations than a traditional CNN layer. To mitigate the computational complexity inherent in QCNNs, we implement a strategy of pruning quaternion filters that minimally impact the network's output performance. The filter pruning process involves the targeted removal of entire quaternion filters and their associated quaternion output. Such a methodical reduction not only decreases the computational complexity but also leads to a significant decrease in the number of parameters within the QCNNs. As a result, we achieve a streamlined version of QCNNs, termed Efficient QCNNs (E-QCNNs), which can yield similar performance while operating with reduced computational overhead and enhanced parameter efficiency compared to original CNNs. Further, we demonstrate that pruning large-scale PANNs alone achieves suboptimal performance compared to the proposed E-QCNN model that leverages quaternion algebra and pruning to reduce computations and memory storage. To this aim, we used the popular large-scale AudioSet benchmark Gemmeke *et al.* (2017) to validate our approach using CNN14, one of the best-performing PANNs models Kong *et al.* (2020*b*). The key advantages and major contributions of this paper can be summarized as follows:

- We propose a simple yet effective framework to reduce parameter count in CNNs by transforming CNNs to QCNNs by leveraging quaternion algebra. Further, we apply pruning to obtain efficient QCNNs (E-QCNNs) to reduce the computational complexity of QCNNs and the parameter count.
- Our framework reduces computations per inference by 70% with 90% fewer parameters with a marginal reduction in performance compared to that of the original CNN.
- Empirically, we demonstrate that E-QCNN obtained via leveraging both quaternion algebra and pruning is advantageous in terms of improved performance, fewer computations, and smaller parameter counts compared to obtaining efficient CNNs by applying pruning alone.

4.3 Related Works

The field of audio tagging has seen significant advancements with the introduction of the AudioSet dataset Gemmeke *et al.* (2017), a comprehensive collection of over 2 million audio clips labeled across 527 sound classes. Popular deep learning models for audio tagging include large-scale convolutional neural networks (CNNs), recurrent neural networks (RNNs), and their variants, which have demonstrated significant improvements over traditional machine learning methods Kong *et al.* (2020*b*). Audio-visual learning with contrastive audio-visual masked autoencoder methods has further helped improve the tagging accuracy of these models as they enable a model's joint learning in audio and visual space Gong *et al.* (2022). Large-scale transformer models and capsule networks have also shown promising results for this task both in supervised and unsupervised settings. The current state-of-the-art (SOTA) for audio tagging include multimodal transformer architecture based autoencoder Gong *et al.* (2022) and model agnostic methods based on pretraining, balanced sampling, data augmentation and label enhancement Gong *et al.* (2021). Although current SOTA models are accurate for audio tagging, due to high computational complexity and large memory requirements, such well-performing models are difficult to deploy on edge devices as optimizing such complex architectures for edge devices is not feasible.

Recently, there has been a focus on reducing the computational and memory requirements of CNNs through pruning techniques. Pruning methods involve eliminating a subset of parameters, such as weights or convolutional filters, from CNNs that contribute minimally to their overall performance Denil *et al.* (2013); Frankle and Carbin (2019). For example, Li et al. Li *et al.* (2017) found eliminating 64% parameters that yield 34% computations does not affect the performance much. Pruning can be performed either by eliminating whole convolutional filters Luo *et al.* (2018) or by eliminating individual weights Wen *et al.* (2016). In the former case, the resultant network is a structured pruned model, where cross-platform inference is supported in off-the-shelf libraries. However, the later case results in an unstructured sparse pruned model that requires specialized software or hardware for speed-up Han *et al.* (2016). Therefore, filter pruning methods are advantageous compared to weight pruning methods. Most filter pruning methods are active, where the importance of the filters is measured using a training dataset. Such active filter pruning methods either involve joint optimization

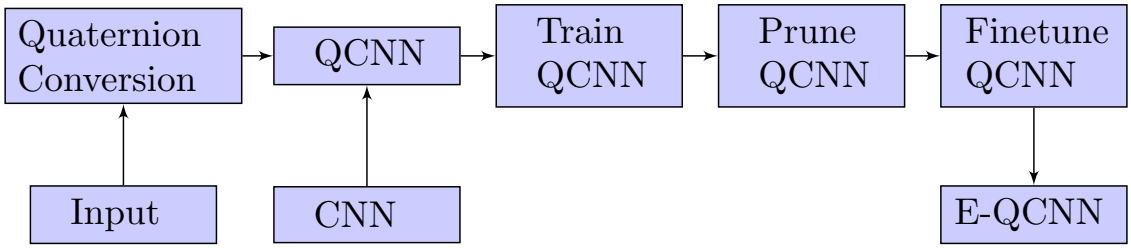


Figure 4.1: Proposed framework to obtain an E-QCNN model.

of network parameters while computing filter importance Lin *et al.* (2019); Luo and Wu (2020) or use filter outputs to measure importance Lin *et al.* (2020); Yeom *et al.* (2021). In contrast, passive pruning methods are data-independent as they directly compute importance using trained filter weights Li *et al.* (2017); Singh *et al.* (2023).

4.4 Efficient Quaternion CNN (E-QCNN)

4.4.1 Building E-QCNN model for Audio Tagging

The overall framework for obtaining an E-QCNN model is illustrated in Figure 4.1, which comprises three stages, namely 1) representing input data in the quaternion domain and transforming the original CNN architecture into QCNN; 2) training the QCNN to achieve comparable performance; and 3) fine-tuning the pre-trained QCNN model after pruning irrelevant filters.

4.4.2 Pruning Quaternion Filters

Parameter saving in a QCNN doesn't result in lower computational complexity in addition/multiplication operations due to shared kernel interactions across various channel groups. To address this, we propose to use filter pruning as a post-processing step to reduce the computational complexity inherent in QCNNs with minimal impact on the network's performance. In this work, we extend the passive filter pruning method from Li et al. Li *et al.* (2017) to QCNNs due to its simplicity, where the pruning is achieved in a layer-by-layer manner. Given a set of QCNN filters, we prune away p least important quaternion filters along real, $\mathbf{i}, \mathbf{j}$ and $\mathbf{k}$ channels independently. The importance of the filters along each channel is measured using their sum of absolute coefficients or

ℓ_1-norm. As demonstrated by Li et al. Li *et al.* (2017), we observed that quaternion filters exhibit similar behavior, i.e., filters with smaller ℓ_1-norm produce output with weak activation compared to other quaternion filters. Hence, such filters are less significant, and eliminating them has a negligible effect on the performance in practice. After eliminating p least significant quaternion filters along each channel in l^{th} layer, we apply the same procedure for other layers as well. After applying pruning across various intermediate layers, we obtain a pruned network that has a reduced number of parameters and computations compared to that of vanilla QCNN. In the end, we perform a final fine-tuning step on the E-QCNN to regain most of the performance lost due to the pruning procedure.

4.5 Experiment Setup

This section provides a system description, experimental protocol, and the dataset used in the experimental study.

4.5.1 Dataset

All experiments in this work are performed on the AudioSet dataset Gemmeke *et al.* (2017). The dataset contains 2 million 10-second-long utterances sampled with 527 audio event classes. This provides a benchmark for designing and evaluating audio models that identify individual audio events covering various human and animal sounds, musical instruments and genres, and everyday environmental sounds. We use the standard train/validation/evaluation set and report the average mAPs over 3 trials on the evaluation set. In all experiments, the standard deviation in mAPs across trials is found to be approximately 10^{-3}.

4.5.2 Evaluation Metric

We report a model's performance using the mean average precision (mAPs) metric, a popular robust metric for imbalanced datasets like AudioSet. In addition, we compare computational efficiency using parameter count and multiply-accumulate opera-

tions (MACs) of various CNN models.

4.5.3 Model Architectures

In this work, we experimented with CNN14, a well-performing model from PANNs Kong *et al.* (2020*b*) as our baseline model with 0.431 mAPs on the AudioSet evaluation set. This model consists of a total of 12 convolutional layers, among which the last six layers contribute more than 90% of the total 81M parameters (with 21G MACs). CNN14 is pre-trained on AudioSet using LogMel spectrograms of size (1000×64) from 10s long audio inputs sampled at 32KHz with a window size of 1024 samples and a hop size of 320 samples. The quaternion versions of CNN14, i.e., QCNN & E-QCNN are trained with LogMel spectrograms-based acoustic quaternions as inputs.

To have an extensive comparison, we also have considered a comparison with various other CNN architectures (with different parameter budgets); namely plain CNNs (CNN6, CNN10) Kong *et al.* (2020*b*), residual networks (ResNet22 Kong *et al.* (2020*b*), ResNet54 Kong *et al.* (2020*b*), DeepRes Ford *et al.* (2019)), efficient networks (MobileNetV1 Kong *et al.* (2020*b*), MobileNetV2 Kong *et al.* (2020*b*), MobileNetV3 Howard *et al.* (2019); Schmid *et al.* (2023), EfficientNet-B2 Gong *et al.* (2021); Tan and Le (2019)) and E-PANNs a pruned only variant of CNN14 Singh *et al.* (2023).

4.5.4 QCNN model Training and Pruning

The training or fine-tuning procedure for QCNN is similar to CNN14 as described in Kong *et al.* (2020*b*). Each QCNN model is trained until the performance converges to CNN14 performance, or the QCNN model is trained for 500k epochs with cross-entropy loss, ADAM optimizer, learning rate (LR) of 1e-3, batch size set to 32 using a single NVIDIA RTX 3090-24GB GPU.

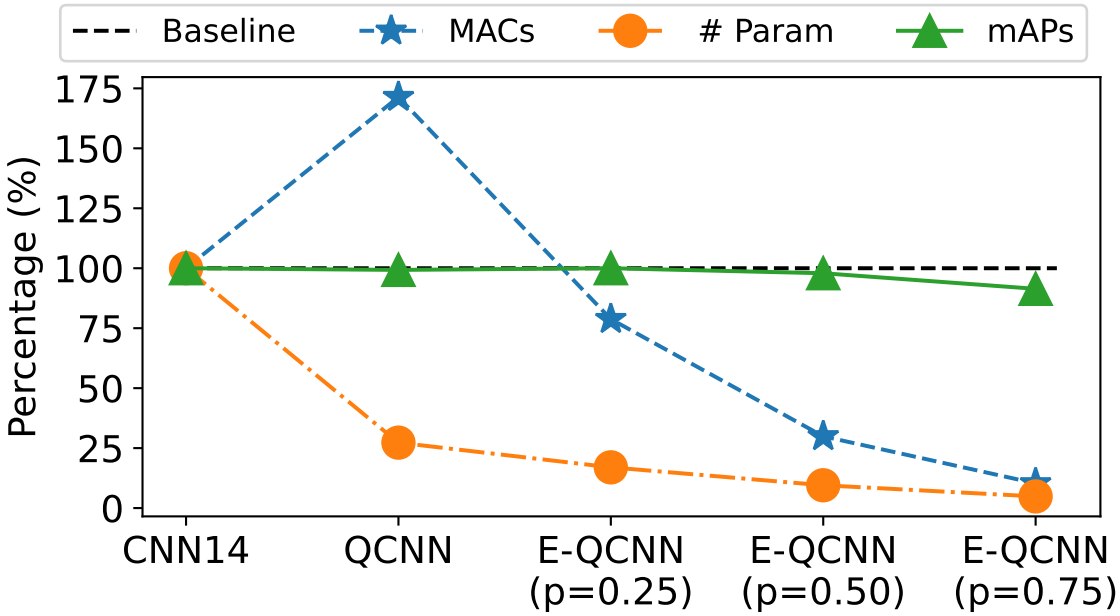


Figure 4.2: Relative comparative performance of QCNN and E-QCNN for various pruning rates with baseline CNN14 model. Here, 100% is equivalent to baseline performance with (MACs, # Param, mAPs) equals to (21G, 81M, 0.431).

4.6 Results and Analysis

4.6.1 Comparison with the baseline System

We first compare the baseline CNN14 with QCNN and E-QCNN in terms of mAPs, model parameters and MACs. The results of this experiment are reported in Figure 4.2. It can be observed that transforming CNN14 into the quaternion domain (QCNN) reduces parameter count by approximately 73% while achieving competitive performance with only 0.3 percentage points reduction in mAPs. However, this comes at approximately a 1.6× increase in computational complexity for the QCNN model compared to that of CNN14. As expected, the computational complexity is addressed in E-QCNN, where a good trade-off is achieved between the pruning rate and mAPs. For instance, pruning 25% quaternion filters from QCNN reduces parameter count and MACs by 83% and 25%, respectively, without any performance loss. Similarly, pruning 50% quaternion filters reduce parameter count and MACs by 90% and 70%, respectively, with only a 0.9 percentage points reduction in mAPs compared to that of CNN14. On the other hand, pruning 75% quaternion filters from QCNN reduces the performance by 3.6 percentage points at the expense of reduction in the parameter count and the MACs

by 95% and 90%, respectively, compared to that of CNN14.

In addition, we analyzed the convergence behavior of the proposed compressed QCNN models. As demonstrated in Figure 4.3, QCNN/E-QCNN models exhibit stable convergence over epochs during training or fine-tuning stages. Such a behavior is often challenging to achieve for structurally compressed models in practice Chaudhary and Abrol (2023), which motivates the use of quaternion models in other speech/audio applications in the future.

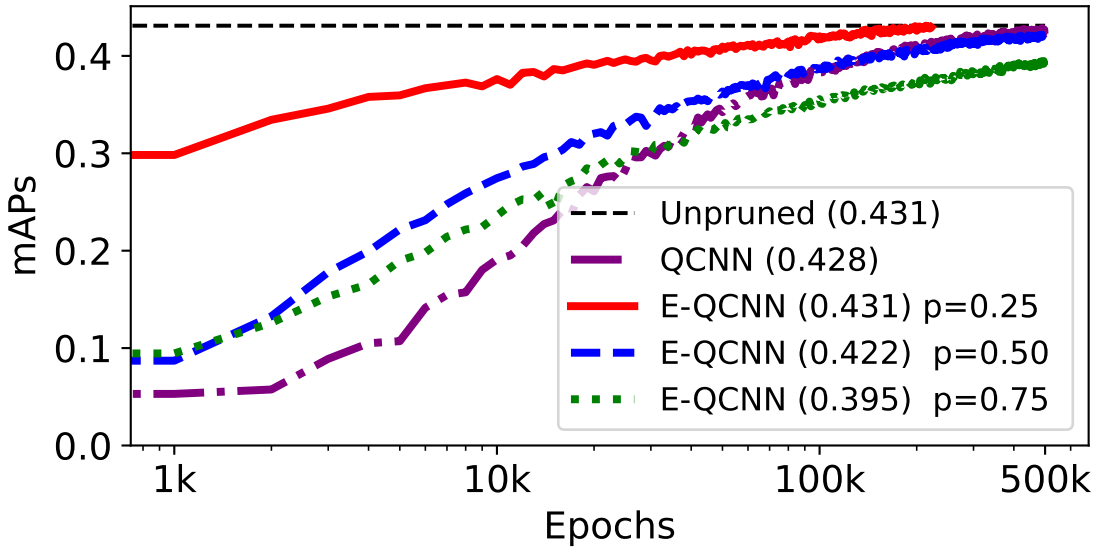


Figure 4.3: mAPs obtained while training or fine-tuning process for QCNN and pruned QCNN at different p. mAPs inside ()

4.6.2 Comparison with Existing Models

In this experiment, we compare the performance trade-off (mAPS vs parameter count) of the proposed QCNN models and existing popular, efficient architectures. The results of this experiment are reported in Figure 4.4. It can be observed that among the top-performing ResNet & CNN14 models, E-QCNN ($p = 25\%$) has the lowest parameter count. Similarly, among existing low-footprint models, namely MobileNetV1/V2, CNN6/10, EfficientNet-B2, and DeepRes models, E-QCNN ($p = 75\%$) has the best mAPs and lowest parameter count. MobileNetV3, an advanced architecture optimized using computationally expensive neural architecture search, exhibits a slightly higher performance (mAPs) at the expense of a few more parameters. It is worth highlighting that for a similar parameter budget, the proposed E-QCNN achieves better performance

(mAPs) than E-PANNs (pruned only CNN14) and low-footprint CNN6 models. This suggests the benefits of obtaining a low-footprint and efficient CNN from a large-size CNN by applying both quaternion transformation and pruning compared to applying pruning alone or training similar-size CNN from scratch.

Table 4.1: Performance comparison with other similar CNN-based existing methods

Network	**Parameter Count**	**MACs**	**mAPs**
ResNet22	63.67M	30G	0.430
ResNet54	104M	54.56G	0.429
QCNN14 (p=25%)	**13.75M**	**16.70G**	**0.431**
CNN6	4.84M	22G	0.343
CNN10	5.22M	28G	0.380
E-PANNs (CNN14) Singh *et al.* (2023)	4.54M	10.79G	0.340
MobileNetV1	4.79M	3.6G	0.389
MobileNetV2	4.07M	2.8G	0.383
MobileNetV3 Howard *et al.* (2019); Schmid *et al.* (2023)	4.88M	0.54G	0.401
DeepRes Ford *et al.* (2019)	26M	-	0.392
EfficientNet-B2 Tan and Le (2019); Gong *et al.* (2021)	9.19M	-	0.3818
QCNN (p =75%)	**3.94M**	**2.18G**	**0.394**

4.7 Conclusion

In this study, we introduce Quaternion neural networks optimized for efficient audio tagging on devices constrained by computational and memory capacities. We illustrate how quaternion models adeptly capture complex interactions across various perspectives of the input data, outperforming their multichannel counterparts typically used in traditional models. Through experiments conducted on the AudioSet dataset, we demonstrate that quaternion models attain similar levels of performance with markedly fewer parameters compared to other models achieving equivalent accuracy. Furthermore, we explore the application of pruning techniques within Quaternion Convolutional Neural Networks (QCNNs) to further reduce the model size without significantly impacting its ability to accurately tag audio content, thereby enhancing the feasibility of deploying advanced audio analysis models on resource-limited devices.

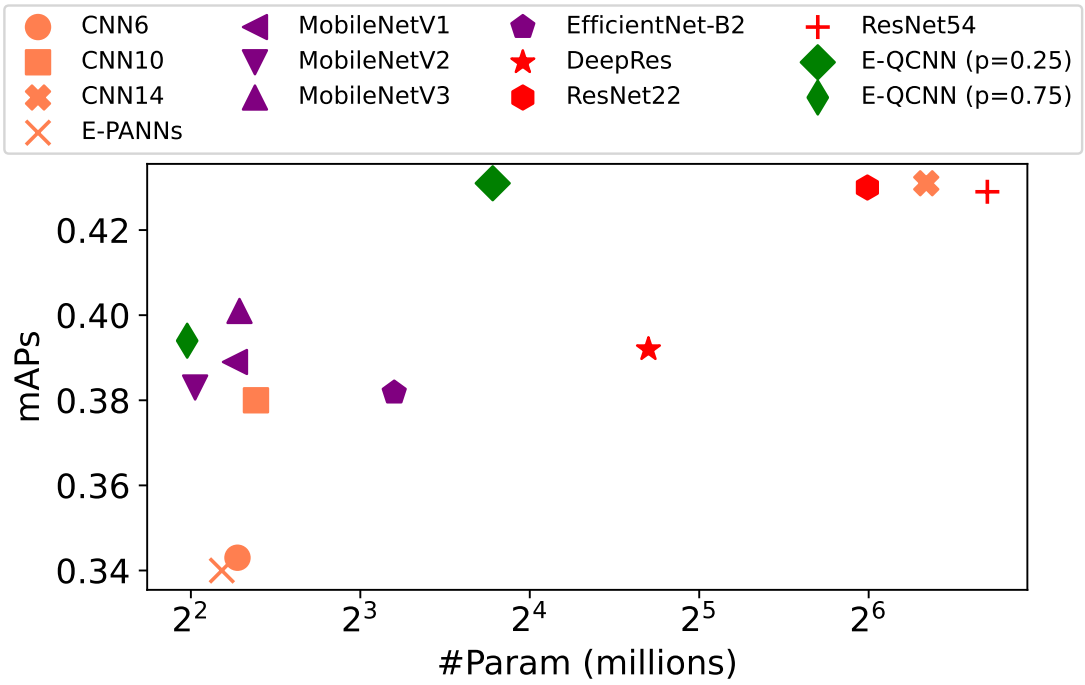


Figure 4.4: mAPs versus parameter count for various audio tagging models.

CHAPTER 5

Study III: Quaternion Neural Vocoder for Speech Synthesis

5.1 TLDR

Neural vocoders have recently evolved achieving superior synthesis quality by leveraging advancements in methods like Diffusion, Flow, Transformers, GANs, etc. However, such models have grown vastly in terms of space and time complexity leading to challenges in the deployment of speech synthesis systems in resource-constraint scenarios. To address this we present a novel low-footprint Quaternion Generative Adversarial Network (QGAN) for efficient and high-fidelity speech synthesis without compromising on the audio quality. QGAN achieves structural model compression over conventional GAN with quaternion convolutions in the generator and a modified multi-scale/period discriminator. To ensure model stability we also propose a weight norm in the quaternion domain. We show the effectiveness of QGAN with large-scale experiments on English and Hindi language datasets. In addition, using loss landscape visualization we provide an analysis of the learning behavior of the proposed QGAN model.

5.2 Introduction

The pursuit of creating an artificial speech that sounds realistic and natural has long been a major goal in the field of speech synthesis. In recent years, with the advent and evolution of deep learning technologies, vocoder-centric, neural network-based systems have dramatically enhanced the quality of speech synthesis. Neural vocoders are pivotal in translating intermediate speech representations into intelligible and natural-sounding speech. These innovative models have explored a diverse range of methodologies, including autoregressive models van den Oord et al. (2016); Kalchbrenner and

et al. (2018), non-autoregressive Flow Prenger *et al.* (2019); Ping *et al.* (2020), Diffusion Huang *et al.* (2022); Chen and et al. (2021), Variational Autoencoders (VAE) Peng *et al.* (2020), and Generative Adversarial Network (GAN)-based models Kong *et al.* (2020*a*); Kumar and et al. (2019). However, the high computational and storage demands of all these models pose significant challenges, particularly in deploying speech synthesis systems in resource-constrained settings. To address this critical limitation, researchers across domains like Speech, NLP, and Vision have utilized several model compression techniques to shrink the neural network footprint, achieving a trade-off between the size, accuracy, and reliability of the model. Popular methods include knowledge distillation, pruning, quantization, and low-rank decomposition Rokh *et al.* (2023); Gou *et al.* (2021); Sainath and et al. (2013). However, these methods involve intricate engineering or network size tuning to ensure reasonable/comparable performance, often demanding substantial amounts of unlabeled data and requiring significant training costs.

In this work, we present a novel quaternion structural adaptation to the GAN-based synthesis framework, achieving a more compact design without sacrificing any aspect of audio quality Gaudet and Maida (2018*b*). In particular, leveraging recent advancements in quaternion convolutional neural models Tay *et al.* (2019*a*); Gaudet and Maida (2018*b*), we introduce the Quaternion Generative Adversarial Network (QGAN), a low-footprint neural vocoder that efficiently models complex relationships in input data to generate high-fidelity and intelligible audio. The proposed structural alteration to the GAN ensures a lossless one-to-one mapping that requires training the model from scratch only once, thus eliminating the need for existing low-rank based lossy compression or the requirement for a complex retraining/distillation pipeline. A low-parameter regime often requires normalization/regularization for training stability. Hence, we introduce the quaternion version of the weight-normalization and adapt quaternion spectral-normalization Farenick and Pidkowich (2003) to the proposed QGAN framework. We implement/adapt all other dependencies required for the quaternion domain conversion of conventional GAN models Goodfellow and et al. (2014), including layers in the generator, discriminator, and loss functions. In addition, we use the real to hypercomplex quaternion ($\mathbb{R} \rightarrow \mathbb{H}$) (and vice-versa) adaptor sub-networks in QGAN to facilitate the training/synthesis using real inputs/outputs.

We validated the effectiveness of the proposed QGAN model in producing realis-

tic, pristine and intelligible speech with experiments on the LJ Speech (English) and OpenSLR (Hindi) datasets. To the best of the author's knowledge, QGAN achieves the best trade-off between model size, inference speed and synthesis quality. In addition, we also demonstrate QGANN's generalization capability to perform high-quality synthesis on unseen speakers. These results not only showcase the versatility and adaptability of our method but also mark a significant step towards inclusive, multi-lingual speech synthesis technologies.

All the experiments are reproducible & the exact model architecture, training/testing recipes and pre-trained model weights will be freely accessible online `https://anonymousvocoders.github.io/`

5.3 Related Works

There exist various types of neural-based vocoders, each one having its own strengths and challenges. Initial vocoders like WaveNet van den Oord et al. (2016) and WaveRNN Kalchbrenner and et al. (2018) offer high voice quality but face inference time challenges due to their autoregressive nature. Flow-based vocoders like Parallel WaveNet van den Oord Aaron and et al. (2018) and ClariNet Ping *et al.* (2019) utilize teacher-student training but require significant computational resources. Vocoders like Glow Kingma and Dhariwal (2018) and RealNVP Dinh *et al.* (2017) offer even high-quality synthesis but demand extensive parameters due to deeper architectures. Similarly, Diffusion-based vocoders like DiffWave Kong *et al.* (2021), and WaveGrad Chen and et al. (2021), also show promising results but have slow inference speeds. Recently, a few attempts like FastDiff Huang *et al.* (2022) have been made to speed up these models given a target perceived audio quality. In practice, GAN-based vocoders are the preferred choice in many speech applications. Examples include WaveGAN Donahue *et al.* (2018), GAN-TTS et al. (2020), MelGANKumar and et al. (2019), Parallel WaveGANYamamoto *et al.* (2020) and HiFi-GANKong *et al.* (2020*a*). Among these, HiFi-GAN, utilising conventional GAN principles, innovative discriminators and loss functions, is a leading choice for achieving state-of-the-art speech synthesis while addressing computational demands. Some of the key advancements that have become de facto in GAN-based neural vocoders to enhance training stability and audio quality

include multi-scale generationYang *et al.* (2020), multi-period Kong *et al.* (2020*a*) & multi-scale Kumar and et al. (2019) discriminators, and feature matching loss Larsen and et al. (2016); Kumar and et al. (2019). In this work, we present QGAN that optimally blends recent advancements in low-footprint quaternion neural models and GAN vocoders to obtain the best trade-off in performance, quality, and efficiency for speech synthesis on resource-constrained devices.

5.4 QGAN Architecture

The QGAN proposed here takes cues from the widely recognized HifiGAN framework Kong *et al.* (2020*a*). Figure 5.1 illustrates the proposed QGAN architecture. It incorporates the multi-receptive field fusion technique Kong *et al.* (2020*a*) in its Generator design and capitalizes on the benefits offered by the Multi-scale and Multi-period discriminator's architecture. Through the integration of Quaternion algebra equipped convolution and transpose convolution layers Parcollet *et al.* (2018*b*), we achieve structural compression, resulting in a reduction of parameter count by a factor of 4. In contrast to real data, quaternion models operate with quaternion data and filters, which are hypercomplex numbers consisting of a real part and three imaginary parts Kumar *et al.* (2023). Unlike multi-channel models that ignore internal correlation among channels, QCNNs can embed various facets of input audio features along with spatial transformations and can better learn the complex relationships within them Gaudet and Maida (2018*b*). QCNNs employ the Hamilton product instead of the standard dot product, which results in generalization across different orientations, scales, and translations more effectively with very few parameters that are shared across channels Parcollet *et al.* (2018*b*). These models have recently gained traction in speech/audio tasks such as speech recognition Parcollet *et al.* (2018*b*), event detection Comminiello *et al.* (2019) and keyword-spotting Chaudhary and Abrol (2023). Readers are encouraged to refer to the supplementary section for more details.

5.4.1 QGenerator

The generator undergoes a complete transformation into a fully Quaternion convolutional neural network (QCNN), where a 4-channel LogMel-spectrogram based acoustic

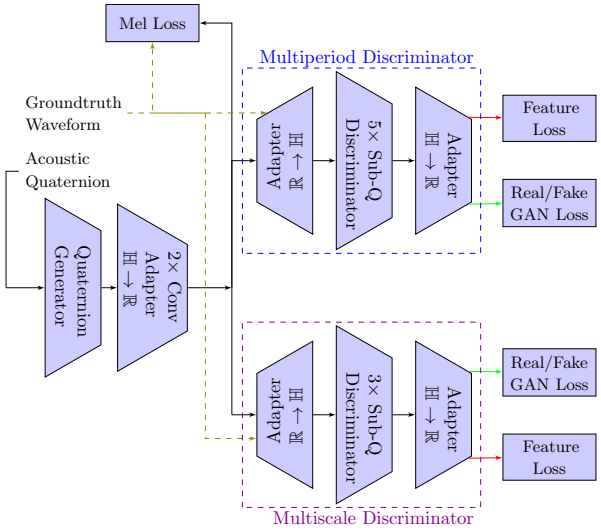


Figure 5.1: Illustration of the proposed QGAN Architecture

quaternion Parcollet *et al.* (2018*b*) serves as its input. Subsequently, this input undergoes upsampling via quaternion transposed convolutions, thereby aligning the temporal resolution of the resulting sequence with that of raw waveforms. Following each application of quaternion transposed convolution, a quaternion multi-receptive field fusion (QMRF) module is systematically deployed. Tailored specifically for the Quaternion domain, this module is adept at capturing patterns across varying lengths concurrently, thus enriching the network's representational capacity. QMRF module aggregates the outputs from multiple residual blocks, each employing distinct kernel size and dilation rate to simulate a range of receptive fields. We empirically demonstrate that leveraging both QCNN and QMRF modules allows us to train the generator to balance between synthesis efficiency and sample quality. Instead of directly mapping latent quaternion representations to raw waveforms, we employ an adaptor module. This sub-network comprising regular conventional convolutional layers, adjusted to the incoming quaternion channels, transforms quaternion outputs to real-valued waveforms.

5.4.2 QDiscriminator

We employ dual-discriminator strategy that enables effective recognition of both the periodic nature and the extended correlations inherent in the audio samples. In particular, the proposed QDiscriminator comprises 1) a quaternion multi-period discriminator (QMPD), drawing inspiration from HiFiGAN, for analyzing various periods within audio signals; and 2) a quaternion multi-scale discriminator (QMSD), akin to the approach

in MelGAN, to assess audio samples across different scales.

Quaternion Multi-period Discriminator

The QMPD employs a series of five sub-quaternion discriminators, each tailored to distinct periodicities present in the audio signals. Since QCNN filters in a layer interact with each channel individually Parcollet *et al.* (2018*b*), the quaternion discriminator can better capture internal correlation among a wide range of temporal sequences per channel, from short-term phonetic transitions to longer-term prosody features. To facilitate the transition between real-valued audio data and quaternion domains, we introduce real-to-quaternion and quaternion-to-real adaptor layers before & after each sub-discriminator, respectively. Each sub-quaternion discriminator within QMPD comprises strided quaternion convolutional layers, followed by a leaky ReLU split activation function. To enhance the overall stability and robustness, we also use quaternion weight-normalization adapted based on the normalization proposed in Salimans and Kingma (2016).

Quaternion Multi-scale Disicriminator

The proposed quaternion multi-scale discriminator (QMSD), inspired by the MelGAN architecture, complements the QMPD by consecutively analyzing audio sequences at multiple scales. QMSD employs three sub-discriminators operating at varied scales: original, $\times 2$ average-pooled, and $\times 4$ average-pooled. Similar to QMPD, each layer of QMSD employs strided quaternion convolution and leaky ReLU activation function. In addition, before/after each sub-discriminator adapter modules are used to ensure seamless domain transitions. Note that group convolutions in the original MelGAN MSD are not used in QMSD as quaternion convolutional operation by design is a combination of group and depthwise separable convolution due to the inherent division of input channels into one real and three imaginary channel groups Gaudet and Maida (2018*b*); Chaudhary and Abrol (2023). Compared to MSD, the proposed QMSD consists of fewer layers in each sub-discriminator. Most layers accompany quaternion weight normalization, and the first sub-discriminator also employs quaternion spectral normalization to ensure smoothed waveforms in QMSD.

Loss Function

The final objective for the proposed QGAN comprising of K sub-discriminators in QMSD/QMPD is defined as

$$\begin{aligned}
\mathcal{L}_G = \sum_{k=1}^{K} \big[& \mathcal{L}_{Adv}(G_Q; D_{Q_k}) + \lambda_1 \mathcal{L}_{FM}(G_Q; D_{Q_k}) \\
& + \lambda_2 \mathcal{L}_{Adv}(D_{Q_k}; G_Q) \big] + \lambda_3 \mathcal{L}_{Mel}(G_Q), \qquad (5.1) \\
\mathcal{L}_{Adv}(D_Q; G_Q) &= \mathbb{E}_{(x,s)} \left[(D_Q(x) - 1)^2 + (D_Q(G(s)))^2 \right] \\
\mathcal{L}_{Adv}(G_Q; D_Q) &= \mathbb{E}_s \left[(D_Q(G_Q(s)) - 1)^2 \right] \\
\mathcal{L}_{FM}(G_Q; D_Q) &= \mathbb{E}_{(x,s)} \Big[\sum_{i=1}^{L} \frac{1}{N_i} \| D_Q^i(x) - D_Q^i(G_Q(s)) \|_1 \Big], \\
\mathcal{L}_{Mel}(G_Q) &= \mathbb{E}_{(x,s)} \big[\| \phi(x) - \phi(G_Q(s) \|_1 \big],
\end{aligned}$$

where ϕ is the function that transforms a waveform into the corresponding Mel-spectrogram, L denotes the total layers in the Qdiscriminator, N_i denotes the number of features in the i-th layer of the discriminator, and λ_i are scaling constants. We use the least square adversarial loss $\mathcal{L}_{Adv}$ described in LS-GAN Mao *et al.* (2017). The QDiscriminator recognizes actual samples as 1 and generated ones as 0. Conversely, the QGenerator is optimized to improve the quality of its outputs and make them indistinguishable from real samples. The perception-based Mel loss $\mathcal{L}_{Mel}$ ensures that the frequency characteristics of the generated audio match those of the actual audio Kumar and et al. (2019); Arık *et al.* (2019). It is an additional reconstruction loss that, alongside the adversarial loss, guides the generator in producing high-fidelity audio. Finally, the feature matching loss $\mathcal{L}_{FM}$ helps improve the generator by comparing features extracted from real samples and those from intermediate layers of the discriminator Larsen and et al. (2016). We employ the ℓ_1 norm for comparing latent features. The aim is to ensure that the QGenerator produces samples with intermediate (quaternion to real) representations similar to real data, improving the quality of the generated output. This also helps capture inter-channel correlations to produce high-quality results while enhancing training stability by providing a richer gradient signal.

5.5 Experiment Setup

5.5.1 Dataset

We have trained two vocoders, one each using publicly available LJspeech[1] (English) and OpenSLR[2] (Hindi) speech dataset. English dataset contains 24 hrs of short audio clips of a single speaker split into train(80%)/validation(5%)/test(15%) set. The Hindi speech dataset contains spontaneous telephone speech recordings where the train (59 speakers) and test (disjoint 19 speakers) splits comprise 100 and 5 hrs of audio, respectively.

5.5.2 Model Configuration, Training and Input Representation

To evaluate the trade-off between generation speed and sample quality, we performed tests with 3 different versions of the QGenerator, namely V1, V2, and V3, while keeping the QDiscriminator the same. For a fair comparison, we repeat the same configurations for the baseline HiFiGAN with real inputs. **English Vocoder:** V1 was configured with a hidden unit size of 512, upsampling rates of [8,8,2,2], kernel sizes of [16, 16, 4, 4], dilation rates of [3, 7, 11], and dilation layers [[1, 1], [3, 1], [5, 1]] repeated 3 times. V2 is a scaled-down version of V1 with a reduced hidden unitsize of 128, yet retaining the same scope of receptive fields. V3 has a hidden unit size of 256, upsampling rates of [8,8,4], kernel size of [16,16,8], dilation rates of [3,5,7], and dilation layers [[1,2], [2,6], [3,12]] repeated 3 times. **Hindi Vocoder:** We only train the V1 model for the Hindi dataset due to computing constraints. We altered the upsample rates to [4,8,2,2] and kept the rest of the architecture same.

Each model is trained for 1000 epochs with total GAN loss, AdamW optimizer, learning rate (LR) of 1e-4, exponential-decay (γ = .99) LR scheduler, batch size set to 16 using a single Nvidia RTX3090 GPU. Inputs to the model are 80-dimensional LogMel-spectrograms (or its quaternion Parcollet *et al.* (2018*b*)) extracted with a window of size 30ms and a shift of 10ms. For a fair comparison or benchmarking, we retrained all models (existing & proposed) with exactly the same experimental setup.

[1] `https://keithito.com/LJ-Speech-Dataset/`
[2] `https://openslr.org/118/`

5.5.3 Evaluation Metric

Speech synthesis quality is gauged by PESQ (Perceptual Evaluation of Speech Quality)Rix *et al.* (2001), STOI (Short-Time Objective Intelligibility) Taal *et al.* (2010), and FAD (Fréchet Audio Distance) using CLAP & PANN embedding models Kilgour *et al.* (2019). In the case of PESQ & STOI, a higher value is better, while for FAD, a lower value is better. We have also used MOS(mean opinion score).

Perceptual Evaluation of Speech Quality (PESQ)

The Perceptual Evaluation of Speech Quality (PESQ) is a standard metric for assessing speech quality. It predicts the subjective listening quality experienced by users based on objective measurements. The PESQ score ranges from -0.5 to 4.5, where higher values indicate better perceived audio quality.

PESQ operates by comparing the original speech signal $s(n)$ to the synthesized or processed speech signal $\hat{s}(n)$, where n represents the time index. The comparison is conducted by modeling the human auditory system's perception of sound and taking into account various distortions that affect speech quality. The PESQ calculation involves a complex series of steps, which includes:

1. Aligning the reference and degraded signals to compensate for any timing offsets.
2. Applying perceptual weighting filters that mimic the human ear's response.
3. Computing the disturbance density and average disturbance values over time.
4. Mapping the results to the PESQ score through a regression model derived from subjective listening tests.

The PESQ score is calculated as follows:

$$\text{PESQ} = f\left(\int_{-\infty}^{+\infty} |S(\omega) - \hat{S}(\omega)|^2 \, d\omega\right)$$

where $S(\omega)$ and $\hat{S}(\omega)$ are the Fourier transforms of the original and synthesized signals, respectively, and f is a function representing the perceptual model.

Short-Time Objective Intelligibility (STOI)

Short-Time Objective Intelligibility (STOI) is an objective metric used to predict the intelligibility of speech signals. It is a measure of how well speech can be understood, especially in noisy environments, and is highly correlated with the intelligibility perceived by human listeners. The STOI score ranges from 0 to 1, with higher scores denoting clearer, more intelligible speech.

STOI works by comparing the short-time temporal envelopes of the clean speech signal $x(n)$ and the processed signal $y(n)$. These envelopes are computed using a short-time Fourier analysis. The correlation between the envelopes over multiple frequency bands is calculated and then averaged to obtain the STOI score. Mathematically, it can be represented as follows:

$$\text{STOI} = \frac{1}{J}\sum_{j=1}^{J}\left(\frac{\sum_{n=1}^{N}(x_j(n)-\bar{x}_j)(y_j(n)-\bar{y}_j)}{\sqrt{\sum_{n=1}^{N}(x_j(n)-\bar{x}_j)^2\sum_{n=1}^{N}(y_j(n)-\bar{y}_j)^2}}\right)$$

where $x_j(n)$ and $y_j(n)$ are the short-time temporal envelopes of the jth frequency band for clean and processed speech, respectively, $\bar{x}_j$ and $\bar{y}_j$ are their means, J is the total number of frequency bands, and N is the length of the envelope segment.

Fréchet Audio Distance (FAD)

Fréchet Audio Distance (FAD) is a measure of the quality of synthetic or processed audio signals. It assesses the similarity between distributions of features extracted from a reference audio dataset and from synthesized audio. The lower the FAD score, the closer the synthesized audio's feature distribution is to the reference, indicating better audio quality.

The FAD score is calculated using the Fréchet distance between two multivariate Gaussians:

$$\text{FAD} = |\mu_r - \mu_s|^2 + \text{Tr}\left(\Sigma_r + \Sigma_s - 2(\Sigma_r\Sigma_s)^{\frac{1}{2}}\right)$$

where μ_r and μ_s are the means of the feature vectors extracted from the reference

and synthesized audio, respectively, Σ_r and Σ_s are the corresponding covariances, and Tr represents the trace of a matrix.

Using embeddings from CLAP and PANN models, the feature distributions are determined, and the FAD score is calculated to quantify the distance between the two audio sources' representations.

Mean Opinion Score (MOS)

The Mean Opinion Score (MOS) provides an assessment of human-judged quality. It is a subjective metric used to gauge the experience of users. In the context of speech synthesis, MOS reflects the perceived naturalness, intelligibility, and overall quality of the synthetic speech.

Listeners are asked to rate the quality of speech samples on a scale, typically from 1 to 5, where a higher score represents better quality. The scores for each sample are then averaged to yield the MOS. A score of 1 may indicate poor quality with significant impairments, while a score of 5 signifies excellent quality indistinguishable from natural speech.

The mathematical representation of the MOS is as follows:

$$\text{MOS} = \frac{1}{M} \sum_{i=1}^{M} R_i \tag{5.2}$$

where R_i is the rating for the i-th speech sample, and M is the total number of ratings gathered.

Model Footprint

The Model footprint is assessed using model size with Parameter Count and inference efficiency with Multiplication-Accumulation operations (MACs). These metrics collectively offer a comprehensive picture of a model's efficiency and effectiveness in synthesizing speech.

Table 5.1: Parameters of HiFiGAN & QGAN vocoders trained on English [E] and Hindi [H] datasets

Model	Generator		Discriminator (MPD, MSD)	
	#Param (M)	MACs (GB)	#Param (M)	MACs (GB)
HiFiGAN V1 [E]	13.95	11.28	29.62, 41.11	82.27, 58.30
HiFiGAN V2 [E]	0.93	2.09	-	-
HiFiGAN V3 [E]	1.46	1.68	-	-
QGAN V1 [E]	3.71	9.01	25.62, 10.68	45.86, 21.85
QGAN V2 [E]	0.29	2.05	-	-
QGAN V3 [E]	0.48	1.57	-	-
QGAN V1 [H]	3.71	9.01	-	-

Table 5.2: Performance metrics of HiFiGAN & QGAN vocoders trained on English [E] and Hindi [H] datasets

Model	STOI	PESQ	FAD (CLAP, PANN)
HiFiGAN V1 [E]	0.976	3.700	0.0149, 1.77×10^{-5}
HiFiGAN V2 [E]	0.952	3.077	0.0271, 1.69×10^{-5}
HiFiGAN V3 [E]	0.951	2.972	0.0207, 5.40×10^{-6}
QGAN V1 [E]	0.9663	3.432	0.0169, 2.50×10^{-5}
QGAN V2 [E]	0.9342	2.704	0.0497, 1.09×10^{-5}
QGAN V3 [E]	0.9257	2.527	0.0583, 6.59×10^{-6}
QGAN V1 [H]	0.9350	3.523	0.0178, 1.79×10^{-5}

5.6 Results and Analysis

5.6.1 Comparison with baseline HiFiGAN

The results of this experiment on the unseen test are reported in Table 3.1, where we have mainly focused on the English vocoder due to computing constraints. It can be observed that the benchmarking underscores the prowess of QGAN, which achieves a $4\times$ reduction in both generator and discriminator parameters across its variants (V1, V2, V3). This significant compression does not compromise quality substantially, as evidenced by the minimal fluctuation in the PESQ, STOI and FAD scores. On the efficiency front, the performance gain for the HiFiGAN V1 over QGAN V1 comes at the expense of a $1.25\times$ increase in MACs. We believe hyper-parameter tuning for training, better computing hardware, and more data can easily bridge this performance gap. Further analysis of QGAN's ultra-compressed models reveals their strong performance when measured against HiFiGAN's V2 and V3 models. This analysis firmly positions QGAN as a highly efficient and competitive neural vocoder for resource constraint scenarios.

5.6.2 Synthesis Quality on Unseen Speakers

To assess the effectiveness of our QGAN model on unfamiliar speakers, we considered the trained QGAN V1 model. Table 3.1 also shows results for this model on 19 disjoint speakers from the OpenSRL (Hindi) dataset. It can be observed that despite its relatively minimal computational footprint, this model demonstrates commendable performance on spontaneous telephone-quality audio across the evaluated metrics. Note that intelligibility is very important for telephony applications that operate at low sampling rates compared to mainstream audio/speech applications. In addition. Indic languages present unique challenges due to their extensive range of syllables and the intricacies of phonetic information. These initial tests indicate a promising avenue for enhancing speech synthesis technology, making it more inclusive, multilingual, and representative of diverse linguistic profiles.

5.6.3 Comparison with existing Vocoders

In this experiment, we have considered popular low-footprint neural vocoders Jang *et al.* (2021). For a fair comparison, we retrained all models (existing & proposed) on the English dataset. Evaluation on the test-set is quantified using PESQ and subjective MOS scores (for 15 male & female speakers). Also, we consider parameter count only for the generator to compare model size. Results in Table 5.3 establish that for comparable model sizes, QGAN achieves the best performance, demonstrating notable scalability and efficiency.

Table 5.3: Comparative analysis of various low-footprint vocoders trained on English dataset.

Model	**MOS**	**PESQ**	**#Param (M)**
UnivNet-c16	3.72±0.08	3.11	4.00
MelGAN	3.56±0.10	2.87	4.34
Parallel WaveGAN	3.07±0.10	2.98	1.44
HiFiGAN V2	3.34±0.09	3.07	0.93
QGAN V1	3.61±0.05	3.43	3.71
QGAN V3	3.21±0.04	2.70	0.29

5.7 Loss Landscape Visualization

Loss landscape visualization is a technique used to represent the optimization surface that a neural network model navigates during training. It involves plotting the values of the loss function over the parameter space of the model. This visualization helps to understand the complexity of the optimization problem, the behavior of different optimization algorithms, and the reasons behind the model's convergence or non-convergence to a solution.

Visualizing the loss landscape can reveal insights into the nature of the loss function, such as the presence of multiple local minima, global minima, saddle points, and flat regions, which can have significant implications for the training process Goodfellow and Vinyals (2014). For example, a very rugged landscape with many sharp minima may suggest that the model could easily get stuck in a suboptimal point, whereas a smoother landscape with a clear path to the global minimum is generally more desirable.

In training neural networks, we optimize a loss function $L(\theta)$ that measures the prediction accuracy across a dataset of features $\{x_i\}$ and labels $\{y_i\}$. Given the high dimensionality of θ, the parameter space, and loss function visualization are constrained to lower dimensions, typically one or two.

For 1D visualization, we often employ linear interpolation between two parameter sets, θ and θ', evaluating the loss along the line segment connecting them. This can be parameterized by α, where $\theta(\alpha) = (1 - \alpha)\theta + \alpha\theta'$, and we visualize $f(\alpha) = L(\theta(\alpha))$. This method has been widely adopted to examine minima characteristics and the effect of hyperparameters like batch size on the loss surface.

However, 1D visualizations are limited and may not reveal non-convexities. It's been noted that loss functions can appear deceptively smooth along paths of optimization. Furthermore, factors such as batch normalization are not accounted for, which could affect the perceived sharpness of minima.

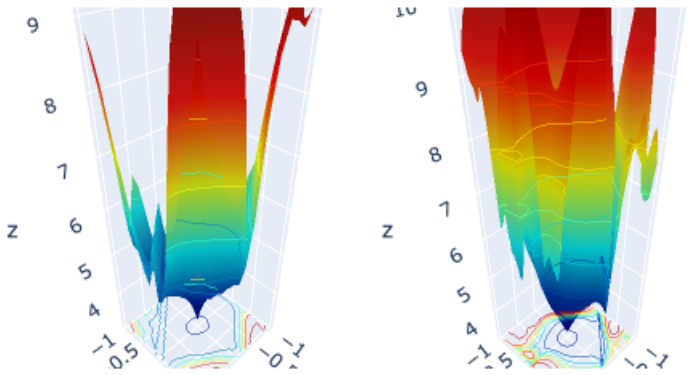
In 2D, contour plots and random direction techniques offer a broader view. From a central point θ^*, we consider two direction vectors, δ and η, to visualize the loss function as:

$$f(\alpha, \beta) = L(\theta^* + \alpha\delta + \beta\eta).$$

This provides insights into the optimization paths and minima diversity. However, due to computational limits, these plots are often of low resolution and cover limited regions, missing out on capturing the true complexity of loss landscapes.

In this thesis, we aim to present high-resolution visualizations across large weight space sections, to better understand how network architecture influences the loss function's non-convexity.

Figure 5.2 shows the loss landscapes for HiFIGAN V1 and QGAN V1 English vocoders, where only the generator parameters are modified. Due to space constraints, only visualization for the final trained model is plotted [3]. Consistent with the earlier studies, we observed that the sharpness of loss landscape minimum correlates well with the respective performance metrics. It can be observed that HiFiGAN has a wider basin but sharper minima. In contrast, QGAN has a wider valley near the minima, which shows potential towards better generalization for large-scale training Li *et al.* (2018) something we defer to future work. In practice, we observed the loss landscape of QGAN to be more stable than HiFiGAN in the initial phase of the training which shows the sensitivity of the sharper minima towards gradient perturbations.



(a) HiFiGAN (b) QGAN

Figure 5.2: Loss-landscape visualization of HiFiGAN and the proposed QGAN.

[3]Interative plots available online `https://anonymousvocoders.github.io/`

5.8 Conclusion

This study introduces QGAN, which informatively uses quaternion convolutions for efficient and high-quality speech synthesis. It significantly compresses model sizes without sacrificing audio fidelity, as demonstrated through extensive testing on English and Hindi datasets. It achieves better/comparable performance than the popular HiFiGAN and existing compressed neural vocoders across various metrics. The model's quaternion dual-discriminator strategy, multi-receptive field fusion, and tailored loss functions enable natural and intelligible speech synthesis. Loss landscape visualization confirms QGAN's training robustness and stability. QGAN's advancements in model efficiency, quality synthesis, and reproducibility mark a significant contribution to speech synthesis, especially for resource-constrained environments.

CHAPTER 6

CONCLUSION

6.1 Recapitulation of Thesis Objectives

The primary purpose of this thesis was to investigate the applicability and efficacy of quaternion neural networks (QNNs) in audio processing tasks, a domain traditionally dominated by real-valued neural networks. This research was motivated by the inherent advantages of quaternion algebra in handling multidimensional data, which is a characteristic of audio signals. Quaternion numbers, which include a real part and three imaginary parts, naturally encapsulate the structure of such data, potentially enabling more efficient computations and improved handling of the spatial and temporal correlations present in audio streams.

6.1.1 Summary of Objectives

The specific objectives set out at the beginning of this research were designed to thoroughly evaluate the potential of quaternion neural networks across various aspects of audio processing. These objectives were:

To Develop and Implement Quaternion Neural Network Architectures

The first objective was to design and implement effective quaternion-based models for specific audio processing tasks, namely keyword spotting, audio tagging, and speech synthesis. This involved adapting existing neural network models to utilize quaternion numbers and assessing their computational efficiency and performance in comparison to traditional models.

To Optimize Computational Efficiency

Another key objective was to enhance the computational efficiency of neural networks for audio processing. This involved integrating quaternion algebra to reduce the dimensionality and complexity of data processing, which is expected to decrease the computational load and enhance the feasibility of deploying these models on devices with limited computational resources.

To Evaluate Performance in Real-world Scenarios

Beyond theoretical and laboratory-based evaluations, it was crucial to assess how these quaternion-based models perform in real-world scenarios. This included testing the models' robustness and accuracy across different audio processing tasks and comparing them with state-of-the-art models. This also included testing the scalability of the Quaternion-based pipelines including Generative AI.

To Explore the Broader Applications of Quaternion Networks

Finally, the core aim of the research was to explore the broader applicability of quaternion neural networks in audio processing and signal processing. This involved theoretical explorations into the potential modifications and improvements over traditional models, and practical assessments in multiple case studies of data processing where multidimensionality is a factor.

6.2 Summary of Key Findings

The core of this thesis centered on applying quaternion algebra to enhance neural network architectures for audio processing, with a specific focus on three key areas: keyword spotting, audio tagging, and speech synthesis. Each area benefited from the unique capabilities of quaternion networks, leading to significant improvements in efficiency and effectiveness.

6.2.1 Keyword Spotting

The quaternion-based models developed for keyword spotting demonstrated a remarkable ability to maintain high accuracy while significantly reducing computational costs. These models were able to outperform the performance of state-of-the-art systems, but with a notably smaller computational footprint, making them ideal for on-device applications where resources are limited.

6.2.2 Audio Tagging

In the realm of audio tagging, the introduction of quaternion transformations coupled with model pruning techniques led to substantial reductions in both computational demands and memory usage. This approach did not only preserve but in some cases enhanced the accuracy of the tagging process, demonstrating that quaternion networks can effectively handle complex audio classification tasks without the extensive resource requirements typically associated with deep neural networks.

6.2.3 Speech Synthesis

For speech synthesis, quaternion-based vocoder models were explored. These models utilized quaternion algebra to decrease the number of parameters needed while still producing high-quality synthesized speech. The effectiveness of these models in generating clear and natural-sounding audio from fewer computational resources highlights their potential for embedding in low-power devices.

The findings from this research underscore the potential of quaternion neural networks to revolutionize the field of audio processing by providing a means to achieve high performance with lower computational and energy costs. The adaptability of these models to various audio-related tasks, coupled with their demonstrated efficiencies, positions them as a compelling alternative to traditional neural network architectures, especially in environments where resource constraints are a critical concern.

6.3 Evaluation of Research Hypotheses and Questions

6.3.1 Assessment of Hypotheses

The core hypotheses of this thesis centered around the assertion that quaternion neural networks could provide significant computational efficiencies and performance enhancements in audio processing tasks compared to traditional real-valued networks. These hypotheses were rigorously tested through a series of experimental studies focusing on three primary applications: keyword spotting, audio tagging, and speech synthesis.

Keyword Spotting

The findings for keyword spotting clearly supported the hypothesis. Quaternion neural networks demonstrated an ability to match the performance levels of state-of-the-art real-valued networks while consuming substantially fewer computational resources. This was evident in the reduced parameter count and the lower computational overhead, validating the hypothesis that quaternion networks can enhance computational efficiency without sacrificing accuracy. Here one notable thing was that the audio length was smaller and classes were only 35. One can ponder whether quaternions would be able to scale up in term of number of classes and audio length. We address these questions in the next two studies. This study also verifies that QNNs were able to demonstrate better learning behavior in deeper layers as compared to their Real counterparts. This also confirms the reason why they are so effective even with lesser parameters.

Audio Tagging

Similarly, in the audio tagging experiments, quaternion networks not only met but sometimes exceeded the performance benchmarks set by conventional networks. The integration of quaternion algebra with pruning techniques further underscored the efficiency gains, as these models maintained high accuracy while significantly cutting down on computational needs. This study is a proof of concept that architectural model compaction techniques can be further applied on QNNs (or along with structural compaction). This case study also validates that Quaternion Networks can scale up to a

large number of classes (527) and perform classification tasks with competitive accuracy. Also, this validates that Quaternion can also handle longer audio streams which could be a doubt from the first study.

Speech Synthesis

In the case of speech synthesis, quaternion-based vocoders showed that high-quality speech could be synthesized with fewer parameters than those required by traditional models. This result substantiates the hypothesis that quaternion networks can effectively reduce computational complexity in tasks involving complex audio signal processing. These models validated the QGAN's effectiveness and demonstrated that quaternions can be used for Generative AI. Also, the loss landscapes visualization gives preliminary evidence that quaternions can generalize better in large-scale training setups. We are confident that this happens due to the restriction on the degree of freedom posed by Hamilton's product. This study also completely evaporates any doubts regarding the scalability of the Quaternions.

6.4 Research Questions Addressed

The research questions posed at the outset of this thesis were aimed at exploring the potential of quaternion networks to revolutionize audio processing by leveraging their multidimensional data-handling capabilities. The questions focused on whether quaternion networks could match or exceed the performance of traditional models in specific audio applications, and whether they could do so with greater computational efficiency.

6.4.1 Performance Comparison

Across all applications tested, quaternion neural networks were able to provide competitive or superior performance relative to traditional real-valued networks. This finding answers the research question concerning the capability of quaternion networks to serve as viable alternatives to existing models, particularly in environments where resource constraints are a critical factor.

6.4.2 Computational Efficiency

The experiments consistently demonstrated that quaternion networks require fewer computational resources, evidenced by reduced memory usage and multiplication-accumulation operations. This effectively addresses the research question regarding the potential for computational efficiency gains with quaternion networks. The positive outcomes here suggest that these networks are not only theoretically capable but also practically advantageous for deployment in resource-limited settings.

6.4.3 Scaling Capabilities

All the covered case studies consistently validated that Quaternion Networks can scale up to a large number of classes and perform classification task with competitive accuracy. Also, this validates that Quaternion can also handle longer audio streams which could be a doubt from the first study. Also, the experiments involving wider and deeper networks have shown great performance, generalization, and learning behavior again giving proof that Quaternion-based Deep learning frameworks can scale well.

6.4.4 Experimental support to Theoretical Claims

Whether it was Multidimensional capabilities or restriction of the degree of freedom of the network, all these theoretical implications of Quaternion were validated through various small and large-scale experiments. Favorable results from log spectral norm and loss landscape visualization give enough experimental support to these theoretical claims. The learning behavior of the network is analyzed and it is validated that quaternions have a slight upper hand over the conventional network.

6.4.5 Broader Implications and Generalizability

The success of quaternion networks in the tested scenarios also provides a foundation for considering their application in other areas of machine learning and signal processing. With the three case studies chosen to tackle various challenges validate that Quaternions are a good fit for audio domain applications. This response to the broader

research question highlights the versatile potential of quaternion networks beyond the scope of audio processing.

In conclusion, the research findings comprehensively support the initial hypotheses and provide affirmative answers to the research questions posed. Quaternion neural networks have been proven to offer distinct advantages in terms of efficiency and performance for audio processing, setting the stage for their expanded use in various technological domains.

6.5 Limitations

Despite the promising results, there are several limitations that must be acknowledged. These limitations highlight areas for future research and potential improvement:

6.5.1 Integration with Existing Technology Stacks

One significant limitation is the integration of quaternion networks into existing technology stacks. Most current systems and architectures are optimized for traditional neural network models.

6.5.2 Drawback of Quaternion Transformation

From real numbers to complex numbers, Order property is lost (lowest upper bound). From complex to quaternions commutativity is lost. From quaternions to octonions associativity is lost. So these losses in property must somehow affect the topology of the resulting network which is yet to be investigated. Also, the use of Hamilton products sometimes increases the MACs as seen in study 2, which needs to be optimized. Hence, a faster method to do Hamilton Products must be developed and integrated into QNNs.

6.5.3 Generalization and Robustness

The generalization of quaternion models to real-world scenarios, where data can be noisy and highly irregular, is another area that requires further exploration.

6.6 Recommendations for Future Research

The promising results obtained from the application of quaternion neural networks in audio processing tasks within this thesis pave the way for numerous avenues of future research. These recommendations are aimed at both extending the application of QNNs into new domains and addressing the limitations identified in the current research.

6.6.1 Expansion into New Domains

Video Processing

Quaternion neural networks have shown significant promise in handling multidimensional audio data. Extending these applications to video processing could potentially revolutionize how video data is managed and analyzed.

Complex Sensor Data Analysis

Another exciting avenue for QNNs is in the analysis of data from complex sensor arrays, such as those used in IoT devices, automotive sensors, or even in industrial monitoring systems. These sensors often capture high-dimensional data that QNNs could process more efficiently, potentially leading to more accurate real-time analytics and decision-making.

Biomedical Applications

The application of QNNs in biomedical imaging, such as MRI or CT scans, where data is inherently three-dimensional, could provide new methodologies for enhancing the speed and accuracy of image analysis. This could be particularly impactful in real-time diagnostic procedures, where speed is critical. They could also be used to simulate and understand cognitive processes that involve spatial reasoning or the navigation of complex environments.

Integration in Current Techniques

Future work could explore middleware solutions that enable the seamless integration of quaternion operations within standard frameworks like Transformers, Large Language Models, Quantisation, or the development of new hardware optimized for quaternion processing.

Generalization and Robustness Testing

To ensure the practical deployment of QNNs, extensive testing on their generalization capabilities and robustness under different operational conditions is necessary. This includes exploring the models' performance in the presence of noise, their resilience to adversarial attacks, and their behavior under resource-constrained scenarios.

Theoretical Advances in Quaternion Learning Algorithms

Further theoretical research into the algorithms that govern learning in quaternion domains is essential. This includes the development of new optimization techniques specific to quaternions, and the exploration of loss functions, activation functions, and decomposition techniques that could enhance the training stability and performance of QNNs.

6.7 Concluding Thoughts

This thesis has successfully demonstrated the application of quaternion algebra within neural network architectures, focusing on improving computational efficiency in audio processing tasks without compromising performance. The development and evaluation of quaternion-based models for keyword spotting, audio tagging, and speech synthesis represent a significant advancement in the field. These models have proven capable of matching or exceeding the performance of traditional neural network architectures while requiring significantly fewer computational resources. The successful implementation across these three critical audio processing tasks underscores the versatility and robustness of quaternion neural networks, proving them to be more than just a theoreti-

cal enhancement but a practical solution to real-world challenges.

The broader impact of this research extends beyond the specific applications investigated here. By introducing quaternion neural networks as a viable option for audio processing, this work contributes to the ongoing evolution of neural network architectures. It offers a compelling alternative that addresses some of the most pressing limitations of current technology—particularly the need for more efficient computing power as data volumes and model complexities continue to grow.

In the context of audio processing, the quaternion approach provides a method to manage the high dimensionality and interrelatedness of audio data more naturally and effectively. This capability makes it especially suitable for applications in embedded systems and mobile devices, where computational resources are limited. Moreover, the principles explored here could be adapted for use in other domains that involve complex, multidimensional datasets, such as video processing, biomedical imaging, and multisensor data integration.

The implications for the field of neural networks are profound. Quaternion networks challenge the prevailing paradigms of network design and offer a pathway towards more sophisticated, resource-efficient models. This shift could lead to the broader adoption of advanced machine learning techniques in everyday technology, making intelligent systems more accessible and sustainable. Furthermore, the potential for reduced energy consumption across data centers implementing these models aligns with growing environmental sustainability goals within the tech industry.

In conclusion, the research presented in this thesis not only enhances our understanding of quaternion neural networks but also sets the stage for their wider application and further development. This work contributes to the foundational knowledge necessary to drive future innovations and paves the way for more efficient, powerful computational tools that are better equipped to handle the challenges of modern technology landscapes.

REFERENCES

1. **Abrol, V.** and **P. Sharma** (2020). Learning hierarchy aware embedding from raw audio for acoustic scene classification. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **28**, 1964–1973.

2. **Arık, S. o.**, **H. Jun**, and **G. Diamos** (2019). Fast spectrogram inversion using multi-head convolutional neural networks. *IEEE Signal Processing Letters*, **26**(1), 94–98.

3. **Berg, A.**, **M. O'Connor**, and **M. T. Cruz**, Keyword transformer: A self-attention model for keyword spotting. *In Interspeech 2021*. ISCA, 2021.

4. **Chaudhary, A.** and **V. Abrol**, Towards on-device keyword spotting using low-footprint quaternion neural models. *In 2023 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. 2023.

5. **Chen, N.** and **Y. Z. et al.**, WaveGrad: estimating gradients for waveform generation. *In International Conference on Learning Representations (ICLR)*. 2021.

6. **Cheng, Y.**, **D. Wang**, **P. Zhou**, and **T. Zhang** (2018). Model compression and acceleration for deep neural networks: The principles, progress, and challenges. *IEEE Signal Processing Magazine*, **35**(1), 126–136.

7. **Comminiello, D.**, **M. Lella**, **S. Scardapane**, and **A. Uncini**, Quaternion convolutional neural networks for detection and localization of 3d sound events. *In ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019.

8. **Cui, Y.**, **K. Takahashi**, and **M. Hashimoto**, Design of control systems using quaternion neural network and its application to inverse kinematics of robot manipulator. *In Proceedings of the 2013 IEEE/SICE International Symposium on System Integration*. IEEE, 2013.

9. **de Andrade, D. C.**, **S. Leo**, **M. L. D. S. Viana**, and **C. Bernkopf** (2018). A neural attention model for speech command recognition. *CoRR*, **abs/1808.08929**.

10. **Denil, M.**, **B. Shakibi**, **L. Dinh**, **M. Ranzato**, and **N. De Freitas** (2013). Predicting parameters in deep learning. *Advances in Neural Information Processing Systems*, 2148–2156.

11. **Dinh, L.**, **J. N. Sohl-Dickstein**, and **S. Bengio**, Density estimation using real NVP. *In International Conference on Learning Representations (ICLR)*. 2017.

12. **Donahue, C.**, **J. McAuley**, and **M. Puckette**, Adversarial audio synthesis. *In International Conference on Learning Representations (ICLR)*. 2018.

13. **et al., M. B.**, High fidelity speech synthesis with adversarial networks. *In International Conference on Learning Representations (ICLR)*. 2020.

14. **Farenick, D. R.** and **B. A. Pidkowich** (2003). The spectral theorem in quaternions. *Linear Algebra and its Applications*, **371**, 75–102. ISSN 0024-3795.

15. **Ford, L.**, **H. Tang**, **F. Grondin**, and **J. R. Glass** (2019). A deep residual network for large-scale acoustic scene analysis. *Interspeech*, 2568–2572.

16. **Frankle, J.** and **M. Carbin** (2019). The lottery ticket hypothesis: Finding sparse, trainable neural networks. *International Conference on Learning Representations*.

17. **Gaudet, C. J.** and **A. S. Maida**, Deep quaternion networks. *In 2018 International Joint Conference on Neural Networks (IJCNN)*. 2018*a*.

18. **Gaudet, C. J.** and **A. S. Maida**, Deep quaternion networks. *In 2018 International Joint Conference on Neural Networks (IJCNN)*. 2018*b*.

19. **Gemmeke, J. F.**, **D. P. Ellis**, **D. Freedman**, **A. Jansen**, **W. Lawrence**, **R. C. Moore**, **M. Plakal**, and **M. Ritter** (2017). AudioSet: An ontology and human-labeled dataset for audio events. *IEEE international Conference on Acoustics, Speech and Signal processing (ICASSP)*, 776–780.

20. **Gong, Y.**, **Y.-A. Chung**, and **J. Glass** (2021). PSLA: improving audio tagging with pretraining, sampling, labeling, and aggregation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **29**, 3292–3306.

21. **Gong, Y.**, **A. Rouditchenko**, **A. H. Liu**, **D. Harwath**, **L. Karlinsky**, **H. Kuehne**, and **J. Glass** (2022). Contrastive audio-visual masked autoencoder. *International Conference on Learning Representations*.

22. **Goodfellow, I.** and **et al.**, Generative adversarial nets. *In Advances in Neural Information Processing Systems*, volume 27. 2014.

23. **Goodfellow, I. J.** and **O. Vinyals** (2014). Qualitatively characterizing neural network optimization problems. *CoRR*, **abs/1412.6544**. URL `https://api.semanticscholar.org/CorpusID:16209268`.

24. **Gou, J.**, **B. Yu**, **S. J. Maybank**, and **D. Tao** (2021). Knowledge distillation: A survey. *International Journal of Computer Vision*, **129**(6), 1789–1819.

25. **Grassucci, E.**, **E. Cicero**, and **D. Comminiello**, *Quaternion Generative Adversarial Networks*. 2022, 57–86.

26. **Graves, A.**, **A.-r. Mohamed**, and **G. Hinton**, Speech recognition with deep recurrent neural networks. *In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. 2013.

27. **Gupta, D.** and **V. Abrol**, Time-frequency and geometric analysis of task-dependent learning in raw waveform based acoustic models. *In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2022.

28. **Han, S.**, **X. Liu**, **H. Mao**, **J. Pu**, **A. Pedram**, **M. A. Horowitz**, and **W. J. Dally** (2016). EIE: Efficient inference engine on compressed deep neural network. *ACM SIGARCH Computer Architecture News*, **44**(3), 243–254.

29. **He, K.**, **X. Zhang**, **S. Ren**, and **J. Sun**, Deep residual learning for image recognition. *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

30. **Howard, A.**, **M. Sandler**, **G. Chu**, **L.-C. Chen**, **B. Chen**, **M. Tan**, **W. Wang**, **Y. Zhu**, **R. Pang**, **V. Vasudevan**, *et al.* (2019). Searching for MobileNetV3. *Proceedings of the IEEE/CVF international Conference on Computer Vision*, 1314–1324.

31. **Huang, R.**, **M. W. Y. Lam**, **J. Wang**, **D. Su**, **D. Yu**, **Y. Ren**, and **Z. Zhao**, FastDiff: a fast conditional diffusion model for high-quality speech synthesis. *In International Joint Conference on Artificial Intelligence (IJCAI)*. 2022.

32. **Huang, X.** and **S. Gai** (2020). Banknote classification based on convolutional neural network in quaternion wavelet domain. *IEEE Access*, **8**, 162141–162148.

33. **Jang, W.**, **D. Lim**, **J. Yoon**, **B. Kim**, and **J. Kim**, UnivNet: A Neural Vocoder with Multi-Resolution Spectrogram Discriminators for High-Fidelity Waveform Generation. *In Interspeech*. 2021.

34. **Kalchbrenner, N.** and **et al.**, Efficient neural audio synthesis. *In International Conference on Machine Learning (ICML)*, volume 80. PMLR, 2018.

35. **Kilgour, K.**, **M. Zuluaga**, **D. Roblek**, and **M. Sharifi**, Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms. *In Interspeech*. 2019.

36. **Kim, B.**, **S. Chang**, **J. Lee**, and **D. Sung**, Broadcasted residual learning for efficient keyword spotting. *In Interspeech*. 2021*a*.

37. **Kim, B.**, **S. Chang**, **J. Lee**, and **D. Sung**, Broadcasted Residual Learning for Efficient Keyword Spotting. *In Interspeech*. 2021*b*.

38. **Kingma, D. P.** and **P. Dhariwal**, Glow: Generative flow with invertible 1x1 convolutions. *In Advances in Neural Information Processing Systems*, volume 31. 2018.

39. **Kong, J.**, **J. Kim**, and **J. Bae**, HiFi-GAN: generative adversarial networks for efficient and high fidelity speech synthesis. *In Advances in Neural Information Processing Systems*, volume 33. 2020*a*.

40. **Kong, Q.**, **Y. Cao**, **T. Iqbal**, **Y. Wang**, **W. Wang**, and **M. D. Plumbley** (2020*b*). PANNs: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **28**, 2880–2894.

41. **Kong, Z.**, **W. Ping**, **J. Huang**, **K. Zhao**, and **B. Catanzaro**, DiffWave: a versatile diffusion model for audio synthesis. *In International Conference on Learning Representations (ICLR)*. 2021.

42. **Kumar, K.** and **et al.**, MelGAN: generative adversarial networks for conditional waveform synthesis. *In Advances in Neural Information Processing Systems*, volume 32. 2019.

43. **Kumar, S.**, **A. Chaudhary**, and **R. K. Singh** (2020). On the learning machine with amplificatory neuron in complex domain. *Arabian Journal for Science and Engineering*, **45**(12), 10287–10309. ISSN 2191-4281.

44. **Kumar, S.**, **R. K. Singh**, and **A. Chaudhary** (2023). A novel non-linear neuron model based on multiplicative aggregation in quaternionic domain. *Complex & Intelligent Systems*, **9**(3), 3161–3183. ISSN 2198-6053.

45. **Kusamichi, H.**, **T. Isokawa**, **N. Matsui**, **Y. Ogawa**, and **K. Maeda**, A new scheme for color night vision by quaternion neural network. *In Proceedings of the 2nd international conference on autonomous robots and agents*, volume 1315. Citeseer, 2004.

46. **Lagunas, F.**, **E. Charlaix**, **V. Sanh**, and **A. M. Rush** (2021). Block pruning for faster transformers. *arXiv preprint arXiv:2109.04838*.

47. **Larsen, L. B.** and **et al.**, Autoencoding beyond pixels using a learned similarity metric. *In International Conference on Machine Learning (ICML)*, volume 48. PMLR, 2016.

48. **Li, H.**, **A. Kadav**, **I. Durdanovic**, **H. Samet**, and **H. P. Graf** (2017). Pruning filters for efficient ConvNets. *International Conference on Learning Representations*.

49. **Li, H.**, **Z. Xu**, **G. Taylor**, **C. Studer**, and **T. Goldstein**, Visualizing the loss landscape of neural nets. *In Advances in Neural Information Processing Systems*, volume 31. 2018.

50. **Liang, T.**, **J. Glossner**, **L. Wang**, **S. Shi**, and **X. Zhang** (2021). Pruning and quantization for deep neural network acceleration: A survey. *Neurocomputing*, **461**, 370–403.

51. **Lin, M.**, **R. Ji**, **Y. Wang**, **Y. Zhang**, **B. Zhang**, **Y. Tian**, and **L. Shao** (2020). HRank: Filter pruning using high-rank feature map. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1529–1538.

52. **Lin, S.**, **R. Ji**, **C. Yan**, **B. Zhang**, **L. Cao**, **Q. Ye**, **F. Huang**, and **D. Doermann** (2019). Towards optimal structured CNN pruning via generative adversarial learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2790–2799.

53. **Luo, J.-H.** and **J. Wu** (2020). AutoPruner: An end-to-end trainable filter pruning method for efficient deep model inference. *Pattern Recognition*, **107**, 107461.

54. **Luo, J.-H.**, **H. Zhang**, **H.-Y. Zhou**, **C.-W. Xie**, **J. Wu**, and **W. Lin** (2018). ThiNet: pruning CNN filters for a thinner net. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **41**(10), 2525–2538.

55. **Luo, L.**, **H. Feng**, and **L. Ding**, Color image compression based on quaternion neural network principal component analysis. *In International Conference on Multimedia Technology (ICMULT)*. 2010.

56. **Majumdar, S.** and **B. Ginsburg**, MatchboxNet: 1D Time-Channel Separable Convolutional Neural Network Architecture for Speech Commands Recognition. *In Interspeech*. 2020.

57. **Mao, X.**, **Q. Li**, **H. Xie**, **R. K. Lau**, **Z. Wang**, and **S. Smolley**, Least squares generative adversarial networks. *In IEEE International Conference on Computer Vision (ICCV)*. IEEE Computer Society, 2017. ISSN 2380-7504.

58. **Martin, C. H.**, **T. S. Peng**, and **M. W. Mahoney** (2021). Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data. *Nature Communications*, **12**(4122).

59. **McCulloch, W. S.** and **W. Pitts** (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, **5**(4), 115–133. ISSN 1522-9602. URL `https://doi.org/10.1007/BF02478259`.

60. **Mordido, G.**, **M. V. Keirsbilck**, and **A. Keller**, Compressing 1d time-channel separable convolutions using sparse random ternary matrices. *In Interspeech*. 2021.

61. **Muckenhirn, H.**, **V. Abrol**, **M. Magimai-Doss**, and **S. Marcel**, Understanding and Visualizing Raw Waveform-Based CNNs. *In Proc. Interspeech*. 2019.

62. **Muppidi, A.** and **M. Radfar**, Speech emotion recognition using quaternion convolutional neural networks. *In ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021.

63. **Nguyen, T. D.**, **D. Phung**, *et al.*, Quaternion graph neural networks. *In Asian conference on machine learning*. PMLR, 2021.

64. **Parcollet, T.**, **M. Morchid**, **P.-M. Bousquet**, **R. Dufour**, **G. Linarès**, and **R. De Mori**, Quaternion neural networks for spoken language understanding. *In IEEE Spoken Language Technology Workshop (SLT)*. 2016.

65. **Parcollet, T.**, **M. Morchid**, and **G. Linarès**, Deep quaternion neural networks for spoken language understanding. *In IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. 2017.

66. **Parcollet, T.**, **M. Morchid**, and **G. Linarès**, Quaternion convolutional neural networks for heterogeneous image processing. *In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2019. ISSN 2379-190X.

67. **Parcollet, T.**, **M. Ravanelli**, **M. Morchid**, **G. Linarès**, **C. Trabelsi**, **R. De Mori**, and **Y. Bengio** (2018*a*). Quaternion recurrent neural networks. *arXiv preprint arXiv:1806.04418*.

68. **Parcollet, T.**, **Y. Zhang**, **M. Morchid**, **C. Trabelsi**, **G. Linarès**, **R. De Mori**, and **Y. Bengio** (2018*b*). Quaternion convolutional neural networks for end-to-end automatic speech recognition. *arXiv preprint arXiv:1806.07789*.

69. **Peng, K.**, **W. Ping**, **Z. Song**, and **K. Zhao**, Non-autoregressive neural text-to-speech. *In International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*. PMLR, 2020.

70. **Ping, W.**, **K. Peng**, and **J. Chen**, Clarinet: Parallel wave generation in end-to-end text-to-speech. *In 7th International Conference on Learning Representations, ICLR*. 2019.

71. **Ping, W.**, **K. Peng**, **K. Zhao**, and **Z. Song**, WaveFlow: A compact flow-based model for raw audio. *In International Conference on Machine Learning (ICML)*. 2020.

72. **Prenger, R.**, **R. Valle**, and **B. Catanzaro**, Waveglow: a flow-based generative network for speech synthesis. *In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2019.

73. **Qiu, X.**, **T. Parcollet**, **M. Ravanelli**, **N. D. Lane**, and **M. Morchid**, Quaternion neural networks for multi-channel distant speech recognition. *In Interspeech*. 2020.

74. **Rix, A.**, **J. Beerends**, **M. Hollier**, and **A. Hekstra**, Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. *In IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings ICASSP*. 2001.

75. **Rokh, B.**, **A. Azarpeyvand**, and **A. Khanteymoori** (2023). A comprehensive survey on model quantization for deep neural networks in image classification. *ACM Trans. Intell. Syst. Technol.*, **14**(6).

76. **Rybakov, O.**, **N. Kononenko**, **N. Subrahmanya**, **M. Visontai**, and **S. Laurenzo**, Streaming keyword spotting on mobile devices. *In Interspeech*. 2020.

77. **Sainath, T. N.** and **et al.**, Low-rank matrix factorization for deep neural network training with high-dimensional output targets. *In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2013.

78. **Sainath, T. N.** and **C. Parada**, Convolutional neural networks for small-footprint keyword spotting. *In Interspeech*. 2015.

79. **Salimans, T.** and **D. P. Kingma**, Weight normalization: a simple reparameterization to accelerate training of deep neural networks. *In International Conference on Neural Information Processing Systems*. 2016.

80. **Schmid, F.**, **K. Koutini**, and **G. Widmer** (2023). Efficient large-scale audio tagging via transformer-to-CNN knowledge distillation. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

81. **Shang, F.** and **A. Hirose** (2013). Quaternion neural-network-based polsar land classification in poincare-sphere-parameter space. *IEEE Transactions on Geoscience and Remote Sensing*, **52**(9), 5693–5703.

82. **Singh, A.**, **H. Liu**, and **M. D. Plumbley** (2023). E-PANNs: sound recognition using efficient pre-trained audio neural networks. *Internoise 2023, Chiba, Greater Tokyo, Japan,*.

83. **Su, W.**, **L. Li**, **F. Liu**, **M. He**, and **X. Liang** (2022). AI on the Edge: a comprehensive review. *Artificial Intelligence Review*, **55**(8), 6125–6183.

84. **Taal, C. H.**, **R. C. Hendriks**, **R. Heusdens**, and **J. Jensen**, A short-time objective intelligibility measure for time-frequency weighted noisy speech. *In IEEE International Conference on Acoustics, Speech and Signal Processing*. 2010.

85. **Tan, M.** and **Q. Le** (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning*, 6105–6114.

86. **Tay, Y.**, **A. Zhang**, **A. T. Luu**, **J. Rao**, **S. Zhang**, **S. Wang**, **J. Fu**, and **S. C. Hui**, Lightweight and efficient neural natural language processing with quaternion networks. *In Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 2019*a*.

87. **Tay, Y.**, **A. Zhang**, **L. A. Tuan**, **J. Rao**, **S. Zhang**, **S. Wang**, **J. Fu**, and **S. C. Hui** (2019*b*). Lightweight and efficient neural natural language processing with quaternion networks. *arXiv preprint arXiv:1906.04393*.

88. **van den Oord Aaron** and **et al.**, Parallel WaveNet: Fast high-fidelity speech synthesis. *In Proceedings of the 35th International Conference on Machine Learning*, volume 80. PMLR, 2018.

89. **van den Oord et al., A.**, WaveNet: a generative model for raw audio. *In Arxiv*. 2016.

90. **Virtanen, T.**, **M. D. Plumbley**, and **D. Ellis**, *Computational Analysis of Sound Scenes and Events*. Springer, 2018.

91. **Warden, P.** (2018). Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition. *ArXiv e-prints*. URL `https://arxiv.org/abs/1804.03209`.

92. **Wei, Y.**, **Z. Gong**, **S. Yang**, **K. Ye**, and **Y. Wen** (2022). Edgecrnn: an edge-computing oriented model of acoustic feature enhancement for keyword spotting. *Journal of Ambient Intelligence and Humanized Computing*, **13**, 1–11.

93. **Wen, W.**, **C. Wu**, **Y. Wang**, **Y. Chen**, and **H. Li** (2016). Learning structured sparsity in deep neural networks. *Advances in Neural Information Processing Systems*, 2074–2082.

94. **Xia, M.**, **Z. Zhong**, and **D. Chen** (2022). Structured pruning learns compact and accurate models. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1513–1528.

95. **Yamamoto, R.**, **E. Song**, and **J.-M. Kim**, Parallel WaveGAN: a fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. *In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020.

96. **Yang, J.**, **J. Lee**, **Y. Kim**, **H.-Y. Cho**, and **I. Kim**, VocGAN: A High-Fidelity Real-Time Vocoder with a Hierarchically-Nested Adversarial Network. *In Interspeech*. 2020.

97. **Yeom, S.-K.**, **K.-H. Shim**, and **J.-H. Hwang** (2021). Toward compact deep neural networks via energy-aware pruning. *arXiv preprint arXiv:2103.10858*.

98. **Yin, Q.**, **J. Wang**, **X. Luo**, **J. Zhai**, **S. K. Jha**, and **Y.-Q. Shi** (2019). Quaternion convolutional neural network for color image classification and forensics. *IEEE Access*, **7**, 20293–20301.

99. **Zhang, S.**, **Z. Shuang**, **Q. Shi**, and **Y. Qin**, Improved mandarin keyword spotting using confusion garbage model. *In International Conference on Pattern Recognition (ICPR)*. 2010.

100. **Zhou, Y.**, **L. Jin**, **G. Ma**, and **X. Xu** (2021). Quaternion capsule neural network with region attention for facial expression recognition in color images. *IEEE Transactions on Emerging Topics in Computational Intelligence*, **6**(4), 893–912.

101. **Zhu, X.**, **Y. Xu**, **H. Xu**, and **C. Chen** (2018). Quaternion convolutional neural networks. *Proceedings of the European Conference on Computer Vision (ECCV)*, 631–647.

LIST OF PAPERS BASED ON THESIS

1. Aryan Chaudhary, Dr. Vinayak Abrol Towards on-device keyword spotting using low-footprint Quaternion neural models *Proceedings IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, (2023). [Codebase]

2. Aryan Chaudhary, Dr. Arshdeep Singh, Dr. Vinayak Abrol, Prof Mark D. Plumbley Efficient CNNs with Quaternion Transformations and Pruning for Audio Tagging *INTERSPEECH 2024*, Volume-NA, Page-NA, (2024).[Codebase]

3. Aryan Chaudhary, Dr. Vinayak Abrol QGAN: Low Footprint Quaternion Neural Vocoder for Speech Synthesis *INTERSPEECH 2024*, Volume-NA, Page-NA, (2024). [Codebase]