ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

NOVEL FRACTIONAL ORDER CALCULUS-BASED AUDIO PROCESSING METHODS AND THEIR APPLICATIONS ON NEURAL NETWORKS FOR CLASSIFICATION AND SYNTHESIS PROBLEMS

Ph.D. THESIS

Bilgi Görkem YAZGAÇ

Department of Electronics and Communication Engineering

Electronics Engineering Programme

OCTOBER 2023

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

NOVEL FRACTIONAL ORDER CALCULUS-BASED AUDIO PROCESSING METHODS AND THEIR APPLICATIONS ON NEURAL NETWORKS FOR CLASSIFICATION AND SYNTHESIS PROBLEMS

Ph.D. THESIS

Bilgi Görkem YAZGAÇ
(504162208)

Department of Electronics and Communication Engineering

Electronics Engineering Programme

Thesis Advisor: Assoc. Prof. Dr. Mürvet KIRCI

OCTOBER 2023

ISTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

KESİRLİ MERTEBEDEN KALKÜLÜS TEMELLİ YENİ SES İŞLEME YÖNTEMLERİ VE BUNLARIN SİNİR AĞLARI ÜZERİNDE SINIFLANDIRMA VE SENTEZ PROBLEMLERİNE UYGULANMASI

DOKTORA TEZİ

Bilgi Görkem YAZGAÇ
(504162208)

Elektonik ve Haberleşme Mühendisliği Anabilim Dalı

Elektronik Mühendisliği Programı

Tez Danışmanı: Doç. Dr. Mürvet KIRCI

EKİM 2023

Bilgi Görkem YAZGAÇ, a Ph.D. student of İTU Graduate School student ID 504162208, successfully defended the dissertation entitled "NOVEL FRACTIONAL ORDER CALCULUS-BASED AUDIO PROCESSING METHODS AND THEIR APPLICATIONS ON NEURAL NETWORKS FOR CLASSIFICATION AND SYNTHESIS PROBLEMS", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Assoc. Prof. Dr. Mürvet KIRCI** ....
Istanbul Technical University

Jury Members : **Prof. Dr. İsmail Serdar ÖZOĞUZ** ....
Istanbul Technical University

Assoc. Prof. Dr. Bülent BOLAT ....
Yıldız Technical University

Prof. Dr. Ece Olcay GÜNEŞ ....
Istanbul Technical University

Prof. Dr. Burcu ERKMEN ....
Yıldız Technical University

Date of Submission : 5 September 2023
Date of Defense : 24 October 2023

To Hazal,

FOREWORD

Acquiring a Ph.D. has been a challenging process. Through all this time many people provided their help and support.

First and foremost, I would like to thank my lovely wife Hazal for being there, whenever I needed.

I would like to thank all members of my family. Their support provided me the means to keep continuing on this very hard task.

Assoc. Prof. Dr. Mürvet KIRCI has been my advisor since my graduate student days. As the supervisor of this thesis work, and as being a great advisor, I would like to express a special appreciation to her.

My thesis committee has provided wonderful and constructive criticism. For this, I would like to express my gratitude to Prof. Dr. İbrahim Serdar ÖZOĞUZ and Assoc. Prof. Dr. Bülent BOLAT.

I feel myself lucky for being a member of Istanbul Technical University (ITU) SMART Laboratory and sharing the place with my colleagues, Sercan AYGÜN, Ph. D. and Halil DURMUŞ, M. Sc. A special gratitude needs to be given to Prof. Dr. Ece Olcay GÜNEŞ. Working environment in our laboratory has been an important supporting point for my work.

I would like to thank all members of Istanbul Technical Univerity Electronics and Communication Departmant.

Last but not least, all my friends have been there during the ups and downs of my Ph. D. journey. For their support and joy, I will be forever in dept.

The financial support for this thesis was provided by ITU Scientific Research Project Unit under Grant MDK-2020-42479.

September 2023

Bilgi Görkem YAZGAÇ
(Electronics Engineering, M.Sc)

TABLE OF CONTENTS

Page

ABBREVIATIONS

ADMM : Alternating Direction Method of Multipliers
AGC : Amplitude-Based-Gated-Complex-Convolution
AI-GC : Amplitude-Informed Gated Complex Convolution
AR : Auto Regressive
ARFIMA : Autoregressive Fractionally Integrated Moving Average
ARIMA : Auto Regressive Integrated Moving Average
ARMA : Auto Regressive Moving Average
ASR : Automatic Speech Recognition
CNN : Convolutional Neural Networks
DeGLI : Deep Griffin-Lim Iteration
DNN : Deep Neural Networks
DSP : Digital Signal Processing
ESC : Environmental Sound Classification
FDE : Fractional Order Differential Equations
FFN : Feed Forward Networks
FFT : Fast Fourier Transform
FGLA : Fast Griffin-Lim Algorithm
FLP : Fractional Linear Prediction
FOC : Fractional Order Calculus
GAN : Generative Adversarial Networks
GL : Grünwald-Letnikov
GLA : Griffin–Lim Algorithm
GLA-ADMM: Griffin-Lim ADMM Algorithm
GMM : Gaussian Mixture Models
HMM : Hidden Markow Models
ISTFT : Inverse Short Term Fourier Transfom
ITU-T : International Telecommunication Union Telecommunication Standardization Sector
kNN : k-Nearest Neighbors
LAS : Listen, Attend and Spell
LP : Linear Prediction

LPC	: Linear Prediction Coefficients
LPCC	: Linear Predictive Cepstral Coefficients
LSF	: Line Spectral Frequencies
LSTM	: Long-Short Term Memory Models
MA	: Moving Average
MFCC	: Mel Frequency Cepstral Coefficients
MIR	: Music Information Retrieval
MOS	: Mean Opinion Dcore
MUSHRA	: Multiple Stimuli with Hidden Reference and Anchor
NAS	: Neural Audio Synthesis
PESQ	: Perceptual Evaluation of Speech Quality
PnP	: Plug-and-Play
PSDF	: Power Spectral Density Function
ReLu	: Rectified Linear Unit
RL	: Riemann-Liouville
RNN	: Recurrent Neural Networks
SC	: Spectral Covergence
SPSI	: Single-Pass Spectrogram inversion
SSNR	: Spectral Signal to Noise Ratio
STFT	: Short Term Fourier Transfom
STFTM	: Short-Time Fourier Transform Magnitude
SVM	: Support Vector Machines
TIMIT	: Texas Instruments / Massachusetts Institute of Technology
TTS	: Text-to-Speech
VTLN	: Vocal Tract Length Normalization
VTLP	: Vocal Tract Length Perturbation

SYMBOLS

${}_{a}D_{t}^{\propto}$	: Fractional differintegral operator with order $\propto$ on a function defined in $[\mathrm{a}, \mathrm{t}]$
$f(\mathrm{t})$	: A continous function
t	: time index
$\propto, \beta$	: Fractional differintegration orders
s	: Laplace domain symbol
$F(s)$	: Function on laplace domain
$\mathcal{L}[\,]$	: Laplace operator
$f'(\mathrm{t}), f^{(n)}(\mathrm{t})$	: Integer order derivatives
h	: Step size
$\binom{\alpha}{\mathrm{j}}$	: Combination operation
!	: Factorial operator
Γ (n)	: Gamma function
θ_{q}	: q^{th} Moving average coefficient
φ_{p}	: p^{th} Auto regressive coefficient
ε_n	: Gaussian noise
L	: Lag operator
d	: Fractional Integration value of a ARFIMA(p, d, q) time series model
$s[\mathrm{n}]$	: Discrete Time Signal
n	: Discrete Time index
Z^{+}	: Positive integers
$x_{\mathrm{n}}, y_{\mathrm{n}}$	: n^{th} values of time series x and y.
$\boldsymbol{p}_{\mathrm{k}}$	: A vector of the values of fractionally integrated discrete time signal
Ω	: A matrix that contains the values k different fractional integrations of a discrete time signal
$\boldsymbol{a}$	: A vector of linear prediction coefficients
a_{k}	: k^{th} linear prediction coefficient
$\boldsymbol{s}$	: An array of values of $s[\mathrm{n}]$
$\boldsymbol{s}'$	: An array of values of $s'[\mathrm{n}]$

dB	: Decibel
Hz	: Hertz
Mel	: Unit of mel frequency scale
f_{Linear}	: Linear frequency scale value
f_{Mel}	: Mel frequency scale value
$f_{H(Mel)}$	: High frequency boundry of mel scale
$f_{L(Mel)}$	: Low frequency boundry of mel scale
$f_{cm(Mel)}$	: Central mel frequency value of m^{th} triangular filter
$H_m[\,]$	: m^{th} triangular filter
$S[\,]$	: Spectrum of a signal
X_m	: Output of m^{th} triangular filter
$X_{m,n}$	: Output of m^{th} triangular filter and n^{th} time index
α_{VTLP}	: VTLP warping coefficient
$f_{H(VTLP)}$	: High frequency boundry for VTLP
f_S	: Sampling frequency
f_{VTLP}	: Frequency scale value after VTLP
b_k	: k^{th} coefficient for Grünwald-Letnikow fractional order differintegral
$(\alpha)_k$	: Porchammer symbol
$[D^{\alpha} f(x)]_{GL}$	: Grünwald-Letnikow fractional order differintegral of function $f(x)$
$\left[D^{\alpha} f(x_j)\right]_{RL}$	: Riemann-Liouville fractional order differintegral of function $f(x)$ at the point x_j
$A_{k,j}$	: k^{th} coefficient for Riemann-Liouville fractional order differintegral
$\mathbf{R}$	: A matrix that contains $A_{k,j}$ values
$\mathbf{f}$	: An array that contains the values of function $f(x)$
$\mathbf{f}_{Linear}{}'$	: Linear frequency scale after Fractional Order Frequency Scale operation
$\mathbf{f}_{Mel}{}'$	: Mel frequency scale after Fractional Order Mel Scale operation
$F(\mathbf{X}\vert\mathbf{W})$	: Neural network representation of more than one layer with input tensor $\mathbf{X}$ and weight tensor $\mathbf{W}$
$f_k(\mathbf{X}\vert\mathbf{W}_k)$	: Representation of a neural network layer k with weight tensor $\mathbf{W}_k$
$*$	: Convolution operator
$\mathbf{B}$	: Bias tensor
$max(0, x)$	: Rectified Linear Unit activation function
R^L	: L sized real number space

$\|\| \|\|_{Fro}^2$	: Frobenius norm
G	: Gabor based transform
x	: An array of real numbers
S	: A matrix of non-negative real coefficients
C^{MxN}	: MxN sized complex number space
C	: A matrix of complex number coefficients
C_1, C_2	: sets that are defined in $C^{\mathrm{M\ N}}$ complex number space
$G^{\dagger}$	: Pseudo inverse of Gabor based transform
P_{C_1}, P_{C_2}	: Projections on the sets C_1 and C_2
X, Y	: A matrix
p	: Constant parameter for FGLA
t_{C_1}, t_{C_2}	: consistency constraint, that satisfy the solution on the sets C_1 and C_2
$prox_{\frac{f}{\rho}}()$	: Proximal operator on a function f with step size ρ
ρ	: Related to step size of the mapped values of proximal operation
$\mathbf{T}_{\mathrm{n}}$, $\mathbf{D}_{\mathrm{n}}$, $\mathbf{K}_{\mathrm{n}}$	: Auxiliary variables for faster convergence in FGLA and GLA-ADMM
$\mathrm{d}^2{}_{C_1}$	: Frobenius norm on set C_1
c	: A constant
C	: A constant, $ln(c)$
j	: $\sqrt{-1}$
ω	: Angular frequency
$\hat{P}_{\mathrm{k}}$	: Power spectral density function, $P(\omega_{\mathrm{k}})$, estimate of k^{th} frequency bin
$\|F_{\mathrm{i}}(\omega)\|^2$	: Power spectral density function of i^{th} band pass filter
$\hat{\omega}_{\mathrm{j}}$	: Instantaneous angular frequency estimate of spectral peak with respect to the frequency bin index j
$\varphi_{\mathrm{m,j}}$	: Phase value for frequency bin index j and time frame m
$Conv_C$	: Complex convolutional layer
$\mathbf{W}_R$	: Real number weight tensor
$\mathbf{W}_C$	: Complex number weight tensor
$\mathbf{W}_{\mathrm{Re}}$	: Real number part of weight tensor
$\mathbf{W}_{\mathrm{Im}}$	: Imaginer number part of weight tensor
$\mathbf{C}_{\mathrm{Re}}$	: Real part of complex coefficients of input tensor
$\mathbf{C}_{\mathrm{Im}}$	: Imaginer part of complex coefficients of input tensor

$\mathbf{E}$	: A Gaussian Noise Tensor
$\odot$	: Piecewise multiplication
$AmpGate$	: Amplitude based gating mechanism
AGC	: Amplitude based gated complex convolution

LIST OF TABLES

Page

LIST OF FIGURES

Page

NOVEL FRACTIONAL ORDER CALCULUS-BASED AUDIO PROCESSING METHODS AND THEIR APPLICATIONS ON NEURAL NETWORKS FOR CLASSIFICATION AND SYNTHESIS PROBLEMS

SUMMARY

This thesis dissertation aims to explore the application of the Fractional Order Calculus (FOC) framework in addressing contemporary problems in audio signal processing. One crucial aspect of present audio signal processing approaches is their reliance on large amounts of data, which necessitates appropriate tools for increasing amount of data. Another important aspect is in relation to the methods used to produce inference models. The neural network approaches dominating the field often require optimization of a large number of parameters. As a result, digital signal processing (DSP) tools are being repurposed to reduce the parameters of neural models. The introductory chapter provides an overview of the dissertation's purpose, which is to investigate whether FOC can provide novel methods to solve problems in neural network-based audio signal classification and reconstruction.

Chapter 2 introduces the FOC framework by explaining its capabilities and complexities. While the complexities of FOC have often caused it to be overlooked in engineering applications, its capabilities have attracted the interest of many researchers in various fields, including audio processing, time series estimation, and image enhancement. Providing examples of FOC based applications on audio signal processing, this chapter aims to provide fammiliarity to the FOC concept.

The dissertation is structured such that each chapter focuses on a specific application of audio signal processing.

Chapter 3 tackles the problem of audio classification, which is categorised by being speech, music or environmental sound signals. Due to the limited availability of data for environmental sound signals, data augmentation methods remain crucial for Environmental Sound Classifaciton (ESC) problems. The chapter presents three FOC based data augmentation methods: Fractional Order Mask, Fractional Order Frequency Scale, and Fractional Order Mel Scale. Fractional Order Mask and Fractional Order Mel Scale methods are applied to Mel Spectrogram and Log-Mel Spectrogram representations of envrionmental sound data. Experiments on ESC problem with neural architectures demonstrate their effectiveness as data augmentation tools in improving the accuracy of neural network models. The findings indicate that employing a data augmentation procedure in combination with the proposed methods can yield a boost of approximately 7.7% in performance for a 5-layer CNN when Log-Mel Spectrograms are used as input. Similarly, the augmented dataset resulted in a increase of over 9% in performance for a 18-layer ResNet.

Chapter 4 delves into audio synthesis and its importance in reconstructing time domain representations of audio signals. The history of vocoding methods and their relation to signal reconstruction approaches are discussed. The chapter focuses on phase reconstruction with methods such as SPSI and spectral consistency based iterative

methods such as the Griffin-Lim Algorithm (GLA) and its novel forms. In this chapter a FOC based method is proposed. The FOC based method models a signal's Power Spectral Density Function (PSDF) using Fractional Differential Equations (FDE), estimating the instantaneous frequency of a peak in a windowed audio spectrum. This method proves effective in phase reconstruction. The results show the usage of FOC framework provided up to 4% better quality than SPSI. The experiments also highlight the proposed method's effectiveness as an initial phase estimator for spectral consistency based iterative methods.

Chapter 5 explores the contemporary research topic of Neural Audio Synthesis (NAS). The chapter discusses advancements in deep neural networks (DNNs) and the need for smaller-sized models. In the literature, the Deep Griffin-Lim Iteration (DeGLI) architecture combines the ideas of spectral concistency based iterative algorithms for speech reconstruction with neural networks and allows for training a small sized submodule as a denoiser. This chapter of dissertation combines the outcomes of chapters 3 and 4 to evaluate the applicability of FOC based methods to NAS. Specifically, a novel augmentation proceducure that combines Fractional Order Frequency Scale and FDE based phase estimation method produces spectrally consistent spectrograms. Experimental results show the effectiveness of the proposed FOC based data augmentation and phase estimation methods in improving audio reconstruction quality.

In conclusion, the results of this thesis can be categorized into three groups. First, it has been shown that the FOC mathematical framework can contribute to the design of new data augmentation methods to address the problem of high number of data requirements, which is a major problem of most data driven approaches of today. Secondly, it is shown that FDE based approaches can produce successful solutions for phase component estimation of a speech signal. In addition, this approach has also shown success in combination with data driven audio synthesis methods. A third result is the demonstration that a data augmentation strategy that produces spectrally consistent spectrograms can provide higher performance for neural network based audio reconstruction models. These three results emphasize that Fractional Order Calculus based methods can have positive results in addressing the challenges of contemporary audio classification and synthesis work.

KESİRLİ MERTEBEDEN KALKÜLÜS TEMELLİ YENİ SES İŞLEME YÖNTEMLERİ VE BUNLARIN SİNİR AĞLARI ÜZERİNDE SINIFLANDIRMA VE SENTEZ PROBLEMLERİNE UYGULANMASI

ÖZET

Alışılmış yaklaşımlara dayalı yöntemlerin hakim olduğu bir bilimsel araştırma alanındaki problemlerin çözümünde alışılmadık bir yaklaşımın ya da daha az uygulanan bir matematiksel çerçevenin araçlarının kullanılmasının yeni çözümler için bir potansiyel yaratması ve umut verici olanaklar ortaya koyması her zaman önemli ve sık karşılaşılan bir olgu olmuştur. Bu tez çalışması, kendisini bu akademik araştırma sürecinin bir parçası olarak konumlandırmaktadır.

Bu tez çalışması sırasında yapılan araştırmalar, ses sinyali işlemenin bir dizi kategorisindeki problemlere farklı çözümler önermek veya var olan çözümlerle elde edilen sonuçları iyileştirmek için Kesirli Mertebeden Kalkülüs'ün (Fractional Order Calculus, FOC) nispeten daha az uygulanan matematiksel çerçevesine dayanan yeni yöntemler önermeyi ve yöntemlerin yeteneklerini incelemeyi amaçlamaktadır.

FOC üç asırdan fazla bir geçmişe sahip bir matematiksel çerçevedir. Bu yüzyıllara uzanan araştırma tarihi boyunca, matematiğin bu alanı ve buradan doğan uygulamalar, tamsayı mertebeli türev-integral çerçevesine dayanan uygulamalara göre daha az ilgi çekmiştir. Özellikle sinyal işleme alanında, yerleşik kalkülüse dayanan yöntemler, tamsayı mertebeli kalkülüsünün gerçel sayılara genelleştirilmiş muadiline göre doğal basitliğe sahip olasından dolayı sıkça başvurulan yöntemler olmuştur.

Bu tez boyunca sunulan yaklaşımların uygulama alanı ses sinyali işleme olarak seçilmiştir. Bu seçimin nedeni, insan makine etkileşimi ve insan iletişiminin çağdaş doğasının gereklilikleridir. Ses Sinyali İşleme problem kategorileri iki ana başlık altında toplanabilir. Bu iki konu sınıflandırma ve sentez problemleridir. Bu iki konuyla ilgili problemleri çözmeye yönelik uygulamalarda, uygulamaya göre özelleşmiş öznitelik yaklaşımları ve çıkarım modelleri geçmişte yaygın olarak kullanılmıştır. Ancak derin sinir ağları temelli uygulamaların yaygınlık kazanmasıyla birlikte veriye dayalı yaklaşımlar yaygınlık kazanmıştır.

Sinir ağı araştırmalarının tarihi, başta işaret edilen bilimsel araştırma süreçlerine benzer bir gelişim göstermektedir. İlk kavramsallaştırılmalarından uzun bir süre sonra ve mevcut donanım cihazlarının gelişmiş işleme yetenekleriyle birlikte, sinir ağı modelleri veya yaygın olarak bilinen adıyla derin öğrenme mimarileri, sinyal işleme ve dolayısıyla ses sinyali işleme alanlarında devrim yaratmıştır. Bu devrimin doğrudan sonucu olarak, yeni akıllı sistemlerin tasarımlarında, verilere yaklaşım şekli değiştirmiştir. Günümüzde yaygın olarak kullanılan çıkarım modellerinin eğitim yöntemleri artan sayıda veriye ihtiyaç duymaktadır. Bu nedenle, var olan sayısal işaret işleme yöntemleri, hem derin öğrenme modellerinin parametre sayılarını azaltarak bellek boyutlarını düşürmek için yardımcı araçlar olarak hem de derin öğrenme modelleriyle elde edilen çıkarım sonuçlarının başarılarını arttırmak için veri uzayını büyütmek amacıyla kullanılır hale gelmiştir.

Bu tez çalışmasında, geleneksel ve çağdaş ses sinyali sınıflandırma ve ses sinyali sentezleme yaklaşımlarının problem çözme yeteneğini artırmak için FOC tabanlı yeni yöntemler önerilmiştir. Tez çalışmasının giriş bölümünün amacı, FOC ve ses işareti işlemenin güncel sorunları gibi tezin temel kavramları hakkında genel bir anlayış oluşturmaktır. Güncel ses işleme yöntemlerinin büyük sayıda veriye ihtiyaç duyması ve derin öğrenme modellerinin eğitim parametrelerin azaltılması önemli mühendislik problemleridir. Giriş bölümünde, bu iki önemli mühendislik problemine yanıt verirken FOC temelli yeni yöntemler denemenin ve tasarlamanın geçerliliği tartışılmaktadır. Dolayısıyla, bu tez temelde FOC çerçevesinin, çağdaş sinir ağı tabanlı ses sınıflandırma ve sentezleme yaklaşımlarından kaynaklanan bir dizi probleminin çözümüne nasıl katkı verebileceğini tartışmaktadır.

Tez çalışmasının ikinci bölümü FOC matematiksel çerçevesinin tanıtılmasına odaklanmıştır. Bu çerçeve mühendislik problemlerinin çözümünde yeni olanaklar sunarken bu çözümlerin de karmaşıklığının artmasına neden olmaktadır. Bu tez çalışmasında artan çözüm karmaşıklığına odaklanmak bilinçli olarak tercih edilmemiştir. Ancak belirtilmelidir ki, özellikle işaret işlemede, karmaşıklığın artması sayısal hesaplama maliyetinin de artmasına neden olmaktadır. Bununla birlikte, bu çerçevenin yetenekleri birçok araştırmacının ilgisini çekmeye devam etmiştir. Literatürde, FOC'nin birçok mühendislik uygulaması bulunabilir. Özellikle işaret işleme, zaman serisi tahmini ve görüntü iyileştirme uygulamaları gibi konulardan oluşan literatür, bu tezdeki önermelere ait kavramların önemli bir kısmını sağlamıştır.

Bu tez, 3. bölümden başlayarak, her bölüm ses işareti işlemenin belirli bir uygulama kategorisine odaklanacak şekilde yapılandırılmıştır. Her bölümde, odaklanılan problem için ilgili literatür arka planı bölümün başında sunulmaktadır. Ardından, önerilen FOC tabanlı yöntemler tanımlanmış ve bu yöntemlerin yeteneklerini göstermek için deneyler gerçekleştirilmiştir.

Bölüm 3'te değinilen problem ses sınıflandırma problemidir. Bu probleme ait alt kategoriler, ilgili literatürün gösterdiği gibi ele alınan ses işaretinin türüne göre belirlenmektedir. Üç temel ses işareti türü konuşma, müzik ve çevresel ses olarak sınıflandırılmaktadır. Diğer iki ses işareti türünün aksine, çevresel ses işaretleri için mevcut veri kümelerin boyutu görece azdır. Çevresel ses verilerinin bu yönü, çevresel ses işleme uygulamaları için önemli bir mühendislik problemidir. Bu önemli problem veri artırma yöntemleri alanındaki araştırmaları da teşvik etmektedir. Tez çalışmasının 3. Bölüm'ünde literatürdeki veri artırma yöntemleri hakkında bilgi verilmekte ve özellikle bir "çarpıtma (warping)" yaklaşımını kullanan VTLP yöntemi ana hatlarıyla açıklanmaktadır. Bu bölümde toplam üç adet FOC tabanlı veri artırma yöntemi sunulmaktadır. İlk yöntem Kesirli Mertebeden Maske (Fractional Order Mask) olarak adlandırılmaktadır. Bu yöntemde, kesirli mertebeden türev-integralin Grünwald-Letnikov tanımında gösterilen filtreleme katsayıları kullanılmaktadır. Literatürde benzer yaklaşımlar görüntü iyileştirme ve kenar algılama uygulamaları için kullanılmaktadır. Önerilen veri artırma yöntemi, α türev mertebesi için 5x5 2D evrişim matrisi üretir. Hesaplanan filtre parametreleri b_0, b_1, b_2, 5x5 evrişim matrisinin merkezinden başlayarak 8 yöne yerleştirilir. Önerilen yöntem, ses verilerinin tüm 2D gösterimlerine uygulanabilir. Ses verileri için en çok kullanılan gösterimler Mel Spektrogram ve Log-Mel Spektrogram gösterimleri olduğundan, önerilen yöntemin incelendiği deneyler bu iki ses verisi gösterimi ile sınırlandırılmıştır.

3. Bölümde önerilen diğer veri artırma yöntemleri Kesirli Mertebeden Frekans Ölçeği ve Kesirli Mertebeden Mel Ölçeği yöntemleri olarak adlandırılmaktadır. Bu iki

yöntem, Bu yöntemde, kesirli mertebeden türev-integralin Riemann-Liouville tanımını için literatürde sunulmuş sayısal yöntemleri kullanmaktadır. Bu yöntemler özünde frekans ölçeği veya Mel ölçeği için yeniden ölçeklendirme araçları olarak tasarlanmışlardır. Yeniden ölçeklendirme ile spektrogram gösterimlerinin frekans indislerine karşılık düşen değerler bir α türev mertebesine göre ilgili fonksiyonun türev ya da integralinin alınmasıyla yeniden belirlenir. 3. bölümdeki deneyler çevresel ses sınıflandırma problemine odaklandığından, Kesirli Mertebeden Frekans Ölçeği yöntemleri bu bölümdedeki deneylerde kullanılmamılşır. Ancak 5. bölümde incelenen problem için bu yöntem önem arz etmektedir. 3. bölümdeki deneyler, sinir ağı modellerinin sınıflandırma doğruluklarındaki artışları incelemek amacıyla gerçekleştirilmiştir. Bu amaçla, deneyler hem 5 katmanlı bir evrişimsel sinir ağı (Convolutional Neural Network, CNN) modeli hem de 18 katmanlı bir ResNet üzerinde gerçekleştirilmiştir. Sonuçlar, önerilen yöntemler birleştirleştirilerek uygulanan bir veri artırma prosedürünün, giriş verisi olarak Log-Mel Spektrogramları kullanıldığında 5 katmanlı CNN için %7,7'ye kadar ve ResNet için %9'dan fazla artış sağlayabildiğini göstermektedir. Elde edilen doğruluk sonuçlarının literatürdeki kıyaslama sonuçlarını aşmadığını altını çizmek gerekir. Ancak deneylerin kapsamı matematiksel bir çerçevenin yeteneklerini göstermekle ilgili olduğundan, elde edilen doğruluk sonucu artışları bu tezin varsayımlarının geçerli olduğunu vurgulamaktadır.

Bölüm 4, ses sentezi probleminin literatür arka planını, tarihçesini ve günümüzdeki önemini sunarak başlamaktadır. Ses işaretleri uygulamarda çoğunlukla öznitelik uzaylarında temsil edildiğinden, bir işaretin zaman düzlemi gösteriminin tekrar inşa edilmesi oldukça önemli bir mühendislik problemidr. Literatürde, özellikle ses kodlama alanında cok sayıda ses tekrar inşa yöntemi önerilmiştir. Bu bölümde önerilen ses işareti tekrar inşa yöntemi ile ilişkisi nedeniyle özellikle SPSI yöntemi ile faz tekrar inşa ve spektral tutarlılık tabanlı yinelemeli yöntemler hakkında bilgiler derinleştirilmiştir. Bu yöntemler, bir genlik spektrogramından geçerli bir zaman düzlemi ses işareti oluşturulmasına odaklanmaktadır. Spektral tutarlılık tabanlı yaklaşımlar, Griffin-Lim Algoritması (Griffin-Lim Algorithm, GLA) ve ondan temellenerek üretilen diğer yinelemeli yöntemlerin uygulanmasına olanak vermektedir. GLA uygulamalarında ilk faz spektrogram tahminleri ile bir genlik spektrogramının üzerinde gerçekleştiren, ardışık matris projeksiyonları kullanılmaktadır. Bir faz tekrar inşa algoritmasının başarısı genellikle spektral yakınsama (SC) ve PESQ ölçü değerlerine göre gösterilir. Önerilen Kesirli Mertebeden Diferansiyel Denklem (Fractional Order Differential Equations, FDE) tabanlı yöntemin yeteneklerini kanıtlamak için, bu ölçümler yapılan deneyslerde kullanılmışlardır. Bu tezde önerilen yöntem, bir işaretin Güç Spektral Yoğunkuk Fonksiyonu'nu (Power Spectral Density Function, PSDF) kesirli mertebeden diferansiyel denklemlerle modelleme yaklaşımını kullanmaktadır. Bu yaklaşım ARFIMA zaman serisi tahminlerinde yaygın olarak kullanılmaktadır. En küçük kareler tahmini ile FDE'nin mertebesi, β parametresini tahmin ederek, kesirli diferansiyel denklemlerin uzun hafıza özelliği, pencerelenmiş bir ses spektrumunun bir tepe noktasının anlık frekansının tahminiyle ilişkilendirilmektedir. Bu ilişkiyi SPSI'ya benzer bir şekilde genlik spektrogramına genişletilmiş yeni bir faz tekrar inşa yöntemi sunulmuştur. Önerilen yöntemin performansını görmek için uygulanan ilk deney grubu, önerilen yöntemin ve SPSI'nın pencere adım boyutunu değiştirerek SC ve PESQ nitel ölçülerine göre karşılaştırılmasını amaçlamaktadır. Sonuçlar, FOC çerçevesinin kullanımının SPSI'dan %4'e kadar daha iyi sonuçlar üretiğini göstermektedir. Ek deneyler, bu yöntemlerin spektral tutarlılık tabanlı yinelemeli yöntemler, yani GLA, FGLA, GLA-ADMM için ilk faz tahmin edicileri olarak kullanımını incelemektedir. İkinci grup deneylerin sonuçları, anlık faz tahmini için FDE modelini uygulayan önerilen ses tekrar inşa yönteminin yeteneklerinin vurgulamıştır.

Sinir Ağlarıyla Ses Sentezi (Neural Audio Synthesis, NAS) oldukça ilgi çeken çağdaş araştırma ve uygulama konusudur. Bölüm 5'teki literatür özetinde gösterildiği gibi, derin sinir ağlarındaki gelişmeler, ses sentezi problemi için oldukça başarılı sonuçlar sağlamaktadır. NAS'nin temel kavramlarında görüldüğü gibi, bu alandaki önemli bir çalışma, farklı donanım tiplerinde uygulanabilmesi için esnek ve daha küçük boyutlu modeller sağlamak için bir sinir ağı uygulamasının parametre sayısını azaltmaya odaklanmıştır. Bu amaçla, filtre tabanlı ses üretimi modelleri sinir ağları ile birleştirilmiştir. Son zamanlarda, spektral tutarlılık yaklaşımı bir sinir ağı ile uygulanmaktadır. Derin Griffin-Lim Yinelemesi (Deep Griffin-Lim Iteration, DeGLI) olarak isimlendirilen bu konsept, daha az sayıda eğitim parametreleri ile başarılı bir ses tekrar inşa süreci oluşturmayı amaçlamaktadır. DeGLI mimarisi, ardışık projeksiyonlar ve bir sinir ağından oluşan daha alt modüllerin tasarımına ve birleştirilmesine dayanır. Bu mimarinin önemli katkısı, bir gürültü arındırıcı (denoiser) olarak eğitilebilmesidir. Bu şekilde eğitilen bir alt modül, kopyaları ile birleştirilir. Bu yaklaşım standart GLA'daki yineleme prosedürünü taklit eder. DeGLI boyut konusunda esneklik sağlar. Donanım kısıtlamalarına bağlı olarak, belirli bir sayıdaki alt modüller birleştirilebilir. Bu tez çalışmasında, FOC tabanlı yöntemlerin NAS problemine uygulanabilirliğini değerlendirmek için bölüm 3 ve 4'ün sonuçları bir araya getirilmiştir. Bu bağlamda, bölüm 3'te önerilen Kesirli Mertebeden Frekans Ölçeği veri artırma yöntemi veri sayısını artırmak için kullanılmaktadır. Başlangıç fazının hesaplanması için önerilen FDE tabanlı faz kestirimi yaklaşımı kullanılmıştır. Bu tezin deneylerinin gerçekleştirildiği donanım, daha küçük boyutlu bir sinir ağının ve daha az miktarda veri ile daha düşük bir örnekleme frekansının uygulanmasını gerektirmiştir. İlk deneyin sonuçları, önerilen ilk faz kesitirimi yönteminin, uygulandığı DeGLI modeli üzerinde standart GLA ile benzer etkiye sahip olduğunu göstermektedir. Bu deneyin sonuçları PESQ ölçüsüne göre gösterilmiştir. Sonuçlar ayrıca, uygulama için parametre boyutunun azaltılmasının daha kötü tekrar inşa kalitesiyle sonuçlandığını göstermiştir. Bu sonuçtan yola çıkarak standart GLA takip eden deneylerin dışında tutulmuştur. Ses tekrar inşa kalitesini artırmak için, önerilen Kesirli Mertebeden Frekans Ölçeği yöntemini kullanan iki veri artırma stratejisi test edilmiştir. Testler için veri kümesi boyutu, farklı α türev mertebesi değerleriyle artırılarak üç katına çıkarılmıştır.Test edilen iki stratejinin önemli farkları şöyledir: Birinci Strateji doğrudan karmaşık spektrogramı kullanır ve veri artırma yöntemini uygulayarak veri sayısını artırır. Buna karşılık, İkinci Strateji Kesirli Mertebeden Frekans Ölçeği yöntemi uygulanmış verinin genlik spektrogramını hesaplar ve FDE tabanlı faz tahmin yöntemini kullanarak yeni bir veri üretir. Birinci Strateji ile yapılan deneyler tekrar inşa kalitesi açısından oldukça başarısız sonuçlar vermiştir. Bunun nedeni bu stratejinin spektral tutarlılık üzerindeki etkisidir. İki büyütme yönteminin SC açısından karşılaştırılması, İkinci Strateji'nin spektral olarak daha tutarlı veriler üretebildiğini göstermiştir. Sonuç olarak, bu yeni strateji, uygulanan DeGLI modelinin ses spektrogramı tekrar inşa kalitesini artırmıştır. Bu sonuç, ses sentezleme modellerini eğitmek için spektral olarak tutarlı artırılmış veriler üreten veri artırma yöntemlerinin potansiyel yetenekleri için bir sezgi sağlamaktadır.

Nihayetinde bu tez çalışmasının sonuçları üç grupta toplanabilir. İlk olarak, FOC matematiksel çerçevesinin günümüzde çoğunlukla uygulanan veri odaklı yaklaşımların önemli bir sorunu olan fazla sayıda veri ihtiyacına yeni veri artırma yöntemlerini tasarlamak konusunda katkı koyabileceği gösterilmiştir. İkinci olarak, FDE tabanlı yaklaşımların, bir konuşma işaretinin faz bileşeni tahmini için başarılı çözümler üretebildiği gösterilmiştir. Ek olarak bu yaklaşım, veri odaklı ses sentezleme yöntemleriyle birlikte de başarı göstermiştir. Üçüncü bir sonuç, spektral olarak tutarlı spektrogramlar üreten bir veri artırma stratejisinin sinir ağları temelli ses tekrar inşa modelleri için daha yüksek başarım sağlayabileceğinin ortaya konmasıdır. Bu üç sonuç, çağdaş ses sınıflandırma ve sentezleme çalışmalarının sorunlarını ele almak için Kesirli Mertebeden Kalkülüs temelli yöntemlerin olumlu sonuçlar doğurabileceğini vurgulamaktadır.

1. INTRODUCTION

It has always been an important and recurrent phenomenon that employing an unconventional approach or using tools of a less established mathematical framework to solve problems of a scientific research area, which has been a domain of methods that are based on standardized approaches, creates a potential of novel solutions and provides opportunities for increased capabilities in that area of scientific research. This thesis dissertation claims to place itself on to this process of academic research.

The researches that are conducted during this thesis work aimed to design and understand the capabilities of the novel methods, that are based on relatively less established mathematical framework of Fractional Order Calculus for solving or improving results of problems of a number of categories of audio signal processing.

Fractional Order Calculus (FOC) is a framework that has a history of more than 3 centuries. During this centuries of history of research, this mathematical research area and the applications that stem from this framework have attracted lesser interest than the well-established integer order calculus. Especially in signal processing, tools of standard calculus have been the methods that provided massive number of approaches, due to integer orders calculus' inherent simplicity with respect to its generalized counterpart.

The application domain for the proposed approaches that are presented throughout this dissertation selected as audio signal processing. This selection has a reasoning because of the needs of the contemporary nature of human machine interaction and human communication. Audio Signal Processing problem categories can be gathered together under two main topics. These two topics are classification and synthesis problems. The applications to solve related problems of these two topics had been extensively employing previous, "tailored" feature extractors and inference models until the current surge of data driven approaches together with deep neural networks.

The history of neural network research demonstrates the similar experience as the scientific research processes defined in the first paragraph of this dissertations. Long after their first conceptualization and together with the improved processing capabilities of present hardware devices, neural network models or with their widely known name, deep learning architectures have revolutionized the fields of signal processing and consequently, audio signal processing. The direct result of this revolution changed the way of approaching the data for the novel intelligent system. In coherence with the way of training of this very capable inference models and due to their needs for increased data numbers, previously defined digital signal processing (DSP) methods became both helper tools to reduce the parameter numbers for deep learning models to decrease their memory size and tools to increase data representation space to provide increased inference results with deep learning models.

In this thesis dissertation, novel methods that are based on FOC has been proposed for increasing problem solving capacity of traditional and contemporary approaches of audio signal classification and audio signal synthesis. To provide a coherent explanation process, the related literature backgrounds for FOC, audio classification problem and audio synthesis problem are provided in the related chapters of this dissertation.

1.1 Purpose of Thesis

This thesis dissertation has several purposes, which are gathered under four chapters. The purpose of Chapter 2 is to provide background information about Fractional Order Calculus. This chapter begins with historical establishment of FOC, and discusses the capabilities and the reasons of its lack of use with respect to the integer or calculus framework. The definition of two most widely used fractional derivatives, which are also used in this dissertation are provided in this chapter. Chapter 2 also aims to provide an understanding the relation of FOC and the theory of fractals. Additionally, a literature background of engineering applications of FOC in the sense of signal processing is also provided in this chapter.

The aim of Chapter 3 is to experimentally show that FOC based approaches can be designed as data augmentation methods for audio signal classification. To provide the reasoning behind the experiments for this purpose, a deductive approach is adopted. In that sense, this chapter begins with categorizing audio classification problem based on type of data. To provide the insight about representation of audio data, Chapter 3 explains audio signal feature extraction methods and provides a detailed overview of Mel Spectrogram representation. Additionally, literature background for intelligent systems that are used for audio classification problems are given with respect to data type being speech, music or environmental sound data. The experiments conducted in this chapter is focused on environmental sound classification (ESC) problem. As shown by the literature for this problem, data augmentation methods have an important implementation area. Consequently, this chapter also includes examples of important data augmentation literature and additionally a specific method that provides a similar approach to the designed methods in this thesis work. In Chapter 3 three FOC based data augmentation methods are proposed. Numerical methods for fractional derivative calculation is also explained in this chapter. By applying data augmentation methods to two neural network models; one being a similar architecture to the neural networks that are used in ESC literature and the other being a well-documented deep learning model, the capabilities of FOC based data augmentation methods on neural network based classification accuracy results are evaluated.

The purpose of Chapter 4 is to present a FOC based instantaneous frequency estimation method for phase reconstruction problem of audio synthesis topic. This chapter begins with broader definition of audio synthesis and its relation to contemporary problems. With providing literature background, chapter 4 also provides definition of vocoder, information about different vocoding methods and phase vocoders. Since its importance for audio reconstruction topic, phase reconstruction problem is explained. A special importance is dedicated to the concept of spectral convergence and spectral convergence based iterative methods for audio reconstruction in chapter 4. Following, the mathematical backbone of the proposed method and its relation to fractional order time series models is explained. Together with phase vocoder method a novel phase reconstruction method is presented. The performance of the method is evaluated with respect to objective measures in comparison to a phase reconstruction method in literature. This comparison is provided with respect to different window overlap values to show the performance in terms of spectral resolution. Additionally, the phase reconstruction method is employed as an initialization method for three spectral consistency based iterative audio reconstruction algorithms. The related results are presented with comparison to the different initialization approaches.

The aim of chapter 5 is to show that FOC based data augmentation strategies can produce capable tools for improving performance of neural network based audio reconstruction models. Chapter 5 begins with explaining elementary concepts of neural audio synthesis (NAS) together with relevant literature review. This chapter provides examples of important approaches to reduce the number of parameters of a neural vocoder design. Additionally, a recently proposed, spectral consistency based neural audio reconstruction model is explained in depth. For the purpose of this chapter, a smaller sized neural network is implemented. After providing the insight about neural network implementation and dataset, first experiment is conducted by combining the phase estimation approach, which is explained in chapter 4. By making use of the linear scale based data augmentation method that is proposed in chapter 3, two differing augmentation strategies are tested in continuation. Lastly in chapter 5, it is shown that, a data augmentation strategy, which takes into consideration the spectral consistency, can increase audio reconstruction performance of a neural network based approach.

1.2 Contribution and Hypothesis

The main perspective that motivates the hypothesis of this dissertation is exploring the capabilities of fractional order calculus framework on contemporary audio signal processing problems. The research and experiments that are conducted for this motivation produced several important contributions to the literature.

a. 2 novel FOC based data augmentation methods are proposed.
b. A FOC based edge enhancement mask approach in the literature is restructured and redesigned with the aim of data augmentation.
c. The novel data augmentation methods are shown to produce promising results for ESC with deep learning
d. Differently from conventional vocoders, a FOC based model, which exploits the memory feature is shown to be capable of estimating instantaneous frequency and consequently provides acceptable reconstructed phase.
e. A data augmentation strategy, which takes into consideration the spectral consistency, can increase audio reconstruction performance of a neural network based approach.

In addition to these main contributions, in chapter 4, it is shown that proposed phase reconstruction method achieves better results than a similar approach in literature for increasing window overlap rates. This result implies that for higher resolutions, the proposed method is preferable. The experiments in chapter 5 shows that, together with phase initialization, the performance of the audio reconstruction neural network which is trained with correctly augmented data provides a substantial increase.

All in all, the hypothesis can be given in the following summarized form: "*The fractional order calculus framework is a capable mathematical framework for providing solutions and novel methods for a number of problems of contemporary neural network based audio signal classification and reconstruction.*"

2. WHAT IS FRACTIONAL ORDER CALCULUS?

It is almost compulsory to talk about the the letter sharing between L'Hospital and Leibniz in any research related to Fractional Order Calculus. For this dissertation, it is decided that there is no need for any examption. The concept of FOC, or an arbitrary order derivative and integral is more than 300 years old. During this three centuries most renowned mathematicians of their times such as L'Hospital and Leibniz as mentinoned, Euler, Lagrange in 18^{th} century and Laplace in 19^{th} century contributed to the concept directly or indirectly [1].

FOC is a generalization of integer order differentiation and integration [2]. The motivation behind the need of generalization of differentiation and integration orders is because of the increasingly accurate description and modelling needs of real world objects and dynamical behaviour of real world systems. The mathematical phenomena of FOC makes it possible to describe real objects more accurately than classical, integer order calculus [2]. Due to the extra free parameter of the non-integer derivation order, fractional order-based methods provide an additional degree of freedom in modeling objects, optimizing performance, and describing natural dynamical behavior with memory [3]. Additionally, FOC is shown to be capable of developing tools for signal processing [4].

The FOC concept has been a relatively less explored field in the various fields of engineering. The reason of this situation stems from mainly two important reasons [5]. First reason is because of the apparent and proven self-sufficiency of classical integer order calculus. The established methods for solving engineering problems in terms of theory and application algorithms are dominantly based on integer order differential equations. Since these approaches provide less complex and established tools, the research and application preferences lean towards integer order differential equation models. The other important reason of this lack of use of FOC based tools comes from a more elemantary reason, especially as it can be seen from the perspective of an engineer. In engineering applications, it is preferred that the basic definitions and assumptions be standardized. The mathematicians are still not agreed on a definition of fractional order differentiation or integration operators. As of today, FOC does not have a fully acceptable geometrical or physical interpretation [5].

As stated before, the FOC is a name for the theory of integrals and derivatives of arbitrary order, which unify and generalize the notions of integer-order differentiation and n-fold integration. This unified notion lead to the differintegral operator as defined in equation (2.1) [3].

$$
{}_{\mathrm{a}}D_{\mathrm{t}}^{\alpha} = \begin{Bmatrix} \dfrac{d^{\alpha}}{d\mathrm{t}^{\alpha}} \quad \alpha > 0 \\ 1 \quad \alpha = 0 \\ \displaystyle\int_{\mathrm{a}}^{\mathrm{t}} (d\tau)^{\alpha} \quad \alpha < 0 \end{Bmatrix} \tag{2.1}
$$

In the literature, numereous definitions for fractional derivatives are proposed, such as Riemann-Liouville (RL), Grünwald-Letnikov (GL), Caputo, Hadamard or Marchoud definitions [6]. The choice of fractional order derivative operator generally depends on the application and the preference of the researchers [7]. In their works, Ortigueria and Machado provided some properties for accepting an operator as a fractional derivative operator [8]. According to them, a fractional derivative should satisfy following five properties.

- The operator should be linear.
- The operator should satisfy identity property as in equation (2.2).

$$
D^{0} f(\mathrm{t}) = f(\mathrm{t}) \tag{2.2}
$$

- The operator should give the same result as integer order derivatives for integer order differentiation order values.
- The operator should satisfy the index law as in equation (2.3) for $\alpha < 0$ and $\beta < 0$.

$$
D^{\propto}[D^{\alpha} f(\mathrm{t})] = D^{\alpha+\beta} f(\mathrm{t}) \tag{2.3}
$$

- The operator should satisfy the Generalised Leibniz rule as given in equation (2.4).

$$D^{\alpha}[f(\mathrm{t})g(\mathrm{t})]=\sum_{\mathrm{i}=0}^{\infty}\binom{\alpha}{\mathrm{i}}D^{\mathrm{i}}f(\mathrm{t})D^{\alpha-\mathrm{i}}g(\mathrm{t}) \tag{2.4}$$

- In a recent work that includes Ortigueira, Valerio et. al. included an additonal property in the sense of signal processing [6]. According to this work a fractional order dervative operator should satisfy the following equation.

$$\mathcal{L}[D^{\alpha}f(\mathrm{t})]=s^{\alpha}F(s) \tag{2.5}$$

where $F(s)$ the Laplace transform of the function $f(\mathrm{t})$ as given in equation (2.6).

$$F(s)=\mathcal{L}[f(\mathrm{t})] \tag{2.6}$$

According to aformentioned papers, RL and GL definitions are valid definitions of fractional order derivatives. Additionally, the numerical methods for estimating these definitions of fractional order derivatives are provided in programming language libraries [9]. Because of these reasons the methods proposed throughout this dissertation are based on these two definitions of fractional order derivatives.

2.1 Grünwald-Letnikov Definition

The Grünwald-Letnikov definition is one of the most frequently used definitions for the general fractional differintegral. The first derivative of a continuous function $f(\mathrm{t})$ can be expressed as follows.

$$\frac{d}{dt}f(\mathrm{t})\equiv f'(\mathrm{t})=\lim_{\mathrm{h}\to 0}\frac{f(t)-f(\mathrm{t}-\mathrm{h})}{\mathrm{h}} \tag{2.7}$$

The second derivative of the function can be calculated as in equation (2.8).

$$\begin{aligned}\frac{d^2}{dt^2}f(\mathrm{t})\equiv f''(\mathrm{t})&=\lim_{\mathrm{h}\to 0}\frac{f'(\mathrm{t})-f'(\mathrm{t}-\mathrm{h})}{\mathrm{h}}\\ &=\lim_{\mathrm{h}\to 0}\frac{1}{\mathrm{h}}[\frac{f(\mathrm{t})-f(\mathrm{t}-\mathrm{h})}{\mathrm{h}}-\frac{f(\mathrm{t}-\mathrm{h})-f(\mathrm{t}-2\mathrm{h})}{\mathrm{h}}]\\ &=\lim_{\mathrm{h}\to 0}\frac{f(\mathrm{t})-2f(\mathrm{t}-\mathrm{h})+f(\mathrm{t}-2\mathrm{h})}{\mathrm{h}^2}\end{aligned} \tag{2.8}$$

Consequently, the third derivative can be calculated as follows.

$$\frac{d^3}{dt^3} f(\mathrm{t}) \equiv f'''(\mathrm{t})$$
$$= \lim_{\mathrm{h}\to 0} \frac{f(\mathrm{t}) - 3f(\mathrm{t}-\mathrm{h}) + 3f(\mathrm{t}-2\mathrm{h}) - f(t-3\mathrm{h})}{\mathrm{h}^3} \tag{2.9}$$

Using these equations, the derivation operation can be generalized.

$$\frac{d^n}{dt^n} f(\mathrm{t}) \equiv f^{(n)}(\mathrm{t})$$
$$= \lim_{\mathrm{h}\to 0} \frac{1}{\mathrm{h}^n} \sum_{\mathrm{j}=0}^{n} (-1)^{\mathrm{j}} \binom{n}{\mathrm{j}} f(\mathrm{t}-\mathrm{jh}) \tag{2.10}$$

Replacing integer n with real number α the equation can be rewritten [2].

$$D_{\mathrm{t}}^{\alpha} = \lim_{\mathrm{h}\to 0} \frac{1}{\mathrm{h}^{\alpha}} \sum_{\mathrm{j}=0}^{\infty} (-1)^{\mathrm{j}} \binom{\alpha}{\mathrm{j}} f(\mathrm{t}-\mathrm{hj}) \tag{2.11}$$

Here, the factorial can be defined with respect to the Gamma function as follows [2],

$$\binom{\alpha}{\mathrm{j}} = \frac{\alpha!}{\mathrm{j}!\,(\alpha-\mathrm{j})!} = \frac{\Gamma(\alpha+1)}{\Gamma(\mathrm{j}+1)\Gamma(\alpha-\mathrm{j}+1)} \tag{2.12}$$

The definition of Euler's Gamma function, which is the generalization of factorial is given in equation (2.13) [2].

$$\Gamma\,(\mathrm{n}) = \int_0^{\infty} \mathrm{t}^{\mathrm{n}-1} e^{-\mathrm{t}}\, d\tau \tag{2.13}$$

The relation of Gamma function to the factorial is given in equation (2.14) [3].

$$\Gamma\,(\mathrm{n}) = (\mathrm{n}-1)! \tag{2.14}$$

2.2 Riemann-Liouville Definition

The Riemann-Liouville definition comes from the n-fold integration [3].

$$\int_a^{\mathrm{t}} \int_a^{\mathrm{t}_n} \int_a^{\mathrm{t}_{n-1}} \dots \int_a^{\mathrm{t}_3} \int_a^{\mathrm{t}_2} f(\mathrm{t}_1) d\mathrm{t}_1 d\mathrm{t}_2 \dots d\mathrm{t}_{\mathrm{n}-1} d\mathrm{t}_{\mathrm{n}} = \frac{1}{\Gamma\,(\mathrm{n})} \int_a^{\mathrm{t}} \frac{f(\tau)}{(\mathrm{t}-\tau)^{1-\mathrm{n}}} d\tau \tag{2.15}$$

In equation (2.15), $\mathrm{n} \in \mathrm{N}$ and $\mathrm{n} > 0$. Consequently, the RL fractional integral can be given as in equation (2.16) [3].

$$ {}_{a}D_{t}^{-\alpha} = \frac{1}{\Gamma\,(-\alpha)} \int_{a}^{t} \frac{f(\tau)}{(t-\tau)^{\alpha+1}} d\tau \tag{2.16} $$

For $(n - 1 < \alpha < n)$, where a and t are the limits of operation the derivation operation can be defined as equation (2.17) [3].

$$ {}_{a}D_{t}^{\alpha} = \frac{1}{\Gamma\,(n-\alpha)} \frac{d^n}{dt^n} \int_{a}^{t} \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \tag{2.17} $$

The RL fractional derivative is defined by combining integer order derivative and fractional integration.

2.3 Engineering Applications of Fractional Order Calculus

Throughout its nearly four centuries of history FOC and its related concepts has been applied to the numerous fields of engineering. Specifically the extra free parameter of the non-integer derivation order provides an additional degree of freedom for modelling objects, optimizing system or filter performance and describing natural dynamical behaviour with memory using fractional order-based methods [3]. An important application of FOC on time series analysis is called Autoregressive Fractionally Integrated Moving Average (ARFIMA) model [4].

The Moving Average (MA) model with q terms MA (q) is of the form as in (2.18), for the coefficients $\theta_i, i \in \{1, \ldots, q\} \subseteq Z^+$.

$$ y_n = x_n + \theta_1 x_{n-1} + \cdots + \theta_q\, x_{n-q} \tag{2.18} $$

The Auto Regressive (AR) models with p terms, AR (p) are of the form as in (2.19), for some coefficients $\varphi_j, j \in \{1, \ldots, p\} \subseteq Z^+$.

$$ y_n = \varphi_1\, y_{n-1} + \cdots + \varphi_p\, y_{n-p} \tag{2.19} $$

The Auto Regressive Moving Average (ARMA) model with p AR terms and q MA terms ARMA (p, q) is given by:

$$ y_n = \left(\varphi_1\, y_{n-1} + \cdots + \varphi_p\, y_{n-p}\right) + \left(x_n + \theta_1 x_{n-1} + \cdots + \theta_q\, x_{n-q}\right) \tag{2.20} $$

For a time series x_n and zero mean white Gaussian noise ε_n the ARMA (p, q) model can be represented as:

$$\left(1-\sum_{\mathrm{j}=1}^{\mathrm{p}}\varphi_{\mathrm{j}}L^{\mathrm{j}}\right)x_{\mathrm{n}}=\left(1+\sum_{\mathrm{j}=1}^{\mathrm{q}}\theta_{\mathrm{j}}L^{\mathrm{j}}\right)\varepsilon_{n} \tag{2.21}$$

where L is the lag operator satisfying the expression $L\,x_{\mathrm{n}} = x_{\mathrm{n}-1}$.

The Auto Regressive Integrated Moving Average (ARIMA) model ARIMA (p, d, q) is a generalization of the ARMA (p, q) model with a differencing parameter $d \in Z^{+}$. The ARIMA model can be extended to the ARFIMA model to obtain:

$$\left(1-\sum_{\mathrm{j}=1}^{\mathrm{p}}\varphi_{\mathrm{j}}L^{\mathrm{j}}\right)(1-L)^{d}x_{\mathrm{n}}=\left(1+\sum_{\mathrm{j}=1}^{\mathrm{q}}\theta_{\mathrm{j}}L^{\mathrm{j}}\right)\varepsilon_{n} \tag{2.22}$$

where the lag operation with respect to the differencing parameter is given in equation (2.23).

$$(1-L)^{d}=\sum_{\mathrm{j}=0}^{\infty}(-L)^{d}\binom{d}{\mathrm{j}} \tag{2.23}$$

Consequently, it is easy to see the relation of the lag operation to the GL definition of fractional derivatives.

$$(1-L)^{d}x_{\mathrm{n}}=\sum_{\mathrm{j}=0}^{\infty}(-L)^{\mathrm{j}}\binom{d}{\mathrm{j}}x_{\mathrm{n}}=\sum_{\mathrm{j}=0}^{\infty}(-1)^{\mathrm{j}}\binom{d}{\mathrm{j}}x_{\mathrm{n}-\mathrm{j}}=D^{d}x_{\mathrm{n}} \tag{2.24}$$

The ARFIMA model is an extension of the ARIMA model where the degree of differencing d can have fractional values [10]. ARFIMA models have found applications in economics, biology, remote sensing, network traffic modeling and similar areas. ARFIMA processes are named with respect to the values of their p, d and q parameters. For example an ARFIMA(0,d,0) process can be defined as equation (2.25).

$$D^{d}x_{\mathrm{n}}=\varepsilon_{n} \tag{2.25}$$

The capabilities of the FOC framework connect it to the theory of fractals. Since introduced by Mandelbrot, the concept fractal theory has been a mathematical framework for explaining self-similar structures in nature [11]. The textural information of a signal is an important aspect for understanding the said signal. This information can be modelled by tools of fractal geometry. For example, under the assumption of a stochastic signal obeys a well-defined fractal model, FOC based methods and models can be derived to estimate the frequency characteristics of a signal. Additionally, the model parameters derived in fractal framework can be beneficial in solving problems such as textural segmentation of a signal [12]. Fractal theory helps in explaining the local properties of a signal and in simplifying the geometrical or statistical description of properties of a signal, regardless of the signal being fractal or not [13].

The relationship between FOC and fractal geometry comes from the fractional differentiation order or, more correctly, the differintegration order of a differential equation. The value of the order gives a metric for the long-term dependence and fractal dimension of a signal. Additionally it has been shown that fractional differintegration has possible uses for a signals phase characteristics. For example, it is shown that GL fractional differintegration causes a phase shift for sinusoids with direct relation to its differintegration order [14].

FOC based models have found use in audio signal processing in roughly two ways. The first family of applications of FOC to the audio signal processing aims to reduce the number coefficients of a model, of which the signal is represented. Practically, this means that a FOC based model provides less filter coefficients with respect to an equivalent integer order filter model. FOC based models can be used for reducing the number of linear prediction (LP) parameters of signal, because by differintegrating a signal by an appropriate order, autocorrelation function of the signal can be manipulated to reduce the LP parameters. The increased signal prediction performance of Fractional Linear Prediction (FLP), which is an approach based on a weighted sum of fractional derivatives of a signal, is documented in an application for speech signal prediction problem [15].

Fractional calculus is a nonlocal approach and, therefore, employs infinite memory. It means that the fractional derivative of a signal depends on all the previous values of the signal. This aspect of FOC framework makes it suitable for dealing with signals with memory. For optimal FLP, some approaches with limited memory have been proposed. Proposed methods for optimal FLP with limited memory have not only provided good results on prediction accuracy but also provided a tool for reducing the number of LP coefficients that are needed for encoding an audio signal [16–18].

The FLP of a signal can be defined as in the equation (2.25) [15]:

$$s'[\mathrm{n}] = \sum_{\mathrm{k}=0}^{\mathrm{K}} a_{\mathrm{k}} D^{\alpha_{\mathrm{k}}} s[\mathrm{n}], \quad \mathrm{n} \neq 0 \tag{2.25}$$

Here, a_{k} is FLP coefficient and α_{k} is the fractional derivative order. To define this in matrix form following definitions are given.

$$\boldsymbol{p}_{\mathbf{k}} = [D^{\alpha_{\mathrm{k}}} s[\mathrm{n}]] \tag{2.26}$$

and

$$\Omega = [\boldsymbol{p}_{\mathbf{1}} \quad \boldsymbol{p}_{\mathbf{2}} \quad \cdots \quad \boldsymbol{p}_{\mathbf{K}}] \tag{2.27}$$

The estimate of a signal sample can be defined as follows:

$$\boldsymbol{s}' = \Omega \boldsymbol{a}^T \tag{2.28}$$

Since,

$$\Omega^T \boldsymbol{s}' = \Omega^T \Omega \boldsymbol{a}^T \tag{2.29}$$

and

$$(\Omega^T \Omega)^{-1} \Omega^T \boldsymbol{s}' = (\Omega^T \Omega)^{-1} \Omega^T \Omega \boldsymbol{a}^T \tag{2.30}$$

the equation (2.30) is reduced into the equation (2.31) with $\boldsymbol{s}' \rightarrow \boldsymbol{s}$.

$$\boldsymbol{a}^T = (\Omega^T \Omega)^{-1} \Omega^T \boldsymbol{s} \tag{2.31}$$

$\boldsymbol{a}^T$ contains the FLP parameters. The limited memory apprroach for FLP uses the following estimation as given in the equation (2.32).

$$D^{\alpha} s[\mathrm{n}-1] = \lim_{\mathrm{h}\rightarrow 0} \frac{1}{\mathrm{h}^{\alpha}} \sum_{\mathrm{j}=0}^{\mathrm{J}} (-1)^{\mathrm{j}} \binom{\alpha}{\mathrm{j}} s[\mathrm{n}-1-\mathrm{j}] \tag{2.32}$$

Inserting this equation into FLP definiton in (2.25) the FLP takes the following form in the equation (2.33) [18].

$$s'[\mathrm{n}] = \sum_{\mathrm{k}=1}^{\mathrm{K}} a_{\mathrm{k}} D^{\alpha_{\mathrm{k}}} s[\mathrm{n}-1], \quad \mathrm{n} \neq 0 \tag{2.33}$$

An example of FLP of a music signal can is given in Figure 2.1. A 20ms part of the 22050 Hz sampled signal is estimated with FLP. The derivative orders are chosen as in the literature [15]. The chosen orders are 0.15, 0.3 and 0.45. Additionally, 30 values are taken into considerition for fractional derivation. In this context 3 FLP coefficients must be calculated.

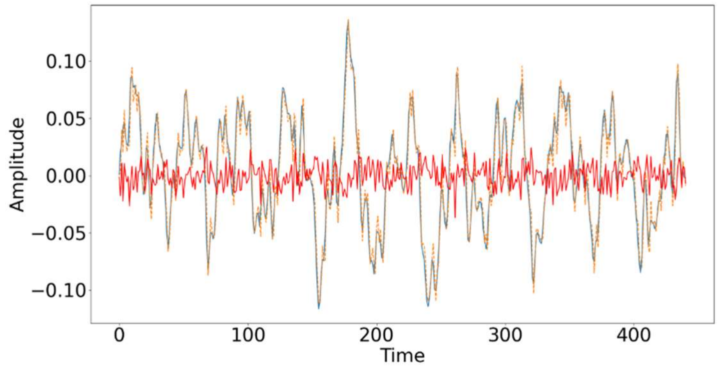


Figure 2.1: FLP estimate (Blue: Original signal, Yellow: FLP estimate, Red: Error).

To compare the results, the 3 coefficient LP of the same sample is given in Figure 2.2.

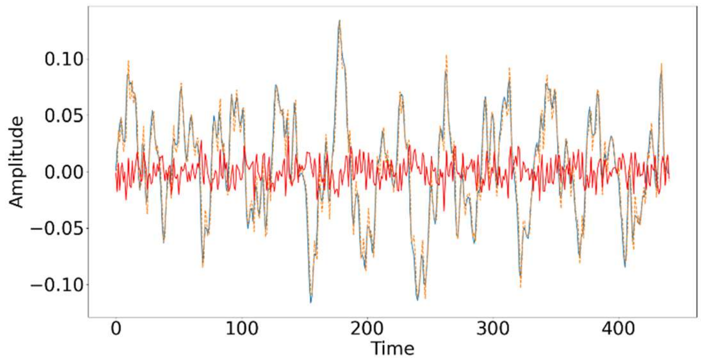


Figure 2.2 : LP estimate (Blue: Original signal, Yellow: LP estimate, Red: Error).

The errors for both estimation approaches are given in Figure 2.3.

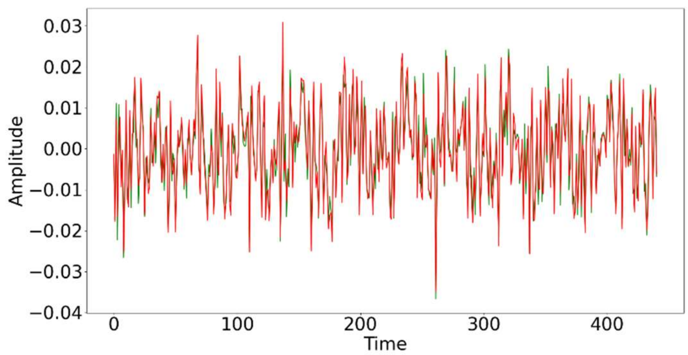


Figure 2.3: Comparison of FLP and LP estimation errors (Green: FLP Error, Red: LP Error**).**

Calculating the power gains for the two estimation procedures with the following formula:

$$Power\ Gain\ (\mathrm{db}) = 10log_{10}\left(\frac{\boldsymbol{s}'\boldsymbol{s}'^{\mathrm{T}}}{(\boldsymbol{s}'-\boldsymbol{s})(\boldsymbol{s}'-\boldsymbol{s})^{\mathrm{T}}}\right) \tag{2.34}$$

the FLP results in 13.31 dB power gain and LP results in 12.75 dB power gain. As it can be seen, FOC based appraches can be implemented to increase signal estimation performance.

Second family of approaches makes use of the relationship of fractional derivative and fractal dimension. In this context, fractional order derivatives are used in audio processing applications as a metric for fractal analysis. In literature, it has been shown that the excitation in an AR model can be assumed as the fractional derivative of Gaussian noise [19].

Fractal theory has been used in the problems of sound classification. The most common form of application of the concept of fractals has been the use of this concept as a tool for generating feature representations. There are successful applications of fractal features to speech recognition, voiced–unvoiced speech separation [20]. Speaker emotion classification is an another area, that fractal theory based features has been applied to [21]. A fusion of fractal geometry-based features was shown to produce comparable results to mel-frequency cepstral coefficients when applied to speech classification problems [22].

Another application area of FOC based approaches in signal processing domain has been image processing tasks [23]. In the most common form of application for this context, Gamma function based parameters of the GL fractional differintegral are treated as discrete filter paramaters. This aproach provides the oportunity to produce fractional differential filter masks for image processing. In practice, the fractional-order masks are used as a part of edge detection algorithms. As the direct consequence of fractional derivative order, which adds a degree of freedom, by only changing the derivative order, filter paramaters can be tuned accordingly to create fractional-order derivative-based filters or masks for increased edge detection or segmentation performance. This approach has found its use in areas from satellite image segmentation [24] to biomedical applications such as brain tomog-raphy segmentation [25].

In the era of deep learning, the FOC has also found its use on the neural network based applications [26-28]. The training procedures of neural networks can be defined as optimization problems and the arbitrary derivative order and memory characteristics of fractional differintegral can be made use of. In a neural network approach for speech recognition task FOC is employed for the means of updating the network weights [29]. The method is used to update the weights iteratively based on the weights of last two iterations. Here the memory property of FOC is beneficial. The consideration of historical weights to find new weights are the main characteristic of the proposed method.

Several recent studies have examined Fractional Order Differential Equations (FDE) in relation to biological models. One area of investigation has been the analysis of bifurcation in fractional-order prey-predator models. These studies have demonstrated that the stability domain can be extended under fractional order, resulting in less chaotic behavior [30, 31]. Another application of fractional-order calculus has been in the realm of Genetic Regulatory Networks, where such models are useful for showing the relationship between gene transcription and mRNA translation in biological cells [32]. Time delay is an important factor in FDE that affects the dynamical behavior of systems. As a result, exploring the impact of time delay on fractional-order neural networks is critical for optimizing and controlling such networks [33]. Understanding bifurcations in fractional-order neural networks is an active area of study, and it is crucial for gaining insight into the dynamical properties of neural networks [34, 35].

3. AUDIO CLASSIFICATION PROBLEM

An audio signal is a digital sound signal whose frequency components generally stays between 20-20000Hz frequency range, which is called the audio band [36]. 20Hz and 20000Hz values are determined as lower and upper limits of human hearing. The values are determined by analysis on human psychoacouistics.

Sound classification and recognition or with a more focused naming based on signal type, audio classification or recognition have an extensive history of research as parts of pattern recognition. An audio classification task can be broadly defined as analysing and categorising an audio signal [37]. The categorising process could be the prediction of the label of an audio sample or segments of an audio stream [38].

The audio classfication task is generally categorised into subtopics based on the type of signal such as speech, music and environmental sounds [39]. Some of the topics that atract research interest mostly are automatic speech recogniton [40], music information retrieval [41] and environmental audio classification [42].

Traditionally, an audio classification task has included two seperate procedure. The first procedure is interested with representation of an audio data and its features [43]. The features of an audio sample is related to its amplitude, duration and frequency characteristics. For convenient representation of the audio signals, hand made feature extraction methods that are taking into acount the human psycoaccoustic knowledge have been proposed [44].

The raw audio signals are one-dimensional time series. In practice, it is a common procedure to apply transforms such as Short Term Fourier Transfom (STFT) to represent an audio signal in a two dimensional time-frequency representation [38].

Some of the conventional methods for repesenting audio signal were results of feature extraction methods such as Linear Predictive Cepstral Coefficients (LPCC) [45], Line Spectral Frequencies (LSF) [46] and Mel Frequency Cepstral Coefficients (MFCC) [47]. Methods such as LPCC and LSF are based on autoregressive model of speech production. In essance, these type of methods represents the signal based on autoregressive filter coefficients of given audio production model [48].

As a most used feature extraction method, MFCC represents an audio signal based on the Mel Scale [49]. Mel Scale is a scale that is modelled to represent human auditory system. Time-frequency representation of an audio signal is called a Mel Spectrogram. For Mel Spectrogram representation, a filter bank with overlapping triangular frequency responses that is distributed according to Mel Scale is employed [50]. From the observation of human auditory system it has been shown that the Mel Scale behaves similar to the linear frequency scale below 1kHz whereas the relationship between linear frequency scale and Mel Scale is logarithmic for higher frequencies [51]. In essence, Mel Spectrogram representation of a signal is a frequency warping operation. The relationship between mel scale and linear frequency scale is defined in (3.1).

$$\mathrm{f}_{\mathrm{Mel}} = 2595 log(\frac{\mathrm{f}_{\mathrm{Linear}}}{700} + 1) \tag{3.1}$$

In (3.1) $\mathrm{f}_{\mathrm{Mel}}$ is the Mel frequency with the unit Mel and $\mathrm{f}_{\mathrm{Linear}}$ represents linear frequency in Hz. With equation (3.2), the central frequency of a filter on the Mel scale can be calculated.

$$\mathrm{f}_{\mathrm{cm(Mel)}} = \mathrm{f}_{\mathrm{L(Mel)}} + \frac{\mathrm{m}\left(\mathrm{f}_{\mathrm{H(Mel)}} - \mathrm{f}_{\mathrm{L(Mel)}}\right)}{\mathrm{M}+1}, \quad 1 \le \mathrm{m} \le \mathrm{M} \tag{3.2}$$

$\mathrm{f}_{\mathrm{cm(Mel)}}$ represents the central frequency of the filter m in Mels. The lower and upper boundaries of the entire filter bank with M triangular filters are $\mathrm{f}_{\mathrm{L(Mel)}}$ and $\mathrm{f}_{\mathrm{H(Mel)}}$ in Mels. The output of the filter m can be defined as in (3.3).

$$X_{\mathrm{m}} = \sum_{\mathrm{k}=0}^{\frac{\mathrm{N}}{2}-1} |S[\mathrm{k}]|^2 |H_{\mathrm{m}}[\mathrm{k}]|, \quad 1 \le \mathrm{m} \le \mathrm{M} \tag{3.3}$$

In (3.3), X_{m} is the filter output, $|H_{\mathrm{m}}[\mathrm{k}]|$ is the frequency magnitude response of the triangular filter m, and $|S[\mathrm{k}]|$ denotes the N point Fast Fourier Transform (FFT) magnitude spectrum of the windowed audio sample.

In most cases, the filter output is logarithmically compressed as in (3.4) in order to model perceived loudness of given signal intensity. This representation of the audio signal is named as Log-Mel spectrogram of the signal.

$$X_{\mathrm{m,n}(log)} = log\left(X_{\mathrm{m,n}}\right), \quad 1 \leq \mathrm{m} \leq \mathrm{M}, 1 \leq \mathrm{n} \leq \mathrm{N} \tag{3.4}$$

In (3.4), n represents time index and N represents the number of time bins of the given spectrogram. When it comes to Deep Learning, it is preferable to use Log-Mel Spectrograms because of its capability to compress the dynamic range of the output by applying a logarithmic scaling to the Mel Spectrogram. This results in a more detailed spectrogram image, which can help a Deep Learning model to learn more effectively.

The second part of audio classification procedure is called the inference part. For the inference from audio signal representations, traditional machine learning methods such as such as k-Nearest Neighbors (kNN) [52], Support Vector Machines (SVM) [53], Decision Trees [54], Gaussian Mixture Models (GMM) [55] or Hidden Markow Models (HMM) [56].

As stated before, in audio processing, creating a suitable representation of the features and selecting the right classifier have traditionally been seen as two separate issues. However, this approach can lead to suboptimal feature design for classification purposes. Deep Neural Networks (DNN), on the other hand, combine feature extraction and objective optimization like classification into a single process. Additionally, the availability of large amounts of data and advancements in hardware technology that allow for parallel processing have resulted in a surge of interest in data-driven approaches that utilize deeper methods that combine feature extraction and inference models to achieve better performance and capabilities [57].

Current dominant procedures for audio signal classifaction combines Mel Spectrogram or Log-Mel Spectrogram representation of data with Deep Learning models [58].

As stated before, audio classification task can be categorised based on the audio sample type. In the context of speech, Automatic Speech Recognition (ASR), which can be defined as converting speech samples into words, covers a substaintial area of research. Together with, speaker recognition [59], and speech translation [60], ASR researches are importan for human-computer communication. Beginning with deep Feed Forward Networks (FFN) [61], since the surge of Deep Learning, Convolutional Neural Networks (CNN) [62], Recurrent Neural Networks (RNN) [63] that uses Long-Short Term Memory Models (LSTM) [64] and their combinations have been widely used for speech based tasks [65]. Additionally relatively new network models such as

Listen, Attend and Spell (LAS) models have achieved impressive results in speech recognition tasks [66]. The LAS model is a type of neural network that consists of three parts: an encoder, an attention module, and a decoder [38]. The encoder is similar to a traditional acoustic model, while the attention module is responsible for aligning the input speech signal with the output text. Finally, the decoder is similar to the language model in a conventional speech recognition system.

Due to significant advancements in speech recognition technology, it has become reliable enough to be used in real-world applications. Virtual assistants such as Google Home, Amazon Alexa, and Microsoft Cortana use voice as the primary way of interacting with users [67]. Transcriptions of speech are also being utilized in a variety of applications, including retrieving information from multimedia sources such as YouTube captions [68].

The range of tasks related to Music Information Retrieval (MIR) includes detecting the beginning and end of musical segments [69], estimating the fundamental frequency [70], analyzing rhythm, such as identifying the beat or tempo [71], detecting the musical key [72] and estimating chords [73]. These analysis task are identified as low level analysis problems. Higher-level analysis in MIR includes detecting instruments, separating sounds, transcribing, segmenting structures, identifying artists, classifying genres or moods, and comparing musical features such as identifying repeated themes, recognizing covers, estimating similarity between pieces, or aligning scores [38].

Similarly to speech related tasks, FFNs,CNNs, RNN and their combination have found their use in musical tasks. While CNNs and FNNs have widely considered for tasks like genre or artist classification, RNN based models have been used tasks like onset, tempo or musical key detection, where the importance of temporal dependency between sample frames increases [41].

Another important category is based on environmental sounds. In this dissertation, the performance analysis of proposed methods are provided based on ESC task.

3.1 Environmental Sound Classifation

The Environmental Sound Classifiaction problem is increasingly interested field by the researches, due to its several fields of application. Applications on fields such as accoustic event detection, context aware inteligent system design and multilable classifier applications related to naming the environmental sound [42].

Additionally to the conventional machine learning methods, like all inteligent system designs, Deep Learning approaches have begun gaining traction. It has been shown that, especially CNN based approaches are well-suited for the task of ESC [74]. One of the reasons of this suitability is that CNN are very capable of detecting energy modulation patterns across time and frequency from spectrogram-like inputs [75]. Another reason is the capability of using 2D convolutional filters. Because the CNN models offer filtering with learnable parameters and varied receptive fields, it becomes increasingly possible to for network to network to learn and recognize spectro-temporal patterns that represent different sound classes, even when part of the sound is masked by other sources in time or frequency [74].

Altough it has been a dynamic research area, ESC problem is a less established field, relatively to speech and music related problems [38]. To effectively learn a complex relationship between input and output and perform well on new data, DNN with large number of learnable parameters require a significant amount of training data. This is because they rely on learning a nonlinear function and need enough examples to generalize and accurately classify new data.

One of the most used datasets for benchmarking on ESC problem is the UrbanSound8k dataset [76]. The UrbanSound8k dataset comprises 8732 urban sounds that have been labeled across 10 different classes. The creators of the dataset recommend using 10-fold cross validation when working with the data.

Several different methods have been used to achieve varying levels of accuracy when working with the UrbanSound8k dataset. Conventional machine learning algorithms, such as Random Forest Classifier have been shown to achieve a baseline accuracy of 68% [77]. The dataset's creators have proposed a CNN that utilizes standard audio data augmentation techniques, which resulted in 79% accuracy [74]. Piczak proposed a CNN architecture that achieved an accuracy of 73.7% [78]. Additionally, a dilated CNN approach was used and produced an accuracy of 78% [79]. A recent study with a CNN model, data augmentation and network regularization claimed to have 95.37% accuracy on Log-Mel Spectrogram features, which left data preparation procedure of experiment vogue [80]. As a result, reproduction of the experiment is not achievable.

As stated before, CNN applications need high number of data. As a result of having relatively scarce environmental data, to improve and explore CNN performance, data augmentation became a widely applied technique in the ESC research field [81].

3.2 Data Augmentation

Modern devices with enhanced capabilities have led to a surge interest in data-driven approaches. Among various machine learning approaches, Deep Learning models have become the preferred choice for many data-driven signal processing applications. These models exhibit improved performance with larger amounts of data [82]. To increase the amount of data for data-driven approaches, researchers have proposed various methods for signal processing tasks based on images, audio, or text, which are collectively known as data augmentation methods [83].

Data augmentation methods were developed to generate additional training data [84]. By utilizing augmented data, the feature space is increased while preserving the labels. Consequently, a classifier is expected to perform better with less overfitting to the training data and yield improved evaluation results [85]. For instance, speech recognition researches have employed artificial data produced using data augmentation methods [86, 87].

Data augmentation methods can be applied to audio waveforms in the time domain or their corresponding spectrograms. The most commonly used methods include noise adding, time stretching, time shifting and pitch shifting [87]. In addition, some literature proposes generating new data by warping the linear frequency scale when obtaining the spectrogram representation of the audio data [88]. To perform noise adding, Gaussian noise is multiplied with a randomly selected noise factor and added to the original signal [89]. Time stretching involves changing the tempo and length of the original audio data while maintaining the pitch using a randomly selected stretching factor [90]. Pitch shifting, which is widely used, involves shifting the pitch by a randomly selected factor. However, it should be used with caution in MIR applications, as changing the pitch can alter the expected frequency characteristics of a musical instrument and result in a change of the label of the original data [91]. Time shifting is another method that either delays or advances an audio waveform [92]. This method aims to create a model that is independent of the signal's start or end.

Recent data augmentation techniques diverge from conventional methods and involve using neural network-based approaches. For instance, CNN are utilized as data augmentation tools for speech data [93]. Another approach involves using Generative Adversarial Networks (GAN), where the generator part is used to produce synthetic data to augment the original dataset [94]. In ESC, a Deep Convolutional GAN based augmentation method is employed, and a classifier consisting of a RNN and a CNN achieves state-of-the-art results [95]

In many deep learning applications involving audio signals, the Log-Mel Spectrogram transformation of an audio sample is treated as an image input for neural networks. Therefore, data augmentation methods developed for image signals are also applied to audio applications. For instance, the SpecAugment method uses sparse image warping, frequency masking, and time masking to modify the Log-Mel Spectrogram representation of an audio sample [88].

As computer vision-based deep learning applications have been a popular research topic, a wide range of data augmentation methods have been developed for image signals. These methods include flipping, rotating, or cropping the image, as well as applying color jittering and edge enhancement methods [83]. For instance, Sobel operator-based edge enhancement has been shown to be a successful data augmentation method for CNN based image classification models [85]. A 2D spectrogam representation of an audio can be perceived as similar to an image. This aspect enables to use data augmentation methods of image processing on audio dependent problems.

3.2.1 Vocal Tract Length Perturbation

The Vocal Tract Length Perturbation (VTLP) method is a commonly used data augmentation technique for speech recognition. VTLP is an implementation of Vocal Tract Length Normalization (VTLN) approach as a data augmentation method. One of the factors that creates the speaker variability is the vocal tract length. In order to achieve consistent speech recognition results, it is important to eliminate speaker variability. One method to achieve this is by warping the linear frequency axis of a spectrogram using a warp factor that is calculated based on the audio sample's statistics. This process is called VTLN. When using the VTLP method as a data augmentation technique, instead of calculating a warp factor based on the statistics of an audio sample, a randomly selected value is used to warp the linear frequency scale of the audio sample [96]. VTLP has applications not only in speaker recognition tasks but also in animal audio classification [97] and environmental sound classification task [98]. Applying different wights of frequency bands can be seen as similar to a procedure such as VTLP. This aproach demonstrated promisisng results in accoustic event detection problem [99]. VTLP approach can be described as in equation.

$$f_{VTLP} = \begin{cases} f_{Linear}\alpha_{VTLP}; & f_{Linear} \leq f_{H(VTLP)} \dfrac{min(\alpha_{VTLP}, 1)}{\alpha_{VTLP}} \\ \dfrac{f_S}{2} - \dfrac{\dfrac{f_S}{2} - f_{H(VTLP)} \min(\alpha_{VTLP}, 1)}{\dfrac{f_S}{2} - f_{H(VTLP)} \dfrac{\min(\alpha_{VTLP}, 1)}{\alpha_{VTLP}}}; & \text{otherwise} \end{cases} \tag{3.5}$$

In this equation f_S is the sampling frequency and $f_{H(VTLP)}$ is a boundary frequency that covers the significant formants. It is generally chosen as 4800 Hz. This function defines a warped linear frquency scale. An arbitrary frequency is mapped to a new frequency f_{VTLP} . In data augmentation applications warp factor α_{VTLP} is chosen in the range of 0.9 and 1.1. Figure 3.1 visualises warped scales according to VTLP approach.

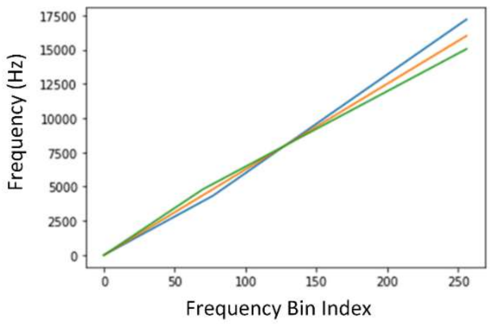


Figure 3.1 : VTLP: Blue- α_{VTLP}=0.9, Orange- α_{VTLP}=1 (linear scale), green- α_{VTLP}=1.1.

In this dissertation we propose FOC based data augmentation methods for audio signals. The first two approaches employ fractional differentiation of linear frequency scale and Mel Scale. These approaches are based on representing the audio signal on a warped time-frequency scale as in VTLP. By using randomly selected fractional derivation order in one, the linear scale and in second, the Mel Scale is warped, therefore, the proposed methods aim to augment linear or Mel Scale based time-frequency representations of audio data. The second approach is based on previous fractional order image edge enhancement methods. Since multiple Deep Learning approaches are treating Mel Spectrogram of any time-frequency representations like images, a fractional order differential based mask is employed, similar to the image enhancement applications. The mask parameters are produced with respect to randomly selected fractional order derivative parameter. In accordance to many works in literature that use Mel Scale representations as dominant data representation method, Fractional Order Mask approach and Fractional Order Mel Scale approach are applied to the Environmental Sound Classification task.

3.2.2 Fractional Order Mask

Fractional order mask applications for edge detection or image segmentation algorithms are mostly based on GL definition of fractional order derivative. This approach is based on multiplying filter coefficients with all the previous values of the function [9, 100].

$$[D^{\alpha} f(\mathrm{x})]_{\mathrm{GL}} = \mathrm{h}^{-\alpha} \sum_{\mathrm{k}=0}^{\mathrm{N}-1} b_{\mathrm{k}} f(\mathrm{x} - \mathrm{hk}) \tag{3.6}$$

In equation (3.6), b_{k} denotes the filter coefficients and can be calculated as in equation (3.7) with respect to the Pochhammer symbol.

$$b_{\mathrm{k}} = \frac{\alpha(\alpha+1)\ldots(\alpha+\mathrm{k}-1)}{\mathrm{k}!} = \frac{(\alpha)_{\mathrm{k}}}{\mathrm{k}!} \tag{3.7}$$

This equation shows that the filter coefficients are values of Gamma Function.

The Fractional Order Mask is proposed for texture enhancement applications. It is shown that by convolving images with fractional differential masks, complex texture details in smooth areas of image could be enhanced [101]. This approach produces 8 convolution kernels for 8 directions in 2D image signal. For a computer tomography image denoising application with CNN, a single kernel which is obtained by adding all the kernels for 8 directions, achieves satisfactory results [102]. The fundamental difference of applied data augmentation method is that the data augmentation is not concerned about finding an optimal derivation order α.

In this dissertation, for creating the kernel for convolution, a very conservative approach is applied for selecting α parameters to prevent label changing of the audio sample. Five α parmeters are selected from a range [0,0.1] and a 5x5 convolution kernel (3.8) is produced for each parameter. For a range selection, ranges [0,0.05], [0,0.15], [0,0.2] were also inspected in following way. For a randomly selected fold augmented datasets that are produced by applying mentioned ranges are used as training data for 10 epochs and classification performance is evaluated. Selecting the range [0,0.05] created an overfit for Mel Spectrogram data. Selecting the other ranges created increasingly an underfit problem. Because of this experiment, [0,0.1] range is found suitable.

$$\begin{bmatrix} b_2 & 0 & b_2 & 0 & b_2 \\ 0 & b_1 & b_1 & b_1 & 0 \\ b_2 & b_1 & b_0 & b_1 & b_2 \\ 0 & b_1 & b_1 & b_1 & 0 \\ b_2 & 0 & b_2 & 0 & b_2 \end{bmatrix} \tag{3.8}$$

Randomly selected three kernels are applied to each Log-Mel Spectrogram or Mel spectrogram representation of audio sample to increase dataset size by three times.

3.2.3 Fractional order scaling

Time-frequency representations can be exploited to increase data size in a dataset. Frequency warping on Mel Scale is not a new approach [103]. Warping methods such as VTLP show that, warping the frequency scale and increasing data size increase a deep learning models' accuracy [96]. In this dissertation two methods based on fractional order differentiation is proposed for data augmentation purposes. In this section, both of them are presented but only the Fractional Order Mel Scale approach is employed for the following ESC experiments.

Similarly to other approaches in literatue, in this dissetation, fractional derivation is applied directly to the Mel Scale itself. Applying randomly selected derivation order, Mel scale is randomly warped and data size is increased.

For fractional derivation of linear or Mel Frequency Scale, the RL definition of fractional derivative is used. It must be noted that, similar approach could be applied with other definitions of fractional derivative such as GL derivative. Numerical algorithm for RL definition of fractional derivative at a point j [9, 104] can be given as in equation (3.9).

$$\left[D^{\alpha} f(x_{\mathrm{j}})\right]_{RL} = \mathrm{h}^{-\alpha} \sum_{\mathrm{k}=0}^{\mathrm{j}} A_{\mathrm{k,j}} f(x_{\mathrm{k}}) \tag{3.9}$$

The $A_{\mathrm{k,j}}$ parameters can be calculated as shown in equation (3.10).

$$A_{\mathrm{k,j}} = \frac{1}{\Gamma(2-\alpha)} \begin{cases} (\mathrm{j}-1)^{1-\alpha} - (\mathrm{j}+\alpha-1)\mathrm{k}^{-\alpha}, & \mathrm{k}=0 \\ (\mathrm{j}-\mathrm{k}+1)^{1-\alpha} + (\mathrm{j}-\mathrm{k}-1)^{1-\alpha} - 2(\mathrm{j}-\mathrm{k})^{1-\alpha}, & 1 \le \mathrm{k} \le \mathrm{j}-1 \\ 1, & \mathrm{k}=1 \end{cases} \tag{3.10}$$

A lower triangular matrix can be produced by calculated $A_{\mathrm{k,j}}$ parameters. This matrix **R** can be seen in equation (3.11).

$$\mathbf{R} = \frac{1}{\Gamma(2-\alpha)} \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ A_{0,1} & 1 & 0 & \cdots & 0 \\ A_{0,2} & A_{1,2} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{0,\mathrm{N}} & A_{1,\mathrm{N}} & A_{2,\mathrm{N}} & \cdots & 1 \end{bmatrix} \tag{3.11}$$

This numerical approach can be represented in a matrix multiplication form as in equation (3.12),

$$[D^{\alpha} f(x)]_{RL} = \mathrm{h}^{-\alpha} \mathbf{R} \cdot \mathbf{f} \tag{3.12}$$

where **R** is a matrix consist of $A_{\mathrm{k,j}}$ parameters as shown in equation (3.11) and **f** is a vector that contains N+1 function values of $f(x)$.

In the proposed data augmentation methods for this dissertation, **f** can represent both the linear frequency scale $\mathbf{f}_{\mathrm{Linear}}$ and Mel scale function $\mathbf{f}_{\mathrm{Mel}}$. It is assumed that function values are sampled with respect to frequency bin indexes, therefore, the value of h is taken as 1. To keep the values in psychoaccoustically acceptable range especially for integration, an additional term that employs N and α is used as a multiplier in linear frequency scale maniplation. This additional term is not used for mel scale maniplation.

By taking the information above into consideration, a method named as the Fractional Order Frequency Scale is given in equation (3.13) . This aproach enables to produce a corresponding value for each value on frequency scale.

$$\mathbf{f}_{\mathrm{Linear}}{}' = (\mathrm{N}+1)^{\alpha} D^{\alpha}{}_{RL} \mathbf{f}_{\mathrm{Linear}} \tag{3.13}$$

For $\alpha = -0.1$ and $\alpha = 0.1$ the process can be visualised as in Figure 3.2.

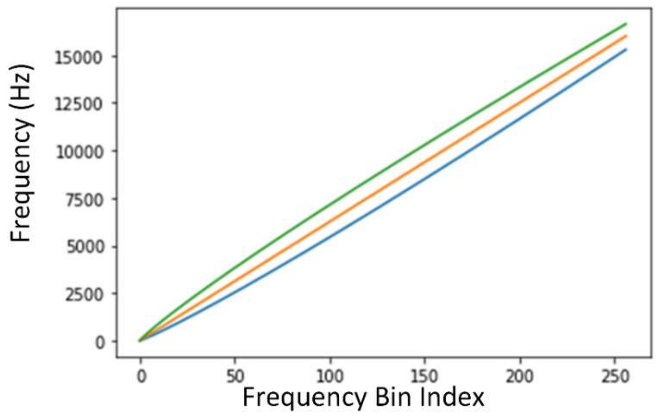


Figure 3.2: Rescaling of frequency scale by deriving with fractional order (Blue: $\alpha = -0.1$, Orange: $\alpha = 0$ (linear scale), Green: $\alpha = 0.1$).

Consequently Fractional Order Mel Scale approach can be defined as in equation (3.14).

$$\mathbf{f}_{\mathrm{Mel}}{}' = D^{\alpha}{}_{RL}\mathbf{f}_{\mathrm{Mel}} \tag{3.14}$$

Similarly to the fractional order mask approach, 5 fractional differintegral operators with derivation order α ranging between [-0.2,0.2] is produced for ESC experiments. The range is chosen as the same way as Fractional Order Mask approach. Other than the selected range, ranges [-0.1,0.1], [-0.3,0.3], [-0.4,0.4] and [-0.5,0.5] also inspected. After inspecting preliminary results, the range for α is chosen as [-0.2,0.2]. Randomly selected 3 orders are applied to each audio sample to obtain 3 fractional Mel Spectrogram representations from every sample.

3.3 Application of FOC Based Data Augmentation Methods on ESC Problem

As stated before, for the experiments, UrbanSound8k dataset which contains 8732 labeled urban sounds from 10 classes is used 10-fold cross validation is applied. Extracted sound samples from the dataset are chosen to be 3s long and padded with zeros if necessary. The applied FFT size for time-frequency transformations is 1024 with 75% overlap and sample rate for all samples is chosen to be 22050 Hz. The Mel Frequency bin size is selected as 128. Beginning from a random starting point 128 time bins are taken into consideration for each sample. The inputs for all experiments are time-frequency representations with the shape of (128,128).

For experiments, offline data augmentation procedure is applied. This means that augmentation procedure is applied before model training. Additionally, to original dataset, augmented datasets with the same size and for each procedure are generated. Furthermore, with each procedure, 3 times increased datasets are produced. For Log-Mel Spectrogram features, Fractional Order Mask procedure is applied after the logarithm operation. The experiments are conducted for both Mel Spectrogram features and Log-Mel Spectrogram features. The dataset sizes for experiments can be seen in Table 3.1.

Table 3.1 : Dataset sizes for each experiment.

Augmentation Procedure	Dataset Size	Feature
Not Augmented (Original)	8732	Mel Spectrogram
Original + Fractional Order Mask	17464	Mel Spectrogram
Original + Fractional Order Mel Scale	17464	Mel Spectrogram
3xFractional Order Mel Scale + 3xFractional Order Mask	52392	Mel Spectrogram
Not Augmented (Original)	8732	Log Mel Spectrogram
Original + Fractional Order Mask	17464	Log Mel Spectrogram
Original + Fractional Order Mel Scale	17464	Log Mel Spectrogram
3xFractional Order Mel Scale + 3xFractional Order Mask	52392	Log Mel Spectrogram

As outlined before, rather than trying to pass benchmark ESC accuracy scores, the focus of this work is to inspect the capabilities of FOC based methods for data augmentation. Therefore, an arbitrary CNN model, similar to the one used in the literature is designed as in Figure 3.3.

The designed network contains two convolutional layers with max pooling and an additional convolutional layer that precedes a global average pooling layer. The global average pooling is used to reduce the needed parameters of dense layer that follows convolutional layers for two reasons. Firstly, the hardware constraints limited the use of larger network models and secondly, in this research case implementing an arbitrary network model does not weaken the point of showing capabilities of fractional order calculus framework.

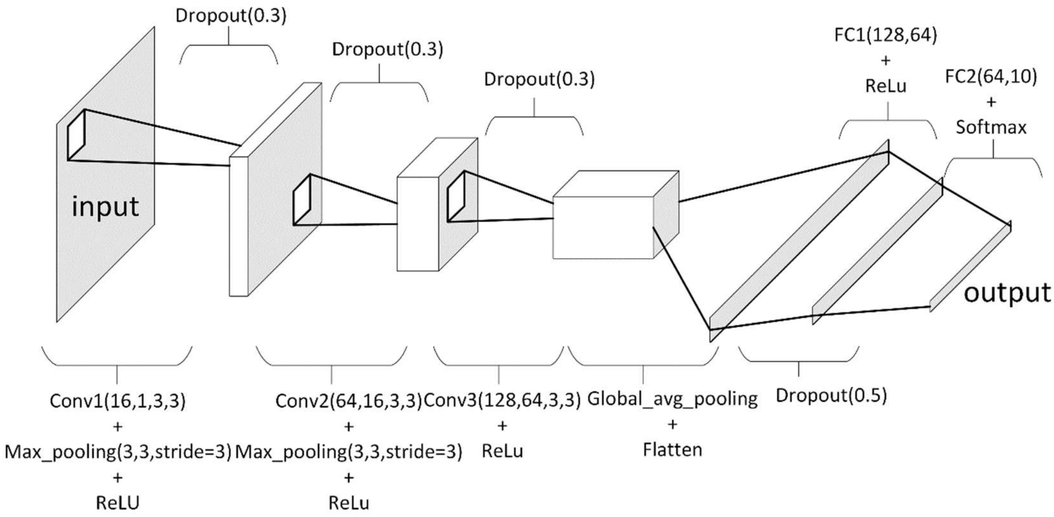


Figure 3.3 : Deep CNN model for ESC with FOC based data augmentation experiments.

Essentially, a DNN is a nonlinear mapping function with learnable weights $\mathbf{W}$. Given an input representation $\mathbf{X}$, the neural network maps input to the output as in (3.15). In classification tasks, the output vector is a vector of class probabilities. In this work, the implemented network has a layer size of 5.

$$Z = F(\mathbf{X}|\mathbf{W}) = f_5(f_4(f_3(f_2(f_1(\mathbf{X}|\mathbf{W}_1)|\mathbf{W}_2)|\mathbf{W}_3)|\mathbf{W}_4)|\mathbf{W}_5) \tag{3.15}$$

First 3 layers are 2D convolutional layers. A convolutional layer can be expressed as in (3.16).

$$Z = f(\mathbf{X}|\mathbf{W}) = h(\mathbf{W} * \mathbf{X} + \mathbf{B}) \tag{3.16}$$

In this expression $h(\cdot)$ is a pointwise activation function. In this implementation the Rectified Linear Unit (ReLu) activation function is used for convolutional layers. ReLu expression can be seen in equation (3.17).

$$h(\mathrm{x}) = max(0, \mathrm{x}) \tag{3.17}$$

Additionally, $*$ represents convolution operation, **W** and **B** are learnable tensors. Bias vector **B** can be excluded in implementations. 2D convolutional layers consists of N input channels and M output channels. Notationally, when **B** is excluded, a 2D convolutional layer can be defined by the shape of tensor **W**, which has the shape of $(\mathrm{M}, \mathrm{N}, \mathrm{m}_0, \mathrm{m}_1)$. $(\mathrm{m}_0, \mathrm{m}_1)$ is the shape of convolutional kernel.

Layers 4 and 5 are fully connected layers. As seen in (3.18), this layer employs a matrix product.

$$Z = f(\mathbf{X}|\mathbf{W}) = h(\mathbf{WX} + \mathbf{B}) \tag{3.18}$$

If **W** is a vector with size N and the output size is M, excluding the bias term **B**, tensor **W** can be represented with the shape of (M, N).

For the experiments conducted for this research, the implemented DNN resembles the network in [74].

2D Convolutional Layer 1: The number of output channels in this layer is 16. The convolutional kernel has the shape of (3,3). The shape of **W** in this layer can be given as (16,1,3,3). This convolutional layer is followed by (3,3) strided max pooling layer and ReLu activation.

2D Convolutional Layer 2: The number of input channels in this layer is 16 and the number of output channel is 64. The convolutional kernel has the shape of (3,3). The shape of **W** in this layer can be given as (64,16,3,3). A (3,3) strided max pooling layer and ReLu activation are applied to the outputs of 2D convolutional layer.

2D Convolutional Layer 3: The number of input channels in this layer is 64 and the number of output channel is 128. The convolutional kernel has the shape of (3,3). The shape of **W** in this layer can be given as (128,64,3,3). ReLu activation is applied to the outputs of 2D convolutional layer. A global average pooling is applied after activation and the outputs are flattened resulting in a vector with size of 128. This operation is applied to further reduce parameter size and training time of our network, due to our hardware constraints.

Fully Connected Layer 1: This layer has 64 output units. The shape of **W** in this layer is (128,64). This layer is followed by ReLu activation.

Fully Connected Layer 2: This layer has 10 output units, which is the same as number of classes. The shape of **W** in this layer is (64,10). This layer is followed by Softmax activation to map layer outputs to class probabilities.

Following each layer Dropout operations with probabilities 0.3 for first 3 layers and 0.5 for fully connected layer 1 are applied. The loss function for model optimization is Cross-entropy Loss. For optimization, ADAM optimizer is employed with learning rate as 0.001 and weight decay parameter as 10^{-3}. Weight decay penalizes squared magnitude of weight values and can be named as L2 regularization. Regularization has been used in deep learning to prevent weight bias and overfitting.

For all experiments, trainings epoch number is selected as 100. While evaluating the classification performance, 10-fold cross validation procedure is applied. UrbanSound8k dataset was published by its creator as divided into 10 folds. In 10-fold cross validation procedure, first, a fold is randomly selected and the network is trained with other folds. The classification accuracy is evaluated on the selected fold. After that, all the network weights and optimizer is initialized again. These steps are repeated until the classification accuracies for all folds are calculated.

The data augmentation method performances are presented with respect to the mean accuracy. In Figures 3.4a and 3.5a, the results for both Mel Spectrogram features and Log-Mel Spectrogram features can be seen.

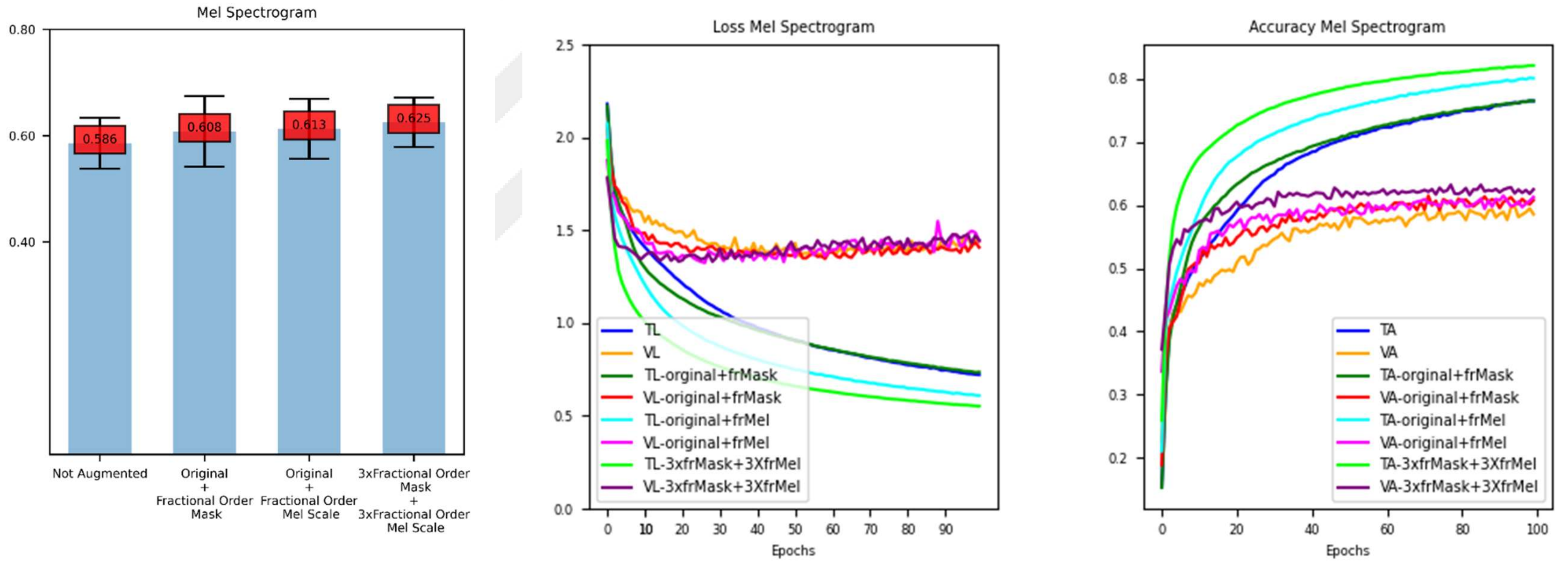
For not augmented Mel Spectrogram features, the accuracy result becomes 58.6%. When we augment the dataset with Fractional Order Mask and add it to the original dataset, we achieve 60.8% accuracy. When the same procedure is conducted for Fractional Order Mel Scale augmentation method, we achieve 61.3% accuracy. When the dataset size is increased by 6 times with both augmentation methods, the mean accuracy for 10-fold cross validation becomes 62.5%.

When the input features are changed to Log-Mel Spectrogram representations, the accuracy result for not augmented case is 64.8%. Augmenting the dataset with Fractional Order Differential Mask and combining the augmented samples with not augmented log Mel spectrogram features results in 65.8% accuracy, whereas, the same procedure for Fractional Order Mel Scale method gives 69.4% accuracy. For the last experiment, the dataset is augmented with both methods separately to increase its size 3 times. After the augmentation, both augmented datasets are combined. This procedure results in 70.3% accuracy. For Log-Mel Spectrogram features, augmenting the dataset 6 times of its original size with fractional order calculus-based data augmentation methods produces 8.48% relative increase in accuracy in the experiments.

In figures 3.4b, 3.5b, 3.4c and 3.5c, the loss and accuracy graphs with respect to the epochs are provided. In figures 3.4b and 3.5b, it is clear that data augmentation reduces the loss both for training and validation data. In figures 3.4c and 3.5c, augmented datasets achieves higher accuracy results both for training and validation. In Figure 3.5c, augmenting the dataset 2 times with Fractional Order Mel Scale augmentation method produces higher training accuracy, on the other hand, augmenting the dataset 6 times with both of the proposed data augmentation methods achieves the highest validation accuracy.

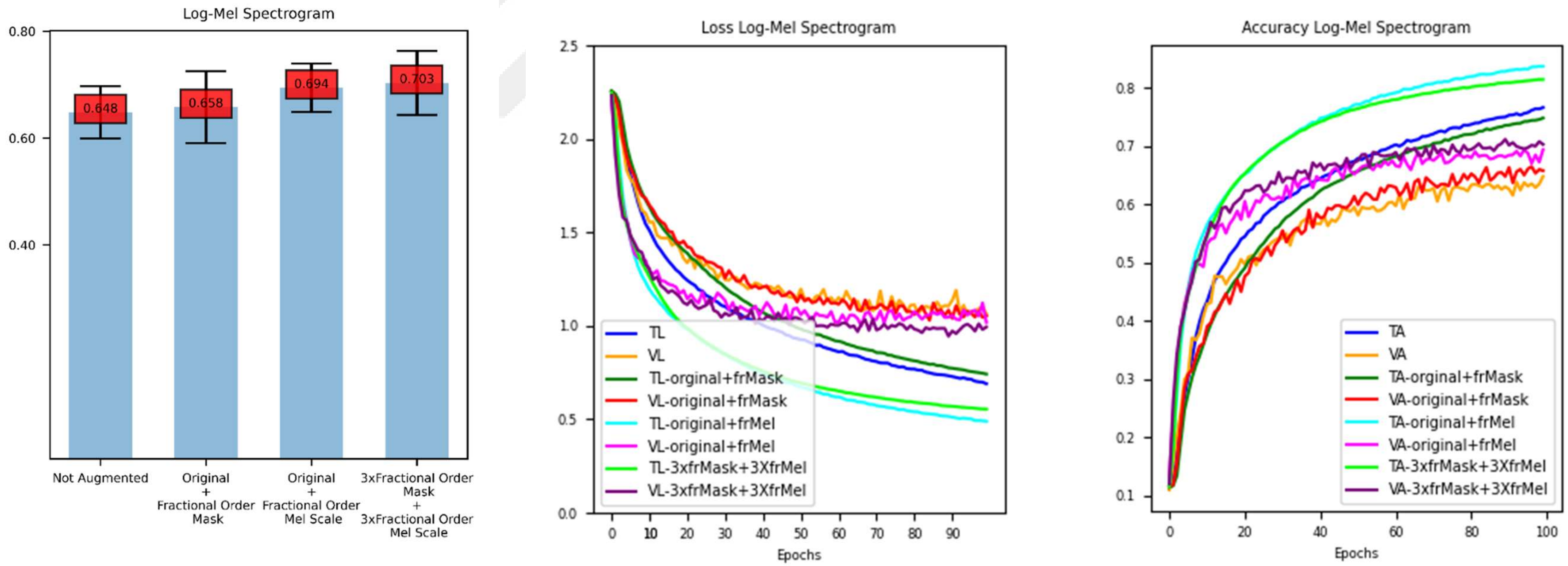
To inspect the results with respect to the increased complexity of network model, output unit size of fully connected layer is increased from 64 to 256. This experiment produce 66.3% accuracy score for not augmented Log-Mel spectrogram features. Repeating the procedure with the fully augmented dataset, results in 71.4% accuracy.

To further understand the performance of data augmentation methods on deeper neural networks an 18-layered ResNet based model is implemented. The two differences of this implementation are on the input layer and on the output fully connected layer. The original ResNet model is designed to have 3 channel image data as input, that's why a 2D convolutional layer with the weight shape of (3,1,3,3) is added. Original ResNet-18 implementation of Pytorch has 100 output classes. That's why, the last fully connected layer is changed to have 10 output units. This network has a parameter size of nearly 100 times of the test network used in the experiments before.



(a) **(b)** **(c)**

Figure 3.4 : Accuracy scores and accuracy and loss graphs for experiments with Mel Spectrograms (TL: Training Loss, VL: Validation Loss, TA: Training Accuracy, VA: Validation Accuracy): (a)Accuracy scores for Mel spectrogram features; (b)Loss curve for Mel Spectrogram features; (c)Accuracy curve for Mel spectrogram features.



(a) **(b)** **(c)**

Figure 3.5 : Accuracy scores and accuracy and loss graphs for experiments with log Mel Spectrograms (TL: Training Loss, VL: Validation Loss, TA: Training Accuracy, VA: Validation Accuracy): (a)Accuracy scores for log Mel spectrogram features; (b)Loss curve for log Mel Spectrogram features; (c)Accuracy curve for log Mel spectrogram features.

In Deep Learning, the increased parameter size generally leads a network to overfit. This experiment showed that in terms of loss, the proposed data augmentation methods were unable overcome the overfitting characteristic of a deeper network like ResNet18 [105]. In Figure 3.6, the data augmentation performance for Log-Mel Spectrogram features of not augmented and 6 times augmented datasets on ResNet presented with respect to the mean accuracy.

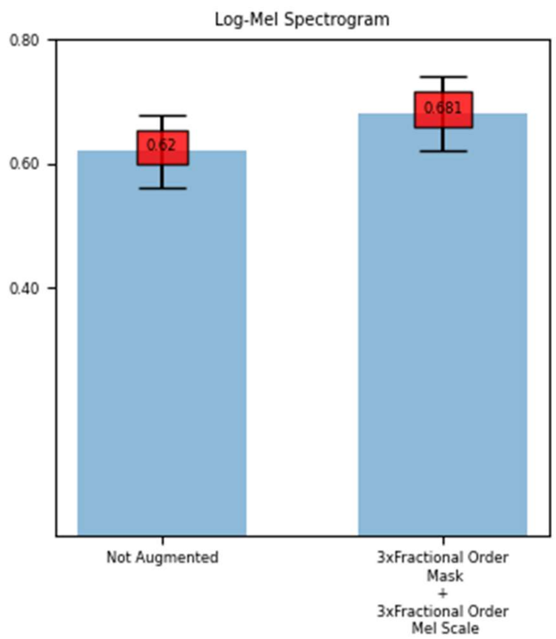


Figure 3.6 : Accuracy scores for ResNet-18 experiment.

The accuracy result for tIn terms of accuracy, augmented dataset achieved nearly 10% of relative increase in terms of validation accuracy after 25 epochs of training. The not augmented dataset resulted in 62% validation accuracy, on the other hand, after the dataset augmented 6 times using fractional order calculus based data augmentation methods, the resulted accuracy becomes 68.1%.

3.4 Comments on Experiments

The FOC provides a capable framework for progressively increasing number of application domains. In this dissertation, it is claimed that FOC based approaches can be successfully applied yet another domain. According to related literature, there haven't been an implementation of FOC based methods for audio data augmentation. Implemented Fractional Order Mask and Fractional Order Mel Scale augmentation methods are proven to be beneficial in ESC problem, as their implementations increased accuracy results.

The Fractional Order Mel Scale approach employs fractional differentiation of Mel scale. It extends the group of frequency warping-based methods for data augmentation. The Fractional Order Mask approach is an application of similar image edge enhancement methods on audio data augmentation domain.

Using Log-Mel Spectrogram have been a more preferred way in Deep Learning, because of the logarithmic scaling the Mel Spectrogram output creates a dynamic range compression. This process results in a more detailed spectrogram image for deep learning model to learn. The proposed data augmentation methods work on Log-Mel Spectrogram representation of audio data better. As seen from the experiments, the augmentation procedure for Mel Spectrogram features achieves up to 6,6% relative increase, whereas, for Log-Mel spectrogram features the relative increase rate becomes 8.48%. This is an expected result. The Fractional Order Mel Scale approach results in amplitude deviations with respect to original sample. Due to the conservative approach in the experiments, smaller derivation orders are selected. Without the range compression, the variations on amplitude values of augmented data and original data become less significant, resulting the model to see too similar training data to original data. More in-depth analysis showed that augmenting the dataset multiple times creates smaller gains with respect to accuracy and in some cases lead the model to overfit. Due to similar reasons as Fractional Order Mel Scale method, Fractional Order Mask method creates greater variations from original data when applied to Log-Mel spectrogram features of audio sample.

The aim of this work is not to overcome benchmark accuracy results of proposed deep learning models for ESC. A similar deep CNN model as in the literature is chosen, but implemented with further global average pooling operation before fully connected layers. Because of the constraints of available hardware, a network with smaller number of learnable parameters is preferred. Since, the main objective is to inspect performance of proposed data augmentation methods, choosing to implement a model like this does not weaken the points of this dissertation. Nevertheless, experiments with increased size of output units for first fully connected layer are conducted. To inspect the results with respect to the increased complexity of network model, output unit size of fully connected layer is increased from 64 to 256. This experiment produced 66.3% accuracy score for not augmented Log-Mel Spectrogram features. Repeating the procedure with the fully augmented dataset, results in 71,4% accuracy. The resulting 7.7% relative increase shows that the points with regards of applying FOC based methods are valid. In addition to the test networks, ResNet-18 model is implemented to understand the performance of the proposed methods on deeper networks. In this experiment, the network with fully augmented dataset achieved more than 9% of relative increase in terms of validation accuracy. On the other hand, inspecting loss and accuracy curves showed that network performance with respect to the overfitting characteristics is not substantially improved. This result leads to the point that to achieve substantial improvement in terms of overfitting, current parameter selection procedure for proposed data augmentation methods are not enough and needs improvement. It must be noted that in this dissertation, the aim is proposing and evaluating different data augmentation methods, therefore, the optimal derivation orders for maximizing classification performance were not the main topic, which would be crucially important for novel feature extractor designs. Additionally, recent studies on deep learning approaches employ more complex optimization policies to overcome overfitting. In the cases of present experiments, it is a conscious choice to keep training procedure simpler to be able to see the effects of augmentation methods in a clearer way.

4. AUDIO SYNTHESIS PROBLEM

In a broad definition, audio synthesis can be perceived as a task of generating audio samples from a sequence of conditional variables [38]. The most critical difference of this task with respect to the audio classification problem is that audio synthesis inference procedure does not necessarily begin with the audio samples.

Audio synthesis problem has always attracted the interest of researchers in the area of audio signal processing. The audio synthesis are spans a wide range of topics from designing synthesizer circuitry, generating music and providing helper tools for artists to Text-to-Speech (TTS) applications with various types of vocoding techniques. In the sense of audio type, music and speech based applications provides to main categories for audio synthesis.

The synthesis techniques for music can be loosely grouped into two main branches. Two main branches of synthesis techniques are additive synthesis and subtractive synthesis [106]. Additive synthesis can be defined as addition of wave signals with different frequency characteristics, which are generated with different oscillators [107]. Subtractive synthesis works based on the filtering preferable frequencies of raw audio wave with rich frequency compounds [108].

Additional branches of musical synthesis techniques are Frequency Modulation Synthesis and Physical Modelling techniques. Frequency modulation technique, which employs both audible carrier and modulator signals to create effects for audience [109]. Physical Modelling techniques are covering the mathematical modelling techniques for musical instruments. For most musical instruments, this will be a coupled set of partial differential equations, describing, e.g., the displacement of a string, membrane, bar, or plate, or the motion of the air in a tube, etc. [110].

Music generation is a subtopic of the sound synthesis. By definition, music generation should include a probabilistic approach, a composition aspect. The early models for music generation used Markov Chains [111], while nowadays, it has become a neural network game [112]. With the increased computational power, modelling the probabilistic nature of composition and synthesis with neural networks gave rise to brilliant methods. Usage of RNN to model remember and forget behaviors or GAN for probabilistic nature brought about powerful tools for sound synthesis area [113].

The context of speech synthesis is one of the most interested areas of research due to its effects on various engineering fields such as building interactive engineering products, audiobooks, navigation services, home automation products or providing quality communication tools [114]. The aim of speech synthesis is to produce natural and intelligible waveform from a set of conditional variables.

The most significant application of speech synthesis procedures has been as a part of TTS applications. TTS applications provides frameworks for generating speech waveforms from text inputs [115]. These frameworks can consist a separate two stage. In this type of applications, first stage is tasked with producing intermediate acoustic features from the input text. This stage is called the acoustic model. The second stage, which is called a vocoder, then produces audio waveforms from the given representations [116]. The overview of this approach can be seen in Figure 4.1. Especially after the surge of deep learning, the models that are motivated to combine the two separate part into one have gained traction [117]. Nevertheless, due the hardship in training compute and requirements of vast amount of data keep the model based vocoding researches relevant.

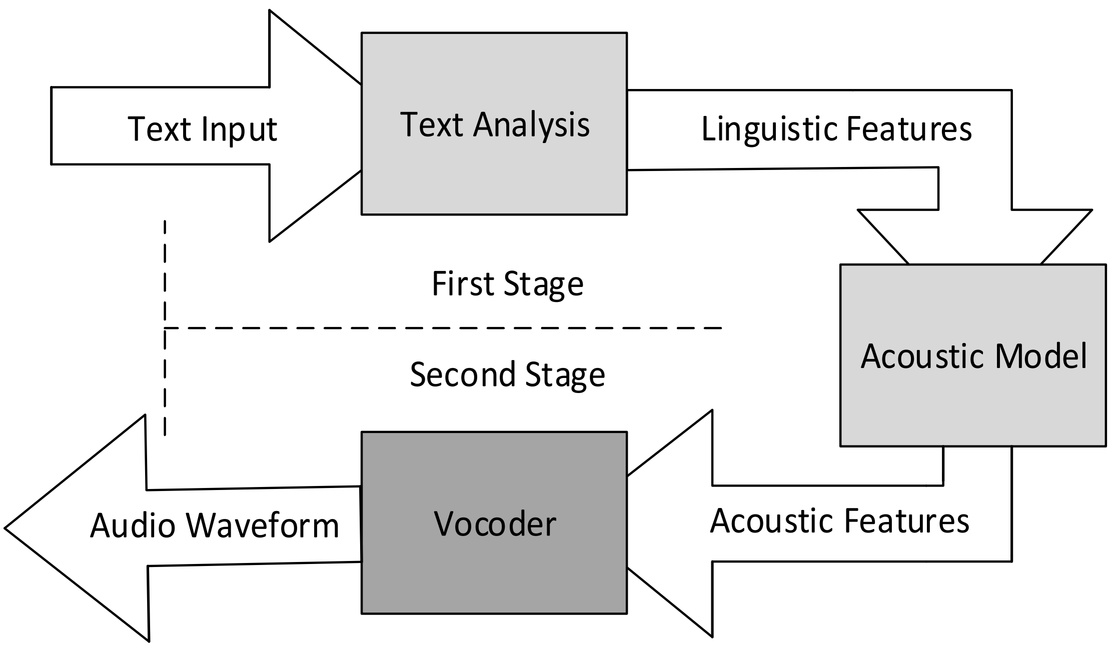


Figure 4.1 : Overview of two stage pipeline of TTS applications.

Vocoder is the short form for "voice coding". The works about vocoders date back to 1930s because of the need of audio communication task [118, 119]. The vocoder types can be categorized into families such as channel vocoder [120], phase vocoders [121] or AR vocoders such as LPC vocoder [122]. In terms of contemporary relevancy, because of their relations in modern approaches, autoregressive and phase vocoders are important.

Subjective naturalness of speech depends on quality of vocoders, accuracy of generative models and effects of over smoothing. Bad quality of vocoder design results in artifacts in synthesized speech.

Extracting vocoder parameters can be viewed as estimation of a generative model parameters given speech. For example, in traditional Linear Prediction Coefficients (LPC) vocoders, linear predictive analysis assumes that the generative model of speech signals is a linear AR process [123].

$$s'[\mathrm{n}] = \sum_{\mathrm{k}=1}^{\mathrm{K}} a_{\mathrm{k}} s[\mathrm{n}-\mathrm{k}] + \varepsilon_n \tag{4.1}$$

In equation (4.1) a_{k} is k^{th} order linear predictor. This approach models human vocal track in terms of LPC. ε_n can be zero mean gaussian process or an impulse train, depending on the unvoiced or voiced sounds in speech. Consequently, it is possible to transmit or compress the speech with reduced number parameters [19]. In RNN based audio synthesis researches, AR model and LP based models provided good synthesis quality and used as benchmark models [124].

A dominant number of audio related applications dictate, the analysis and modification of the Short-Time Fourier Transform (STFT) and the Short-Time Fourier Transform Magnitude (STFTM) audio signals [125]. Audio enhancement, time and pitch modification or reverberation analysis are some of the examples for this procedure. In terms of TTS, transforming acoustic features to time-frequency representations such as STFTM is relatively easier than producing the waveform itself. In such cases, the complex characteristics, the phases of a signal are generally lost.

In the literature, it has been shown that appropriate estimation of phase from the STFTM is possible [126]. This approach is called as audio reconstruction. Methods for audio reconstruction are usually a member phase vocoder family. The phase vocoder method represents a signal as a combination of sine waves, where the key factors that need to be identified through analysis are the changing amplitude and frequency of each individual sine wave over time, which are present in STFTM representation [121]. This approach also provides analysis tools for additive synthesis methods [127].

Additionally, spectral consistency based approaches are used for audio reconstruction. The limited length of signal segments and the form of the spectral analysis window cause dependencies between the spectral coefficients of neighboring frequency bands, known as spectral leakage which affects both spectral amplitudes and phases [128]. Griffin and Lim's approach exploits these relationships by iteratively estimating spectral phases based on spectral amplitudes of a speech signal [129]. In this method, the STFT and its inverse are computed repeatedly while keeping the spectral amplitude fixed and only updating the phase.

The STFTM based phase reconstruction and iterative methods can be used together to improve audio reconstruction performance [130]. In this dissertation, experimental results for such cases for the proposed phase estimation method is presented.

Since consistency based approaches play an important for this dissertation, following a summarized background information about them is provided.

4.1 Phase Reconstruction

Signal reconstruction from STFTM spectra is a topic that never loses its relevance. This subject, also referred to as the phase recovery problem or spectrogram inversion problem, has attracted the attention of various re-searchers, and important studies have been carried out. The most used approach for solving this problem is the well-known Griffin–Lim Algorithm (GLA), which was first introduced by Griffin and Lim as an iterative repetition of inverse STFT (ISTFT) and STFT by considering initial phase conditions and exploiting the property of spectrogram redundancy [125].

Because GLA and its derivatives are iterative algorithms, they are time-consuming [129, 131]. Therefore, in areas where application speed is a concern, different algorithms have been proposed such as Single-Pass Spectrogram Inversion (SPSI). The SPSI not only outputs applicable results, but also provides a better initial phase estimate for iterative methods such as GLA [132]. Recently, various methods are proposed for noniterative signal reconstruction problems that claimed improved results concerning SPSI [126, 133]. Moreover, with the emergence of deep learning, neural models have produced state-of-the-art results [124, 134-136]. As a common rule, recent methods provided results in comparison with SPSI, thus solidifying its place as a benchmark [137].

Historically, phase recovery methods have been model-based approaches. Speech and musical signals, which are defined as semi-stationary signals, have been represented with sinusoidal models. A sinusoidal model takes a speech signal of a given frame length, the time length in which the audio signal is accepted as stationary, as a weighted sum of sinusoids with different phases, frequencies, and amplitudes [107]. The sinusoidal model has been used in both analysis and synthesis applications of audio signal processing. Phase vocoder applications are used for audio signal synthesis for audio processing applications [121]. During synthesis, a given encoded audio signal or a feature set of an audio signal is used to reconstruct the signal in the time domain. Generally, STFT magnitudes are analyzed to define phase and frequency components of sinusoids.

For example, the SPSI algorithm first detects spectral peaks and applies quadratic interpolation techniques around the neighborhood of peak bins to obtain an estimate for instantaneous frequency. First, the magnitude of the STFT of the signal is calculated; then, a quadratic function that passes through an amplitude point at a peak bin and amplitude points in its two neighborhoods is fitted [138]. Next, with the Phase-Locked Vocoder approach, signal phases are obtained. A linear interpolation gives the phase estimate, which is called the phase accumulator. In the frequency domain, if a peak has happened, a simple estimate for phases of its adjacent bins is a 180° shift. If the peak is not in the center but between two peak bins, it is assumed that one of the adjacent bins also influenced the phase. Therefore, a phase alternation strategy dependent on the location of the instantaneous frequency estimate was proposed [139].

The following sub-chapter intends to provide a deeper understanding about spectral consistency based signal reconstruction.

4.1.1 Spectral consistency based signal reconstruction

The STFT is a commonly used technique for analyzing signals in the time-frequency domain. However, it has some inherent limitations due to the windowing process involved, which can result in redundant information in the spectrogram.

One issue with the STFT is spectral redundancy, which means that the coefficients obtained from the STFT may not necessarily form a valid spectrogram [129]. Spectral redundancy occurs because the windowing process in the STFT introduces overlap between adjacent windows, causing redundant information in the frequency domain. This redundancy can lead to multiple possible signals having the same STFT coefficients, making it impossible to uniquely reconstruct the original signal from the STFT alone.

In fact, there may be no signal that exactly matches a given spectrogram obtained from the STFT. This is because the STFT is a representation of the signal in the time-frequency domain, and the inverse process of reconstructing the original signal from the STFT magnitude is ill-posed and may not have a unique solution.

The problem of reconstructing a signal can be formulated as constructing a signal from a non-valid STFT magnitude. The problem can be expressed as finding a signal $\mathbf{x}^* \in R^{\mathrm{L}}$ from a given set of non-negative coefficients $\mathbf{S}$, such that the magnitude of the STFT of $\mathbf{x}^*$, $|G\mathbf{x}|$, is as close as possible to $\mathbf{S}$ [140]. L represents the dimension of the space.

As a measure of closeness the L2-norm provides a sufficient measure. The mathematical formulation for the problem in terms of optimization can be described as a minimization problem. Since *G* is a frame dependent Gabor based transform and $\mathbf{S} = |\,\mathbf{S}\,|$ is the real positive coefficients, the problem of finding a signal $\mathbf{x}^*$ that has the valid spectrogram can be defined as in the following form [129].

$$minimize_{\mathbf{x}\in R^L} \| |G\mathbf{x}| - \mathbf{S} \|_2 \tag{4.2}$$

The problem can be translated as that $\mathbf{S}$ is a valid STFT magnitude if there exists an $\mathbf{x}$ such that $|G\mathbf{x}| = \mathbf{S}$.

In convenience to optimization problem deifinitions, the problem can be defined with optimization variable on the coefficient side.

$$minimize_{\mathbf{C}\in C^{\mathrm{M}\times\mathrm{N}}} \| |\mathbf{C}| - \mathbf{S} \|_2 \; s.t. \quad \exists \mathbf{x} \in R^L | \mathbf{C} = \mathrm{G}\mathbf{x} \tag{4.3}$$

The measure of error for this problem is given in the following form.

$$E(\mathbf{x}) = \frac{\| |G\mathbf{x}| - \mathbf{S} \|_2}{\|\mathbf{S}\|_2} \tag{4.4}$$

This error measure can be represented in the form of Spectral Signal to Noise Ratio (SSNR) form.

$$SSNR(\mathbf{x}) = -10 \, log_{10}(\mathrm{E}(\mathbf{x})) \tag{4.5}$$

Another representation of this measure is the Spectral Covergence (SC) [131].

$$E(\mathbf{x}) = \frac{\|\mathbf{S} - |G\mathbf{x}|\|_2}{\|\mathbf{S}\|_2} \tag{4.6}$$

The SC is one of the most used objective speech qualitiy metrics.

The GLA is a double projection algorithm. It employs iterative projections of signal on set C_1, which is the set of the admissable points of optimization problem (4.3) and set C_2, which is the set of coefficients minimizing optimization problem (4.3). C_1 and C_2 constraints are sets that are in $C^{\mathrm{M \times N}}$ [129]. Here, M corresponds to the number of frequency channels and N corresponds to the number of time indexes.

Since C_1 is the set of admissible points for problem, it is a hard constraint. It corresponds to the set of coefficients $\mathbf{C}$ that can be reached from the solution $\mathbf{x}^* \in R^{\mathrm{L}}$ by applying transform G.

$$C_1 = \{\mathbf{C} | \, \exists \mathbf{x} \in R^L \; s.t. \, \mathbf{C} = G\mathbf{x} \, \} \tag{4.7}$$

The constraint C_1 forces the solution to satisfy the consistency criterion. The projection can be defined as two transforms, ISTFT and STFT [130]. In equation (4.8), STFT is denoted as G and its pseudo inverse ISTFT is denoted as $G^{\dagger}$.

$$\mathrm{P}_{C_1}(\mathbf{C}) = GG^{\dagger}\mathbf{C} \tag{4.8}$$

Constraint C_2 can be defined as the set of coefficients minimizing (4.3). Since C_1 is the hard constraint of the problem, C_2 is the soft constraint that can be defined as below.

$$C_2 = \{\mathbf{C} \in C^{M\,N} | \, |\mathbf{C}| = \mathbf{S}\} \tag{4.9}$$

This constraint forces non-negative coefficients $\mathbf{S}$ to be equivalent to the coefficients $\mathbf{C}$ that are in the set C_1. This soft constraint can be met with following projection onto C_2.

$$\mathrm{P}_{C_2}(\mathbf{C}) = \mathbf{S} \cdot e^{\cdot j \angle \mathbf{C}} \tag{4.10}$$

The GLA can now be formulated as shown in the Algorithm 4.1.

Algorithm 4.1 Griffin-Lim Algorithm [129]

Fix the initial phase $\angle \mathbf{C}_0$
Initialize $\mathbf{C}_0 = \mathbf{S} \cdot e^{\cdot j \angle \mathbf{C}}$
Iterate for n=1,2,… **do**

$$\mathbf{C}_\mathrm{n} = \mathrm{P}_{C_1}\left(\mathrm{P}_{C_2}(\mathbf{C}_{\mathrm{n}-1})\right)$$

Until convergence

$$\mathbf{x}^* = G^\dagger \mathbf{C}_\mathrm{n}$$

In Fast Griffin-Lim Algorithm (FGLA) an extra step for updating GLA is proposed for faster convergence. In this approach the difference between $\mathbf{C}_\mathrm{n}$and $\mathbf{C}_{\mathrm{n}-1}$ is defined as $\Delta\mathbf{C}_\mathrm{n}$. Additionally a constant parameter p that is generally selected between 0 and 1 is added for tuning. Naming the intermediate step output as $\mathbf{T}_\mathrm{n}$ the FGLA can be defined as Algorithm 4.2.

Algorithm 4.2 Fast Griffin-Lim Algorithm [129]

Fix the initial phase $\angle \mathbf{C}_0$
Initialize $\mathbf{C}_0 = \mathbf{S} \cdot e^{\cdot j \angle \mathbf{C}}, \mathbf{T}_0 = \mathrm{P}_{C_2}\left(\mathrm{P}_{C_1}(\mathbf{C}_0)\right)$
Iterate for n=1,2,…**do**

$$\mathbf{T}_\mathrm{n} = \mathrm{P}_{C_1}\left(\mathrm{P}_{C_2}(\mathbf{C}_{\mathrm{n}-1})\right)$$
$$\mathbf{C}_\mathrm{n} = \mathbf{T}_\mathrm{n} + p(\mathbf{T}_\mathrm{n} - \mathbf{T}_{\mathrm{n}-1})$$

Until convergence

$$\mathbf{x}^* = G^\dagger \mathbf{C}_\mathrm{n}$$

It must be noted that, for $p = 0$ the FGLA corresponds to the ordinarry GLA.

In the literature faster algorithms for phase recovery with GLA based methods are proposed [131]. One of the proposed methods employs Alternating Direction Method of Multipliers (ADMM) [141]. The intuition behind this method is based on the interpretation of an optimization problem as in (4.11)

$$min_{\mathbf{X}} f(\mathbf{X}) + g(\mathbf{X}) \tag{4.11}$$

As defined constraint sets C_1 and C_2, the Griffin-Lim based phase recovery problem can be defined with respect to the consistency constraint t_{C_1} that satisfies the solution on set C_1 and t_{C_2} that satisfies the solution on set C_2.

$$min_{\mathbf{X}} t_{C_2}(\mathbf{X}) + t_{C_1}(\mathbf{X}) \tag{4.12}$$

The solution of this optimization can be approximated using ADMM with employing proximal operators. Proximal operators are used for solving convex optimization problems [142]. The solution of the optimization problem defined above is calculated by an iterative procedure as shown in (4.13).

$$\begin{gathered}\mathbf{C}_{\mathrm{n+1}} = prox_{\frac{f}{\rho}}(\mathbf{D}_{\mathrm{n}} - \mathbf{K}_{\mathrm{n}}) \\ \mathbf{D}_{\mathrm{n+1}} = prox_{\frac{g}{\rho}}(\mathbf{C}_{\mathrm{n+1}} + \mathbf{K}_{\mathrm{n}}) \\ \mathbf{K}_{\mathrm{n+1}} = \mathbf{K}_{\mathrm{n}} + \mathbf{C}_{\mathrm{n+1}} - \mathbf{D}_{\mathrm{n+1}}\end{gathered} \tag{4.13}$$

In this procedure f corresponds to the constraint t_{C_2} and g corresponds to the constraint t_{C_1}. Additionally, $\mathbf{D}_{\mathrm{n}}$ and $\mathbf{K}_{\mathrm{n}}$ are auxiliary variables for faster convergence in ADMM [131]. The definition of proximity operator for an arbitrary variable **Y** on a function f can be defined as below as in [131]:

$$prox_{\frac{f}{\rho}}(\mathbf{Y}) = argmin_{\mathbf{X}}\, f(\mathbf{X}) + \frac{\rho}{2}\|\mathbf{Y} - \mathbf{X}\|_{Fro}^2 \tag{4.14}$$

The parameter ρ is related to step size of the mapped values of proximal operation. The smaller is the value of ρ, the higher the movement towards the minimum [142]. Setting only t_{C_2} as hard constraint the optimization problem reduces to following form:

$$min_{\mathbf{X}}\, t_{C_2}(\mathbf{X}) + \frac{1}{2}\left\|\boldsymbol{X} - \mathrm{P}_{C_1}(\mathbf{X})\right\|_{Fro}^2 \tag{4.15}$$

As it can be seen, t_{C_1}constraint is relaxed to satisfy minimization of squared distance. In terms of proximity operator $prox_{\frac{\mathrm{d}^2{}_{C_1}}{2\rho}}(\mathbf{Y})$, the iterative algorithm can be rewritten.

Here, $\mathrm{d}^2{}_{C_1}$ corresponds the frobenius norm on set C_1 and 2ρ coresponds to step size of the proximal operation. As stated in [131], proximity operator of squared distance has the following explicit form of (4.16).

$$prox_{\frac{d^2 c_1}{2\rho}}(\mathbf{Y}) = \frac{1}{1+\rho}(\rho\mathbf{Y} + P_{c_1}(\mathbf{Y})) \tag{4.16}$$

After plugging auxiliary variables into the relevant places the ADMM based GLA with relaxed consistency can be defined as Algorithm 4.3.

Algorithm 4.3 Griffin-Lim ADMM Algorithm (GLA-ADMM) [131]

Fix the initial phase $\angle\mathbf{C}_0$ and $\rho > 0$

Initialize $\mathbf{C}_0 = \mathbf{S} \cdot e^{\cdot j\angle\mathbf{C}}, \mathbf{D}_o = \mathbf{C}_0, \mathbf{K}_0 = 0$

Iterate for n=1,2,…**do**

$$\mathbf{C}_n = P_{c_2}(\mathbf{D}_{n-1} + \mathbf{K}_{n-1})$$

$$\mathbf{D}_n = prox_{\frac{d^2 c_1}{2\rho}}(\mathbf{C}_n + \mathbf{K}_{n-1})$$

$$\mathbf{K}_n = \mathbf{K}_{n-1} + \mathbf{C}_n - \mathbf{D}_n$$

Until convergence

$$\mathbf{x}^* = G^{\dagger}\mathbf{C}_n$$

4.1.2 FOC based instantaneous frequency estimation for audio reconstruction

In time-series applications or fractal theory, the use of differential equations is valid and has many applications. In time-series analysis, an ARFIMA (0, d, 0) process with unit increments of time index t can be generalized as a fractional differential equation [143].

$$\frac{d^{\alpha}}{dt^{\alpha}} f(t) = w(t), t \in [0, T-1] \tag{4.17}$$

In equation (4.17), the $f(t)$ signal is a one-dimensional vector with a length of T. Most audio processing applications deal with normalized and sampled signals with respect to specified sample rate. To avoid dependence on the sample rate and reaching physically incorrect models, in this proposed approach, $f(t)$ is taken as a dimensionless signal, and t as a dimensionless variable of time [12]. The Power Spectral Density Function (PSDF) of $f(t)$ is given by $|F(\omega)|^2$, and $w(t)$ is white noise. The PSD of white noise is a constant *c*; hence, the following equation

$$|j\omega|^{2\alpha}|F(\omega)|^2 = c. \tag{4.18}$$

The PSD of a function, modeled as a FDE, can be estimated as

$$\hat{P}_{\mathrm{k}} = P(\omega_{\mathrm{k}}) = \frac{c}{{\omega_{\mathrm{k}}}^{\beta}}, \mathrm{k} = \left[1,2, \dots, \frac{\mathrm{NFFT}}{2}\right], \tag{4.19}$$

where $\beta = 2\alpha$ and NFFT is the size of FFT window applied to the signal frame.

From Equation (4.19), it can be seen that, depending on the value of $\beta,$ the system exhibits different characteristics in terms of long-range dependence and frequency response.

If $\beta < 0$, the system in equation (4.19) becomes a differentiator. The spectrogram of the system will be dominated by high-frequency components as the differentiator behaves like a high-pass filter. Consequently, the system will not exhibit long-range dependence.

If $\beta > 0$, the system in Equation (3) becomes an integrator. The spectrogram of the system will be dominated by low-frequency components as the integrator behaves as a low-pass filter. Consequently, the system will have long-range dependence. As a result, estimating the value of β should provide the a spectrograms or a part of the spectrograms behaviour.

The estimation of differential equation order has been investigated in time-series analyses. One of the most extensively used solutions is a linear regression model, as proposed by Geweke and Porter-Hudak [144]. The least-squares error method can be applied to obtain estimates for β and C and create a regression line.

$$e = \sum_{\mathrm{k}} \left(ln\, P_{\mathrm{k}} - ln\, \widehat{P_{\mathrm{k}}}\right)^2 = \sum_{\mathrm{k}} [ln\, \mathrm{P}_{\mathrm{k}} - (C - \beta\, ln\, \omega_{\mathrm{k}})]^2, \tag{4.20}$$

where $C = lnc$.

This method is simple to implement. To evalute the result, only the spectrogram magnitudes are needed, which is the same as the most common case for signal reconstruction. If this method is applied to appropriately normalized values, the significant workload of least-squares estimation can be reduced.

To find the values of C and β, which minimize this error function, the equations $\frac{\partial e}{\partial \beta} = 0$ and $\frac{\partial e}{\partial C} = 0$ should be calculated. Solving these two equations yields the following expressions for β and C:

$$\beta = \frac{N\sum_{\mathrm{k}}(ln\,P_{\mathrm{k}})(ln\,\omega_{\mathrm{k}}) - (\sum_{\mathrm{k}} ln\,\omega_{\mathrm{k}})(\sum_{\mathrm{k}} ln\,P_{\mathrm{k}})}{(\sum_{\mathrm{k}} ln\,\omega_{\mathrm{k}})^2 - N\sum_{\mathrm{k}}(ln\,\omega_{\mathrm{k}})^2}, \tag{4.21}$$

$$C = \frac{1}{N}\sum_{\mathrm{k}}(ln\,P_{\mathrm{k}}) + \frac{\beta}{N}\sum_{\mathrm{k}} ln\,\omega_{\mathrm{k}}. \tag{4.22}$$

Using equations (4.21) and (4.22), following estimation can be given.

$$ln\,P_{\mathrm{k}} = -\beta\,ln\,\omega_{\mathrm{k}} + C. \tag{4.23}$$

This approach can be used to estimate instantaneous frequency with the modifications described below. Figure 4.2 shows a diagram of the proposed approach.

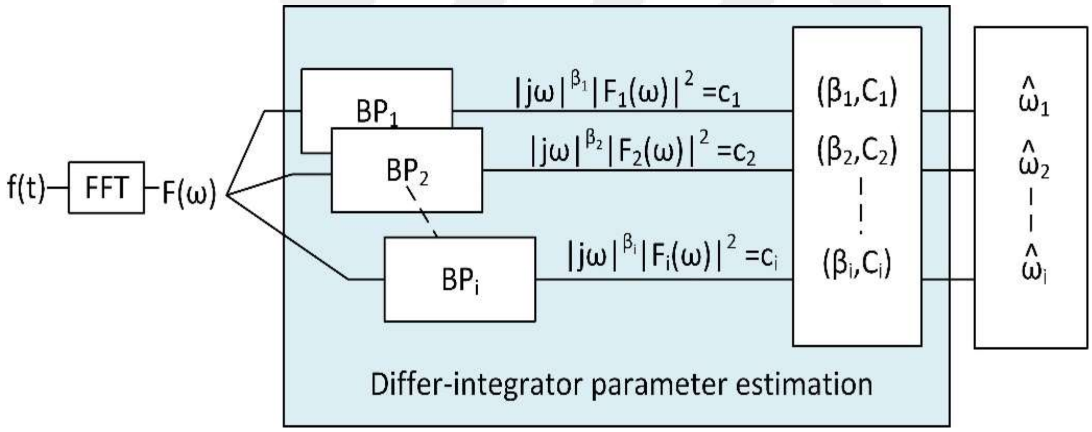


Figure 4.2 : Proposed instantaneous frequency estimation with FDE around spectrogram peak frequencies.

First of all, spectral peaks and their two adjacent points in the spectrogram are taken into consideration; then, for every peak bin index j these data points are modelled according to the differential equation in equation (4.18) and obtain their PSDFs as

$$|j\omega|^{2\alpha}|F_{\mathrm{i}}(\omega)|^{\,2} = c_{\mathrm{i}}. \tag{4.24}$$

Equation (4.24) models PSDF of i^{th} band pass filter output that corresponds to a peak bin index j of a signal frame m. Accordingly, an estimate for new P_{k} using Equations (4.19) and (4.23) can be created as

$$P_{\mathrm{k}} = \omega_{\mathrm{k}}{}^{-\beta_{\mathrm{i}}} c_{\mathrm{i}} \text{ and } ln\, P_{\mathrm{k}} = -\beta_{\mathrm{i}}\, ln\, \omega_{\mathrm{k}} + C_{\mathrm{i}}. \tag{4.25}$$

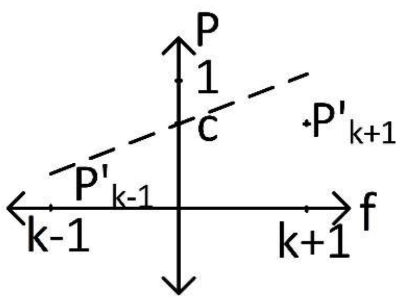
Then, through calculating the β and C values a regression line that passes through the peak and its adjacent points can be produced. The algorithm for this procedure is given in Algorithm 4.4.

Algorithm 4.4. Pseudocode for FDE-based instantaneous frequency estimation on a signal frame.

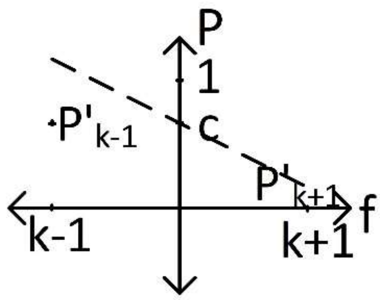
Input: Spectrogram magnitude of the signal frame $|F(\omega)|$, NFFT
Output: $\widehat{\omega}_{\mathrm{j}}$, *sign*($\beta_{\mathrm{i}}$)
Calculate logarithmic power spectrogram $ln\, P(\omega)$
Assign 0.5, 1, 2,3 to $\omega_{\mathrm{k}-1}$, ω_{k}, $\omega_{\mathrm{k}+1}$, and N
$\omega_1{}' \leftarrow ln\omega_{\mathrm{k}-1}$
$\omega_2' \leftarrow 0$
$\omega_3' \leftarrow ln\omega_{\mathrm{k}+1}$
$\mathrm{denominator1} \leftarrow \omega_1' + \omega_3'$
$\mathrm{denominator} \leftarrow (\omega_1' + \omega_3')^2 - \mathrm{N}\left(\omega_1'^{\,2} + \omega_3'^{\,2}\right)$
for every spectral peak bin j **in** $ln\, P(\omega)$ **do:**
    $P_{\max} \leftarrow max(lnP_{\mathrm{k}-1}, lnP_{\mathrm{k}}, lnP_{\mathrm{k}+1})$
    $P_{\min} \leftarrow min(lnP_{\mathrm{k}-1}, lnP_{\mathrm{k}}, lnP_{\mathrm{k}+1})$
    if $lnP_{\mathrm{k}-1} < ln\, P_{\mathrm{k}+1}$ **then:**
        $P'_1 \leftarrow (ln\, P_{\mathrm{k}-1} - P_{\min})/(P_{\max} - P_{\min})$
        $P'_3 \leftarrow 0$
        $P'_2 \leftarrow 1$
        $\mathrm{nominator1} \leftarrow 1 + P'_1$
        $\beta_{\mathrm{i}} \leftarrow [(P'_1\omega_1') - \mathrm{nominator1} \cdot \mathrm{denominator1}]/\mathrm{denominator}$
        $C_{\mathrm{i}} \leftarrow (\mathrm{nominator1}/\mathrm{N}) + \beta_{\mathrm{i}} \cdot \mathrm{denominator1}/\mathrm{N}$
    else if $lnP_{\mathrm{k}-1} > ln\, P_{\mathrm{k}+1}$ **then:**
        $P'_1 \leftarrow 0$
        $P'_3 \leftarrow (ln\, P_{\mathrm{k}+1} - P_{\min})/(P_{\max} - P_{\min})$
        $P'_2 \leftarrow 1$
        $\mathrm{nominator1} \leftarrow 1 + P'_3$
        $\beta_{\mathrm{i}} \leftarrow [(P'_3\omega_3') - \mathrm{nominator1} \cdot \mathrm{denominator1}]/\mathrm{denominator}$
        $C_{\mathrm{i}} \leftarrow (\mathrm{nominator1}/N) + \beta_{\mathrm{i}} \cdot \mathrm{denominator1}/\mathrm{N}$
    else:
        $P'_1 \leftarrow 0$
        $P'_3 \leftarrow 0$
        $P'_2 \leftarrow 1$
        $\mathrm{nominator1} \leftarrow 1$
        $\beta_{\mathrm{i}} \leftarrow 0$
        $C_{\mathrm{i}} \leftarrow 1/3$
    endif
$\widehat{\omega}_{\mathrm{j}} \leftarrow 2\pi(\mathrm{j} + C_{\mathrm{i}}\beta_{\mathrm{i}})/\mathrm{NFFT}$
return $\widehat{\omega}_{\mathrm{j}}$, *sign*($\beta_{\mathrm{i}}$)
endfor

As seen in proposed algorithm two modifications to the regression algorithm is applied. Firstly, $ln\,P_{\mathrm{k}}$ values of 3 points are normalized between 0 and 1; secondly, constant values for $\omega_{\mathrm{k-1}}$, ω_{k}, and $\omega_{\mathrm{k+1}}$ are taken as 0.5, 1, and 2. These two normalization processes provide getting linearized frequency indices around the origin and more distributed data points. Moreover, because of having normalized frequency values, it is expected that the fractional differentiation order corresponds to the slope of the tangent line. Furthermore, by normalizing frequency values, the equation is forced to be in coherence with the theory in [144], which employs a least-squares estimator of the slope parameter in linear regression, formed using only the lowest-frequency ordinates of the Log-Periodogram. Lastly, these predefined values help reduce computational costs.

After the calculation of the regression line, two cases are present. Figures 2a,b show these two cases denoted as case a and case b. In case a, $\beta_{\mathrm{i}} < 0$, and, in case b, $\beta_{\mathrm{i}} > 0$. The sign of the β_{i} value shows whether the frequency for the actual peak is on the left- or right-hand side of the plane. If $\beta_{\mathrm{i}} < 0$, then it can be percieved as a system that did not exhibit long-range dependence; therefore, the spectrogram of the system will be populated by a high-frequency component and the regression line will be an increasing line that indicates instantaneous frequency is on the right-hand side of the plane. If $\beta_{\mathrm{i}} > 0$, it can be concurred that the system exhibits long-range dependence, and, being an integrator, its spectrogram will be populated by low-frequency components. Similarly to case a, it can be evaluated that the instantaneous frequency is on the left-hand side of the plane because it creates a decreasing line.



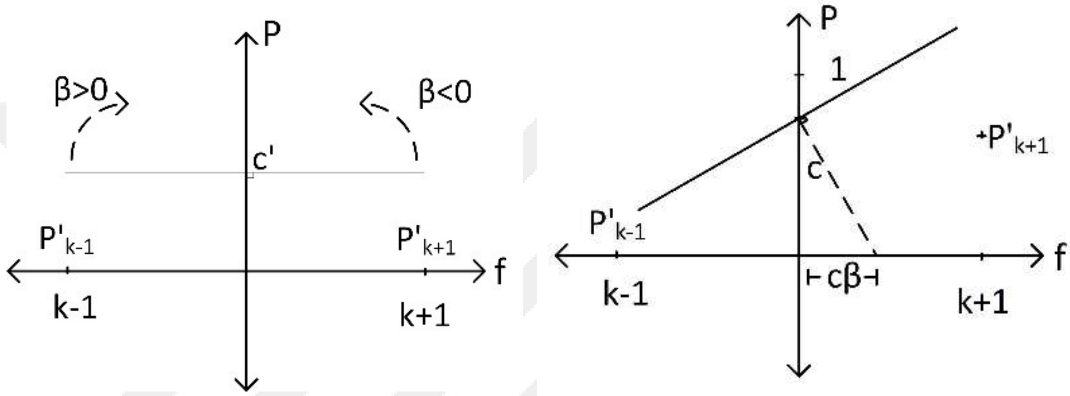
(a)



(b)

Figure 4.3 : Regression lines around a peak with respect to beta: (a)Case a: $\boldsymbol{\beta < 0}$; (b)Case b: $\boldsymbol{\beta > 0}$.

To estimate instantaneous frequency, the line that is normal to the regression line at the point $P'_{\mathrm{k}}(0, C)$ is calculated. The frequency estimate of the model is the value that the normal line hits the x-axis in the frequency domain. This assumption is based on the case of a peak having equally valued adjacent points. In that case, after normalization of $P'_{\mathrm{k-1}}$ and $P'_{\mathrm{k+1}}$, values are equal to 0 and $|P'_{\mathrm{k-1}} - P'_{\mathrm{k}}| = |P'_{\mathrm{k+1}} - P'_{\mathrm{k}}|$; therefore, there is no increasing or decreasing trend. As seen in Figure 4.4a, the regression line becomes orthogonal to the magnitude axis, indicating that instantaneous frequency is indeed at index k.



(a) **(b)**

Figure 4.4 : Calculating estimated instantaneous frequency: (a)When β = 0, the normal to the regression line points to the peak frequency bin; (b)Upon changing slope of the regression line, the intersection point of the normal line on the frequency axis moves $|C\beta|$ away.

In Figure 4.4b, if $\beta_{\mathrm{i}} < 0$, the normal line will hit the x-axis at $|C_{\mathrm{i}}\beta_{\mathrm{i}}|$ on the right-hand side of the plane. This indicates a new instantaneous frequency estimate. Instantaneous frequency can be calculated as follows:

$$\widehat{\omega}_{\mathrm{j}} = \frac{2\pi(\mathrm{j} + C_{\mathrm{i}}\beta_{\mathrm{i}})}{\mathrm{NFFT}}, \tag{4.26}$$

where j is the frequency index of the spectral peak.

From now on, the same approaches proposed by Dolson [121] and Puckette [139] can be applied to estimate phase values for frames and frequency bins, similarly to SPSI.

Phase values for frequency bin j and frame m can be given as

$$\varphi_{\mathrm{m,j}} = \varphi_{\mathrm{m-1,j}} + hop\ size \cdot \widehat{\omega}_{\mathrm{j}}. \tag{4.27}$$

From [139], it is known that phase values for adjacent bins of the peak can be estimated as 180° (π)-shifted values. Depending on the sign of β_{i}, the same scenario as used in SPSI is applied because the instantaneous frequency estimate stays halfway between the peak and adjacent frequency bins. Depending on the value, the weight of the adjacent frequency bins on the phase will change. Thus, if $\beta_{\mathrm{i}} < 0$, the neighbor bin on the right will also influence the phase; accordingly, bins on the right will have the phase value of the peak bin, and bins on the left will have the π-shifted value of the peak bin. If $\beta_{\mathrm{i}} > 0$, the complete opposite would apply.

When all the phase values have been computed, the function will return to the time domain by applying inverse FFT with the Hanning window. Then, the frames will be added in an overlapping manner to finally reconstruct the signal. The whole process is summarized in Figure 4.5.

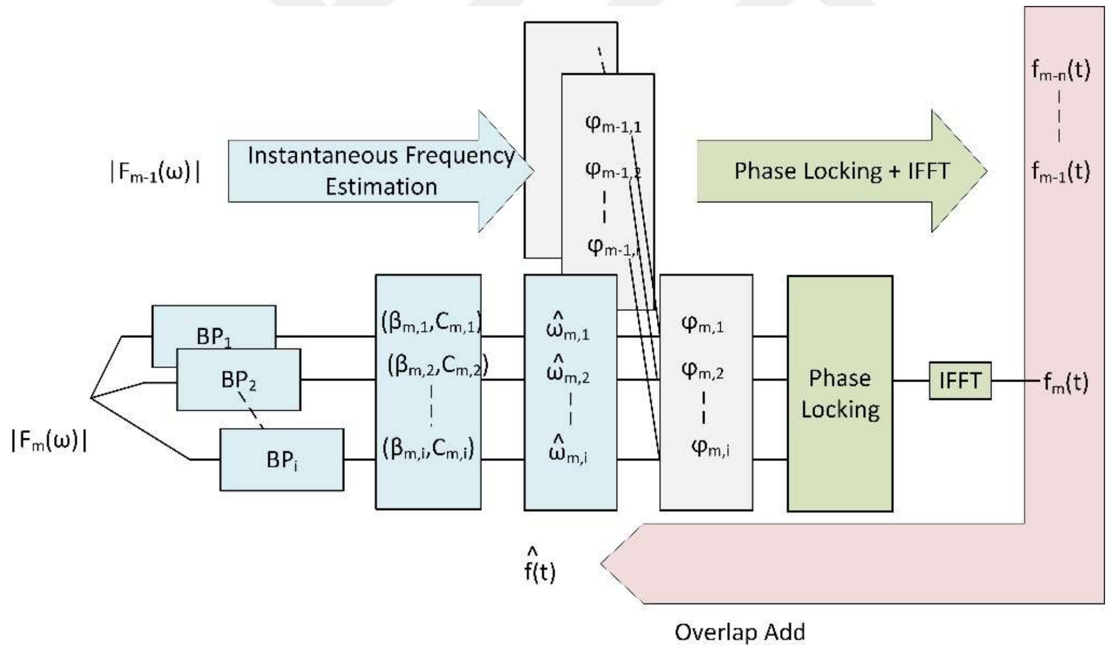


Figure 4.5 : Proposed signal reconstruction flow.

4.1.3 Application results

For the tests, a subset of the Texas Instruments / Massachusetts Institute of Technology (TIMIT) database is used. TIMIT is a read speech corpus, which has been used for benchmarking speech processing implementations [145]. The corpus contains 16 bit, 16 kHz speech samples from various dialects of American English. In this present experiment, 50 male and 50 female speech samples are used.

Reconstruction performance was evaluated by two objective measures: SC [129] and Perceptual Evaluation of Speech Quality (PESQ) [146]. SC is commonly used as a loss function or an objective measure and can be given as with plugging STFT projection into relevant position.

$$SC = \frac{\| |STFT(\mathbf{x})| - \mathbf{S} \|_{Fro}^2}{\|\mathbf{S}\|_{Fro}^2}. \tag{4.28}$$

In Equation (4.28), $\mathbf{S}$ is the target magnitude spectrogram, $\mathbf{x}$ is the signal, and $\|\cdot\|_{Fro}$ denotes the Frobenius norm. This measure is generally used in the logarithmic form $10log(SC)$. It is widely stated that spectral convergence is not a highly correlated measure with human perception. The PESQ measure was proposed by the International Telecommunication Union Telecommunication Standardization Sector (ITU-T) for providing a highly correlated measure with subjective evaluation metrics. PESQ employs auditory transform that reflects human auditory perceptions, thereby producing highly correlated results with subjective evaluation methodologies such as Mean Opinion Dcore (MOS) and Multiple Stimuli with Hidden Reference and Anchor (MUSHRA) for 16kHz sampled data.

As stated throughout the text, a well-known property of the STFT spectrogram is that it is a redundant representation because it is computed by overlapping windowed short-term frames of a signal. Spectrogram consistency is an integral part of signal reconstruction algorithms such as Griffin–Lim. In this regard, overlap value is important because it affects spectrogram resolution and consistency. An increased hop factor which means increased number of windowing also increases aliasing. In the tests, 512 sample length Hamming windows with three different hop sizes of 64, 128, and 256 were applied to the signals. The FDE based method is compared to SPSI and random phase.

4.1.3.1 Comparison of FDE based method to SPSI with respect to the hop size

Figure 4.6 shows the SC and PESQ boxplots for comparison of the FDE-based method and SPSI with hop sizes of 64, 128, and 256. Table 1 gives comparisons of FDE, SPSI, and random phase concerning the average results of SC and PESQ.

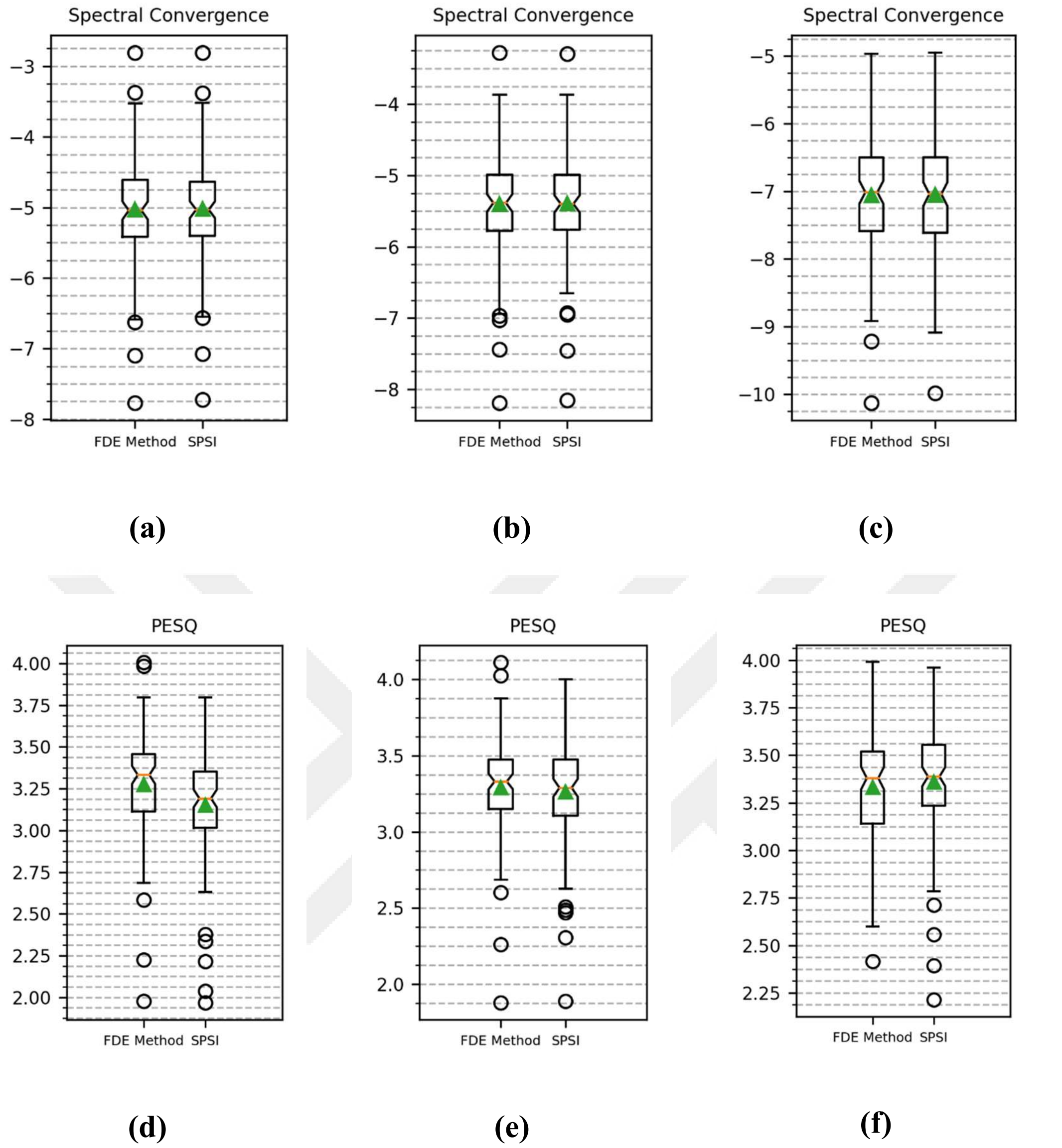


Figure 4.6 : SC and PESQ boxplots for different hop sizes: (a)SC boxplot of FDE method and SPSI for hop size = 64, overlap = 87.5%; (b)SC boxplot of FDE method and SPSI for hop size = 128, overlap = 75%; (c)SC boxplot of FDE method and SPSI for hop size = 256, overlap = 50%; (d)PESQ boxplot of FDE method and SPSI for hop size = 64, overlap = 87.5%; (e)PESQ boxplot of FDE method and SPSI for hop size = 128, overlap = 75%; (f)PESQ boxplot of FDE method and SPSI for hop size = 256, overlap = 50%.

Table 4.1 and Figure 4.6 show that the FDE based method and SPSI achieved nearly identical SC results, with the FDE based method having the slightest upper hand. In Table 1, SC performance has a greater improvement when using the FDE based method upon increasing hop size. Because of the small differences, it can be concurred that the proposed FDE based method and SPSI behaves similarly in the sense of spectrogram convergence, concerning decreasing overlap value and resolution. Inspecting the value distribution in Figure 4.6a–c validates this result.

Table 4.1 : Mean SC and PESQ measurement results of SPSI, FDE method, and random phase concerning increasing hop size.

Hop Size	Method	SC	PESQ
64 (87.5% overlap)	SPSI	−5.02086	3.150896
	FDE Method	−5.02798	3.27584
	Random	−1.08779	1.644972
128 (75% overlap)	SPSI	−5.39316	3.261765
	FDE Method	−5.39769	3.289174
	Random	−1.55655	1.627104
256 (50% overlap)	SPSI	−7.05101	3.360622
	FDE Method	−7.06164	3.331025
	Random	−2.30093	1.579338

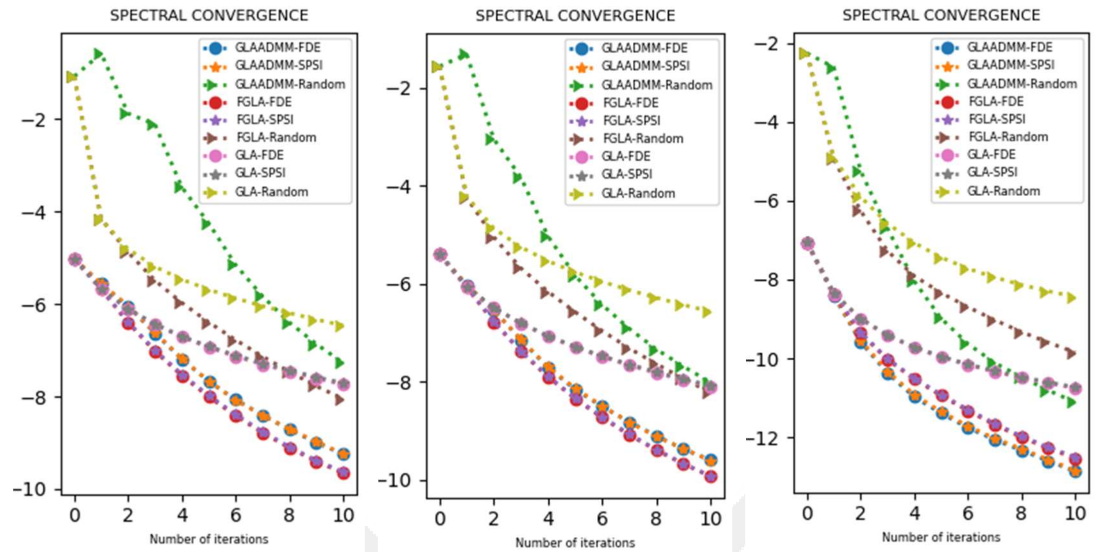
In terms of the PESQ score for decreasing overlap, the FDE based method produces better results for all different overlap values, but in the marginal case of 50% overlap. When the hop size was 64, the FDE based method achieves a 4% increase in PESQ score concerning SPSI. From Figure 5d,e, it can be seen that the PESQ value deviation is also smaller in the cases with 87.5% and 75% overlap. It can be said that, for higher-resolution cases, which result in fewer aliases, the FDE based method constructs perceptually better signals.

4.1.3.2 Comparison of FDE based method and SPSI as initial values to GLA based methods

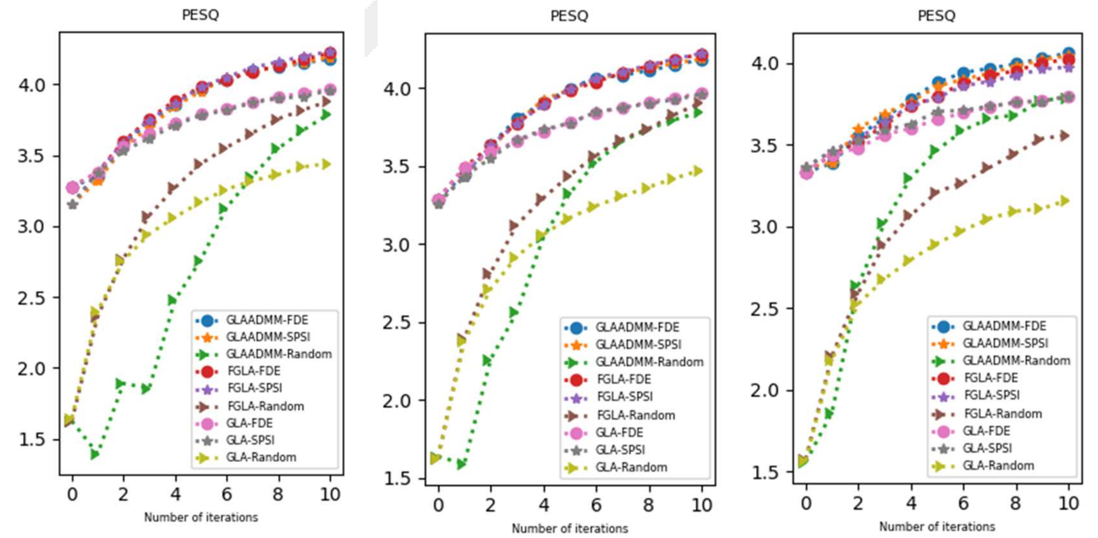
The FDE based method, SPSI, and random phase are applied to three GLA based iterative reconstruction methods as initial phase values and compared the deviation of SC and PESQ metrics with iterations.

In all cases, the FDE based method produces similar, if not slightly better, results to benchmark SPSI in terms of SC. Because the difference was substantially small, it can be claimed that the FDE based method and SPSI would produce similar SC results for high resolution cases when the two constraints of GLA iteration are set as hard constraints.

In Figure 4.7a and 4.7d, it can be seen that the initial results for the FDE based method is higher than both SPSI and random phase initialization. Beginning from the first iteration, the PESQ results for FDE method based initialization and SPSI initialization coincide. Over the iterations, all iterative methods produce similar SC and PESQ results with FDE and SPSI initialization.



(a) **(b)** **(c)**



(d) **(e)** **(f)**

Figure 4.7 : GLA, FGLA, and GLA-ADMM comparison with SPSI, FDE method, and random phase values as initial phases: (a)SC for 10 iterations for hop size = 64, overlap = 87.5%; (b)SC for 10 iterations for hop size = 128, overlap = 75%; (c)SC for 10 iterations for hop size = 256, overlap = 50%; (d)PESQ for 10 iterations for hop size = 64, overlap = 87.5%; (e)PESQ for 10 iterations for hop size = 128, overlap = 75%; (f)PESQ for 10 iterations for hop size = 256, overlap = 50%.

Figure 4.7b and 4.7e show similar results for increased hop size. Upon decreasing the overlap to 50%, it can be seen that the GLA-ADMM method's performance increases because it was tailored to perform better under relaxed the consistency. Figure 4.7c shows that, for SC evaluation, the GLA-ADMM method results in smaller values in the earlier part of the iteration. The small difference concerning the initialization method indicates the consistency of results with smaller hop sizes. Expectedly, the GLA-ADMM method also performes well in terms of the PESQ score. Figure 4.7d, shows that the evaluation curves for the FDE-based method and SPSI follow a similar trajectory with results of smaller hop sizes. Random phase initialization creates a distinction for the 50% overlap case. Because of the nature of the ADMM method, GLA-ADMM exploits random values and creates a speedy increase. All in all, it can be stated that the FDE based method and SPSI can both increase GLA based iterative reconstruction performance with similar results and produce better results than random value initialization.

4.1.4 Comments on the experiments

Although the inherent complexity of FOC and the apparent self-sufficiency of integer order calculus has led to a relative under-exploration of applications of the fractional order framework to signal processing, it has been shown that many systems in science and engineering can be modeled more accurately by fractional order than integer order derivatives, and many such methods have been developed to solve the problem of fractional systems. Moreover, the theory of fractals and fractional order time-series modeling has found some applications for various sound synthesis problems. As a subtopic of sound synthesis, the audio reconstruction problem can benefit from the non-integer order differintegration model due to its ability to model a signal memory, i.e., the dependence of a signal sample on previous samples.

The experiment presented in this study provides valuable insights into the application of fractional-order calculus in audio reconstruction, specifically in the context of GLA based iterative methods. The results show that the proposed FDE based method performs comparably, if not better, than benchmark methods such as SPSI in terms of SC and PESQ metrics. The experiment also demonstrates the impact of overlap and hop size on the performance of the methods, indicating that higher-resolution cases with fewer aliases result in better perceptual quality.

In this experiment the fractional order calculus framework is applied to the audio reconstruction problem. This approach is based on conventional vocoder topologies. Unlike conventional vocoders, in this work, bandpass filter outputs around peak frequencies are modeled as fractional-order differential equations. By applying this model, the memory feature of fractionally integrated models are exploited and linked to the instantaneous frequency estimation. The results of the experiment is evaluated using two measures. SC is one of the most used measures for similar works, whereas PESQ is a more correlated measure with human auditory perception. The produced results are listed concerning the window overlap using these measures. By doing so, the method performance is evaluated regarding spectrogram resolution and consistency. By applying the fractional order framework to the phase reconstruction problem, it is shown that a method based on a fractional-order differential equation model can achieve better PESQ scores than the benchmark SPSI method in high-resolution conditions, along with similar spectral convergence values. Furthermore, using SPSI and the proposed FDE based method as initialization tools for three different GLA based methods, it is shown that the proposed phase reconstruction method produce similar results to benchmark SPSI on iterative algorithms.

Using FDE based approach to model the frequency component of the phase gradient achieves up to a 4% increase in signal reconstruction. It can be expected that modeling the time component of the phase gradient with FDE is also possible and that this approach can further increase evaluation performance, which can be addressed in future researches. Moreover, in this framework, it is possible to increase the adjacent point numbers for differ-integrator value estimation for the frequency component of the phase gradient and evaluate its effect on phase reconstruction performance.

5. NEURAL AUDIO SYNTHESIS

Neural Audio Synthesis (NAS) is one of the most interested research areas in contemporary audio signal processing community. Since the beginning of the Deep Learning era, published literature for generating audio waveforms with Deep Learning models rapidly increased. This phenomenon is due to the published successful results of neural audio synthesis models, in terms of providing quality audio waveforms [38].

5.1 Elementary Concepts of NAS

As stated in the preceding chapters of this dissertation, audio synthesis, or in particular, text to speech synthesis, deal with the problem designing two stages; acoustic model that models text input to auditory features and vocoder that provides audio waveform from the acoustic features.

Neural vocoder is the name of the concept that provides audio waveforms from the acoustic features with designing DNN based models [116]. A DNN for vocoding can employ various types of neural network approaches. For example, CNN [124], RNN [134], GAN [147] and Transformer [148] based approaches have all implemented for speech or other audio signal synthesis tasks.

Providing best synthesis quality for NAS models has been and will always be an important aspect of the research. But the need of applied models to end devices with limited resources gave way to the family of applications that are focused on creating smaller neural models for vocoding [134]. Because of the high number of parameters for training, the most successful NAS models exhausts the limited hardware resources of various implementation areas. Therefore, implementing conventional DSP approaches together with neural networks [135], incorporating neural models into consistency-based audio reconstruction methods and providing flexible models for training and inference [136] in the sense of end devices became relevant topics.

It is widely considered that WaveNet architecture is the benchmark generative architecture [149]. The WaveNet takes a raw audio sample as input. It models the joint probability of a waveform as a product of conditional probabilities [124]. In practical applications the Wavenet structures generate signal with conditioning on acoustic features such as Mel Spectrograms [150].

The WaveNet employs CNN architecture. Defining causal convolutions to keep output causal and dilate a mask to reduce computational cost, they eliminate the need for recurrent connections and create a faster architecture than RNN based approaches as a result of parallelism provided by CNN architecture [151]. The dilated convolution approach also provides an increased receptive field.

With applying different optimization techniques, a team in DeepMind succeeded achieving the first sequential neural model, capable of achieving real time audio synthesis on the hardware like mobile phone CPUs [134]. The WaveRNN achieves competitive results with WaveNet due to recurrence implemented a data-driven approach to audio synthesis. In a simpler sense, WaveRNN is consisted of two neural networks that one essentially models the most significant half of 16-bit sampled speech and the other models the least significant half [134].

GAN approaches are recent development in neural network models. A GAN is basically composed of two networks. One is a discriminator and other is generator. Generator network generates fake outputs to discriminator network. Discriminator network takes real and fake data and tries to decide the weather its fake or not. GANs are formidable in trying to express a known data in the sense of features. This straightforward sampling facility can make it applicable to the audio synthesis problems. Donahue et. al implemented two models as WaveGAN and SpecGAN, the first implementation of GANs in audio synthesis [147]. In their models the name describes the input type. As the WaveGAN takes waveform as input while the input for SpecGAN is spectrogram of data.

In this dissertation, the end-to-end approaches for audio synthesis is intentionally excluded. The successful implementations of such models are well documented in the literature. The underlying idea in this research has always been to focus on the neural synthesis models that employ reduced number parameters due to the hardware constraint of the present research.

AR vocoding approaches occupy a substantial area in contemporary research. The inherent idea of these methods is easy to understand, as these approaches are based on the AR source-filter modelling of speech. As a result, the research steps that are taken for designing efficient, reduced sized networks should be taken into consideration.

As stated throughout this dissertation estimating a waveform through acoustic features has substantial amount of computational and memory cost. For example, most used implementation of WaveNet model needs to optimize over 4 million parameters [152]. Consequently, DSP approaches that are well documented in the literature are implemented to relieve the neural network from the task of estimating neural network of the filter parameters of source filter model of speech with the aim of reducing neural network parameters. Such implementations are known as hybrid vocoder architecture [38].

Some of the examples of such implementations are LPCNet [135], GlotNet [151] and ExcitNet [153]. The diagrams in Figure 5.1 gives the processes of some AR neural vocoders.

To understand the intuition behind the hybrid vocoders, distinct features and their relation to the end-to-end approaches deserve to be studied. For example, LPCNet is a WaveRNN variant that combines linear prediction with recurrent neural networks to significantly improve the efficiency of speech synthesis [135]. LPCNet architecture is based on two neural network models. LPCNet first takes windowed frames of data and applies it DSP to produce a feature vector. These feature vector consists of 18 Bark Frequency Cepstral Coefficients (BFCC), period and pitch correlation. This feature vector has two use. First, a convolutional neural network is applied. Output of this network produce a learned condition set. Second, the linear prediction coefficients for autoregressive model is calculated using conventional DSP techniques. Second neural network in the LPCNet approach is a WaveRNN-like model. Together with condition set this network is tasked to predict excitation of the source-filter model of speech. In other hybrid vocoders, the main differences are with respect to the acoustic features and techniques to estimate autoregressive filter parameters and employed neural network architecture to estimate source characteristics of speech. For example, in GlotNet, a WaveNet model takes a set of acoustic features, whose output then fed into a digital filter that models vocal tract and glottal excitation through a process called Glottal inverse filtering [154].

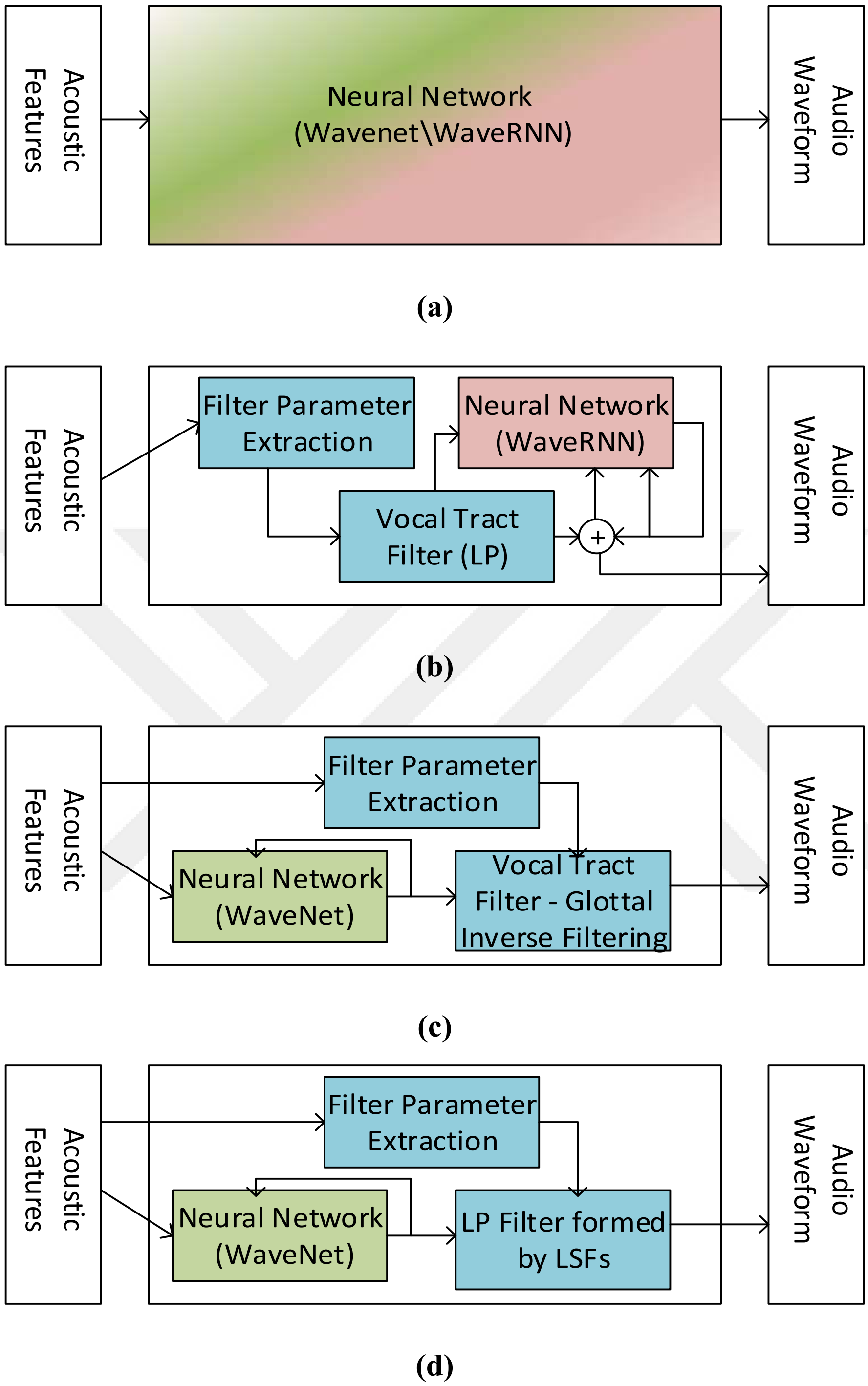


Figure 5.1 : Block diagram overviews for some hybrid approaches: (a)Wavenet/WaveRNN; (b)LPCNet; (c)GlotNet; (d)ExcitNet.

In Table 5.1, numbers of training parameters for baseline and most used implementations for some hybrid neural vocoders are given. Number of training parameters is an important aspect of neural network design. In terms of training time of neural networks, increased number of parameters results in longer optimization time. Additionally, memory resources of hardware to perform optimization procedures can be constrained by the increased number of weights.

Table 5.1 : Number of training parameters for some neural vocoders.

Neural Vocoder	Number of training parameters
Wavenet-30 [152]	4.57M
WaveRNN-896 [134]	3M
LPCNet [155]	1.24M (843K)
GlotNet [151]	602K – 1.56M

As stated before, implementing conventional DSP algorithms together with a neural network aims to obtain quality results with smaller sized models. To achieve that, other than AR models, a spectral consistency-based hybrid architecture is also shown to be beneficial.

In this dissertation a recent approach for audio reconstruction with deep neural networks called Deep Griffin-Lim Iteration (DeGLI) is applied [136]. Proposed by Masuyama et. al., DeGLI combines the concepts of spectral consistency-based phase reconstruction methods, deep unfolding to represent an iterative algorithm as a neural network and Plug-and-Play (PnP) denoising approach to increase convergence of iterative approach.

5.2 Deep Griffin-Lim Iteration

The ordinary Griffin Lim iteration can be defined as following.

$$\mathbf{C}_{\mathrm{n}+1} = \mathrm{P}_{C_1}\mathrm{P}_{C_2}(\mathbf{C}_{\mathrm{n}}) \tag{5.1}$$

The definition of the projections P_{C_1} and P_{C_2} are given in chapter 4 as equations (4.8) and (4.10). As in FGLA or GLA-ADMM, various updating mechanism can be applied the output of these two successive projections. DeGLI iteration applies similar approach as the GLA. Defining the successive projections as an intermediate state $\mathbf{T}_{\mathrm{n}}$, the iteration procedure can be given as

$$\mathbf{C}_{\mathrm{n}+1} = \mathbf{T}_{\mathrm{n}} - F\big(\mathbf{T}_{\mathrm{n}}, \mathrm{P}_{C_2}(\mathbf{C}_{\mathrm{n}}, \mathbf{S}), \mathbf{C}_{\mathrm{n}} | \mathbf{W}\big) = DeGLI(\mathbf{C}_{\mathrm{n}} | \mathbf{W}) \tag{5.2}$$

In this equation $F(.|\mathbf{W})$ is a deep neural network with trainable parameters $\mathbf{W}$. $DeGLI(.|\mathbf{W})$ is the function definition of a sub-block. Figure 5.2 shows the block diagram view of a DeGLI sub-block. In the Figure 5.2, $\mathbf{S}$ is the amplitude spectrogram of a complex spectrogram $\mathbf{C}$.

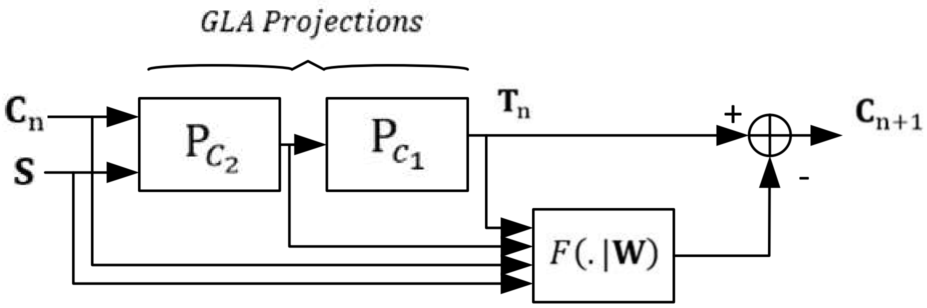


Figure 5.2 : Block diagram view of a DeGLI sub-block.

DeGLI is a method for reconstructing audio signals that combines the GLA with DNN. To reconstruct the phase for given STFTM, the framework uses a number of concatenated sub-blocks. Each sub-block contains a DNN. The number of sub-blocks can be adjusted to control the computational cost at inference time. Because of this design property the DeGLI allows the same trained DNN to be used in different applications with varying computational requirements. Similar approaches for defining iterative algorithms as neural networks are called as deep unfolding [156]. The DeGLI differs from deep unfolding by employing same DNN together with DSP based projections for every iteration step of GLA. This approach enables user to decide sub-block size for inference. Figure 5.3 shows the implementation of DeGLI for N iterations.

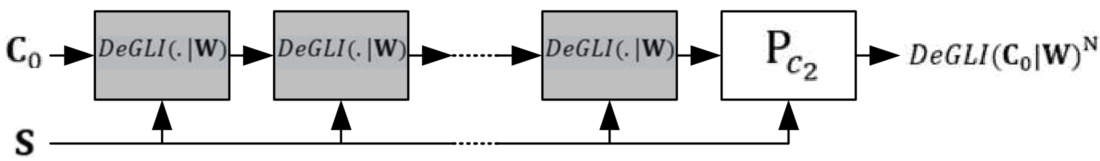


Figure 5.3 : DeGLI implementation for N iterations.

A unique feature of DeGLI is its training strategy for the DNN, which is specifically designed for iterative use. The DNN is trained using a denoising task with a single sub-block, which reduces the memory required for training and improves stability compared to end-to-end training. A similar approach in literature for is called Plug-and-Play. In PnP, arbitrary denoiser models are applied to increase the training efficiency [157]. In DeGLI, the denoiser model is not arbitrary. The DeGLI training approach considers GLA projections and designs the training sub-block with a task of increasing spectral consistency. The denoiser overview of DeGLI sub-block training can be seen in Figure 5.4.

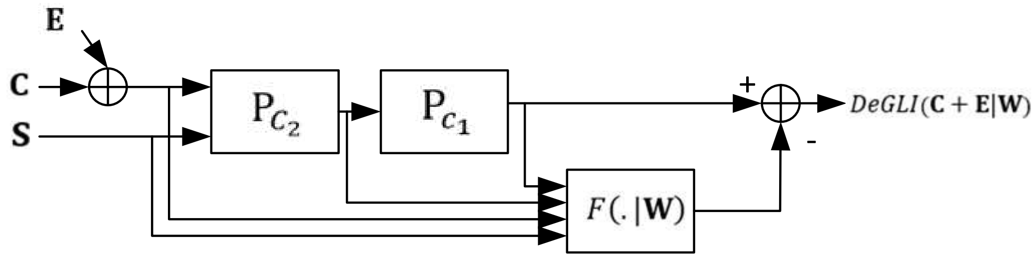


Figure 5.4 : DeGLI as denoiser.

The reason of using DNN in subtractive form is to make use of residual learning, which has been shown effective tool for denoising in the Deep Learning community. This approach presents compatibility with modified GLA implementations, less computational load, and also prevents instability caused elementwise multiplication and division. The computational coast and required memory for training this framework is substantially smaller with respect to an end-to-end training approach of common deep unfolding procedures.

It must be noted that in the training procedure magnitude of noisy data is always replaced by clean data, making DNN focus on reducing phase disturbance. That's why, the task becomes a phase reconstruction task. Additionally, this approach gives insight for training with augmented data. Since the successive GLA projections are present in training procedure of the sub-block, this framework prefers the "clean" data to have a consistent phase.

DeGLI employs a complex DNN that efficiently handles complex STFT coefficients using gating calculated from the magnitude of complex features and the given amplitude. The proposed training strategy and Complex DNN result in efficient training and high-quality output signals in regards of the subjective and also objective quality measures.

As stated by the researchers that proposed DeGLI any DNN can be incorporated into sub-block. But they have opted to implement a novel DNN called Amplitude-Informed

Gated Complex Convolutional Neural Network. Complex Convolution Layer is a previously introduced neural network layer by [158].

A 2D complex convolution is described as in equation (5.3).

$$Conv_C(\mathbf{C}|\mathbf{W}_C) = (\mathbf{W}_{\mathrm{Re}} * \mathbf{C}_{\mathrm{Re}} - \mathbf{W}_{\mathrm{Im}} * \mathbf{C}_{\mathrm{Im}}) + j(\mathbf{W}_{\mathrm{Re}} * \mathbf{C}_{\mathrm{Im}} + \mathbf{W}_{\mathrm{Im}} * \mathbf{C}_{\mathrm{Re}}) \tag{5.3}$$

Complex convolution layer can be regarded as generalization of the projection of spectrogram consistency with respect to the definition in [130], in which the successive ISTFT-STFT operations with additional parameter for faster reconstruction of time-frequency representation is defined as 2D-convolution. Practically, complex convolution layer replaces this additional parameter with trainable weights of a convolutional layer. Additionally, a gating mechanism, based on amplitude spectrogram is added to introduce another nonlinearity which employs the sigmoid function.

$$AmpGate(\mathbf{C}|\mathbf{W}_R) = Sigmoid(|\mathbf{C}| * \mathbf{W}_R) \tag{5.4}$$

Combined structure of complex convolution layer in (5.3) and amplitude based gating mechanism in (5.4) is named as Amplitude-Based-Gated-Complex-Convolution (*AGC*) layer.

$$AGC(\mathbf{C}|\mathbf{W}_C, \mathbf{W}_R) = Conv_C(\mathbf{C}|\mathbf{W}_C) \odot AmpGate(\mathbf{C}|\mathbf{W}_R) \tag{5.5}$$

In practice, the DNN in subblock also concatenates original spectrogram amplitude to the magnitude of the complex output of previous *AGC* layer. This structure is named as Amplitude-Informed-Gated-Complex-Convolution (*AI-GC*) layer. The overview of the DNN implemented in Deep Griffin-Lim iteration can be seen in the Figure 5.5.

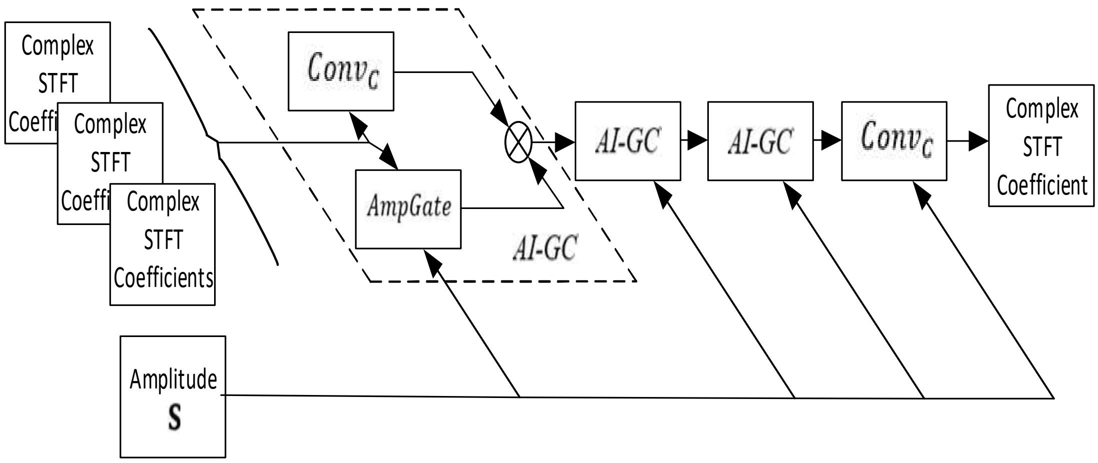


Figure 5.5 : The structure of DNN in the DeGLI.

The model size, and training epoch parameters for original DeGLI implementation can be seen in following Table 5.2.

Table 5.2 : Network Parameters of Original DeGLI [136].

Neural Network Architecture Parameters	
# of Amplitude Informed Gated Convolutional Layers	3
# of Complex Convolutional Layers	1
# of Channels	64
Filter size of AI-GC	5x3
Filter size of Last Complex Convolutional Layer	1x1
Stride for Convolutional Layers	1x1
# of Parameters	380k
Training Parameters	
Optimizer	ADAM
Initial Learning Rate Step Size	0.0004
Batch Size	32
# of Epochs	300
Randomly selected SNR values for Denoiser	[-6,12] dB

5.2.1 Experiments with DeGLI

Due to hardware constraints of the setup that is used for experiments that are conducted for this dissertation, actions are taken to reduce network parameter size. For instance, due to memory problems 32 batch size cannot be implemented. Additionally, predicted 300 epochs training time for 8 batch showed a 1-week time.

The implemented network for the present experiments and its training parameters are selected as given in the Table 5.3.

Additionally, similar reasons that lead to implement a smaller network also dictated to use a different dataset than original DeGLI paper used. Original DeGLI paper employs LJSpeech dataset, on the other hand for this dissertation, TIMIT dataset has to be used for training the neural network. Comparisons of the two datasets are provided in Table 5.4.

Table 5.3 : Implemented Network Parameters for DeGLI tests

Neural Network Architecture Parameters for small DeGLI	
# of Amplitude Informed Gated Convolutional Layers	3
# of Complex Convolutional Layers	1
# of Channels	32
Filter size of *AI-GC*	5x3
Filter size of Last Complex Convolutional Layer	1x1
Stride for Convolutional Layers	1x1
# of Parameters	98k
Training Parameters	
Optimizer	ADAM
Initial Learning Rate Step Size	0.0004
Batch Size	16
# of Epochs	100
Randomly selected SNR values for Denoiser	[-6,12] dB

Table 5.4 : Datasets for DeGLI tests.

Parameters for Datasets		
Dataset	LJSpeech	TIMIT
Sampling Rate	22050 Hz	16000 Hz
Window Length	1024	512
Hop Length	256	128
# Training samples	24064	11071

The smaller DeGLI block is trained on 16000Hz sampled, 1 second of duration TIMIT samples. The window length for STFT and ISTFT is 512 and hop length is 128. The network is trained for 100 epochs with ADAM optimizer that has the learning rate of 0.0004. The network is trained on smaller dataset size, smaller epoch number, with worse temporal and frequency resolution. The results are tested with respect to the GLA with different initial phase estimations. Performance is examined with respect to PESQ score. The results are both tested on a subset of TIMIT and LJSpeech samples.

LJSpeech samples are downsampled to 16000Hz resolution for consistency with network implementation and PESQ.

For tests GLA implemented for 150 iterations and for every 5 iteration PESQ scores are calculated. Similarly, DeGLI implemented with max 150 concatenated subblocks. The subblock number is incremented by 5. Figure 5.6 shows the comparison of GLA and smaller DeGLI implementation on TIMIT dataset.

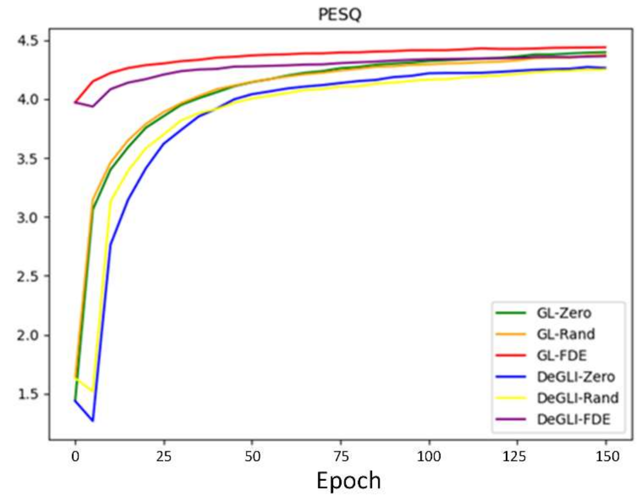


Figure 5.6 : Comparison of GLA and Smaller DeGLI on TIMIT.

The tests suggest that the DeGLI implementation suffers from reduced parameter number. But the main concern of this work is to understand the effect of novel FOC based methods on neural network based approaches.

As can be seen in the results, the FOC based phase estimation method has similar positive effect on DeGLI as GLA. Additional test on LJSpeech produce similar results. The results are provided in Figure 5.7.

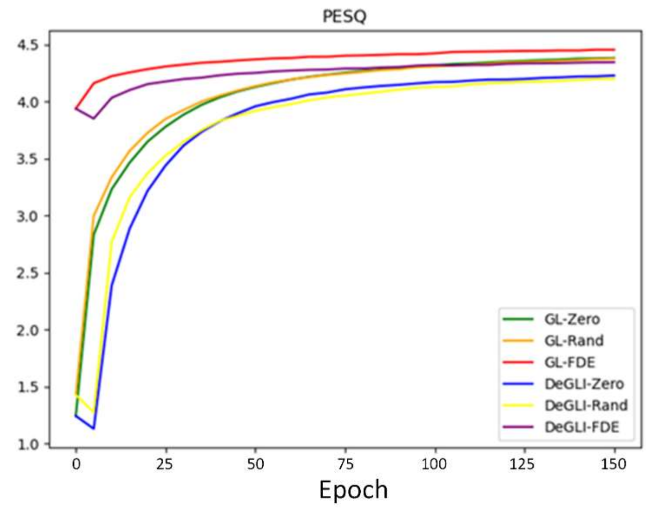


Figure 5.7: Comparison of GLA and Smaller DeGLI on LJSpeech.

5.2.2 Experiments with novel augmentation strategies

As stated in chapter 3, two FOC based scaling methods for data augmentation are proposed. For the audio reconstruction, the linear scale based method that was proposed in chapter 3 is applied because the spectral consistency based methods employ linear spectrogram based time-frequency representations.

As it is explained in chapter 3, contrary to VTLP, deriving linear frequency scale provides a continuous mapped scale, and a more conservative variation.

In Figure 5.8, the effect of the fractional order approach on linear frequency scale with respect to the integrations of 5 different α values [-0.1, -0.05 ,0, 0.05, 0.1] can be seen.

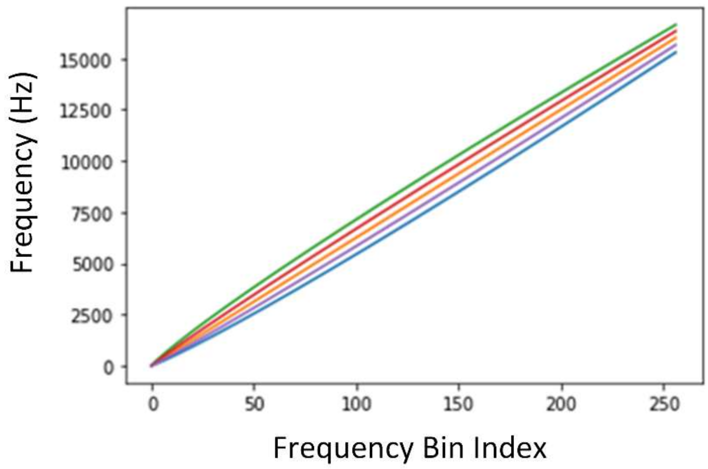


Figure 5.8 : Derived frequency scales. Blue: $\boldsymbol{\alpha = -0.1}$, Purple: $\boldsymbol{\alpha = -0.05}$, Orange: $\boldsymbol{\alpha = 0}$, Red: $\boldsymbol{\alpha = 0.05}$ and Green: $\boldsymbol{\alpha = 0.1}$ (green).

Using the mapped frequency scales, two augmentation strategies are applied. Augmentation Strategy 1 is quite straightforward. Complex STFT matrix is mapped and warped on fractionally derived frequency scale to produce complex augmented STFT matrix.

Augmentation Strategy 2 differs from strategy 1 with added method to its output. Complex STFT matrix is mapped and warped on fractionally derived frequency scale to produce complex augmented STFT matrix as in strategy 1. Additionally, the amplitude of complex augmented STFT matrix is calculated and new phase structure is produced by estimating with FDE based Phase Estimation method to produce new augmented STFT. Figure 5.9 shows the block diagram of the two augmentation strategies.

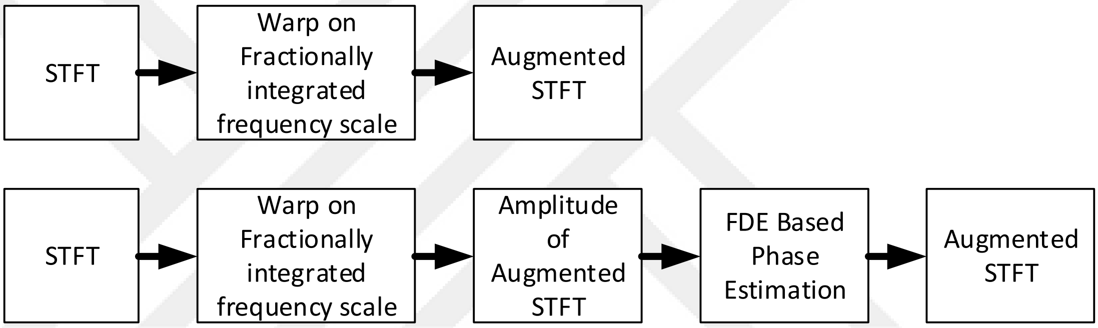


Figure 5.9 : Augmentation strategies: (above) Strategy 1, (below) Strategy 2.

To compare the performance of the both augmentation strategies, first test is conducted by applying randomly selected three differentiation order from the order array [-0.4, -0.3, -0.2, -0.1, 0.1, 0.2, 0.3, 0.4] to each training sample to increment dataset size 3 times using Augmentation Strategy 1. In Figure 5.10, the PESQ results for the first test is provided.

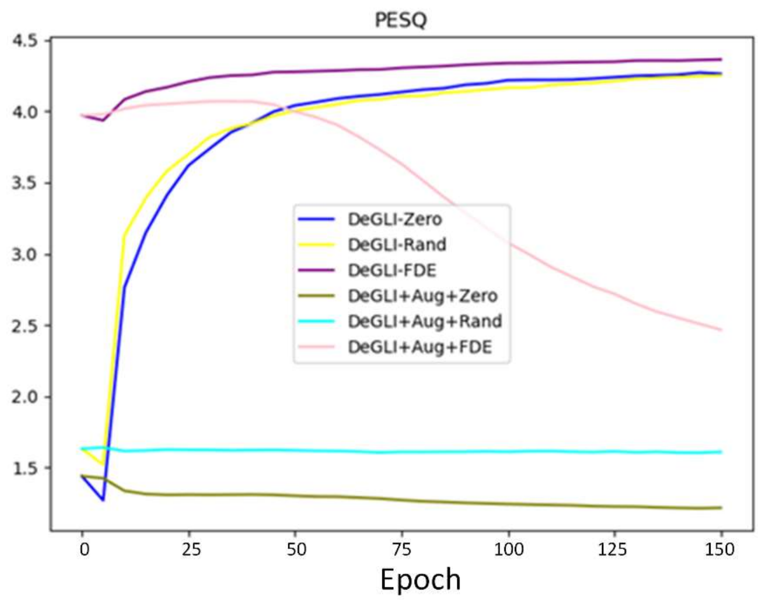


Figure 5.10 : Test 1 of Augmentation Strategy 1 with orders [-0.4, -0.3, -0.2, -0.1, 0.1, 0.2, 0.3, 0.4].

This result shows that the network becomes unable to learn the results when augmented with large variations.

After that original data is added to the network. The dataset size becomes 4 times incremented. Figure 5.11 shows the results for second test.

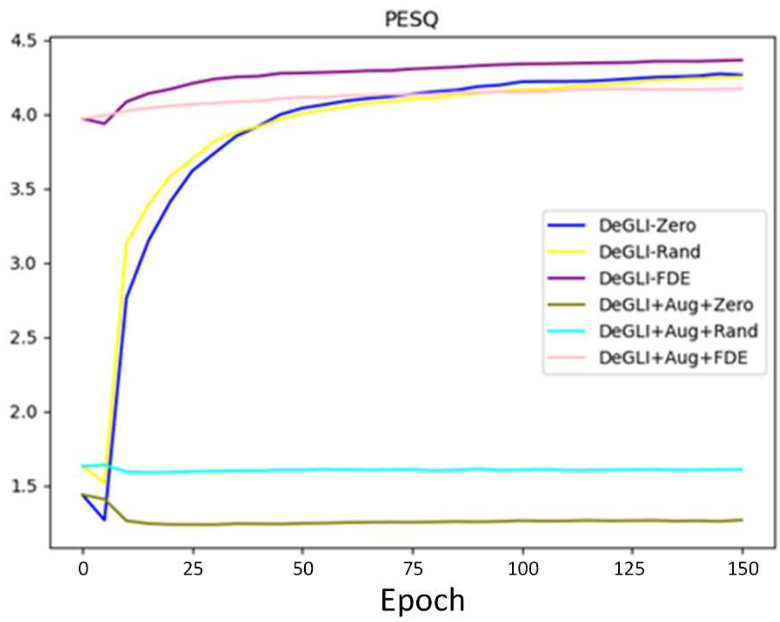


Figure 5.11 : Test 2 of Augmentation Strategy 1 with orders [-0.4, -0.3, -0.2, -0.1, 0.1, 0.2, 0.3, 0.4].

The result implies that adding original data improves augmented data results. But still the result is worse than not augmented data.

Due to the results of first two tests, the dataset is augmented with narrower range of α values. From the differentiation order array [-0.1, -0.05, 0.05, 0.1] randomly selected 2 is applied the dataset. Additionally, original data is added to augmented dataset to increase its size 3 times. In Figure 5.12, the corresponding results are provided.

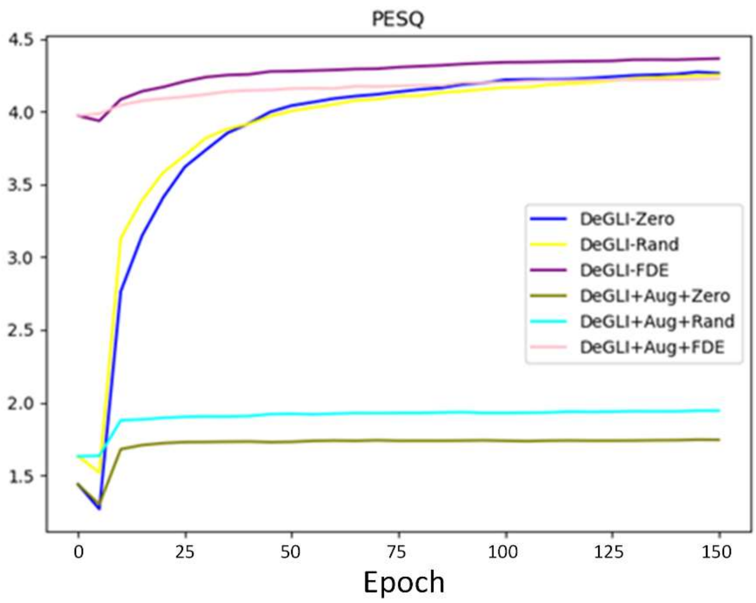


Figure 5.7 : Test 3 of Augmentation Strategy 1 with orders [-0.1, -0.05, 0.05, 0.1]

This result showes that the differentiation order variation should be narrower. But it also implies that Augmentation Strategy 1 is not applicable to this problem. The Augmentation Strategy 1 is applied on the complex STFT matrices. This approach leads to a worse phase consistency. Since in essence, the DeGLI uses phase consistency like GLA, another augmentation strategy that can implement a consistent phase should be applied.

For these reasons Augmentation Strategy 2 is applied. In this strategy, after augmenting the dataset, the amplitude STFT is calculated and from this augmented STFT, using Fractional Order Phase Estimation Method, new complex STFT is produced. That way a new data with new and consistent phase is produced. In Figure 5.13 spectral consistency results for both augmentation strategies are provided.

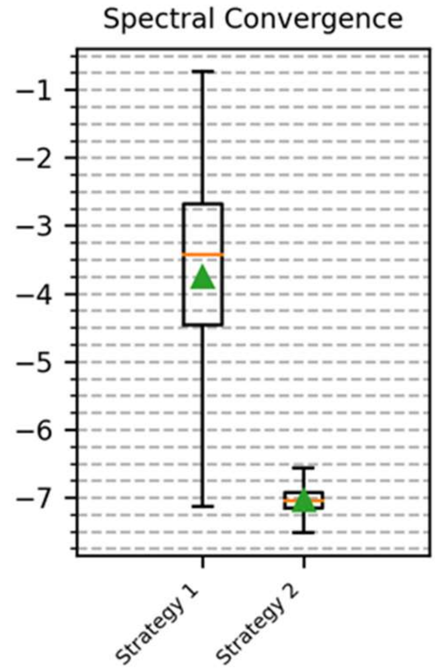


Figure 5.8 : log-SC comparison of both augmentation strategies.

From the Figure 5.13, it can be seen that when calculated over 30% TIMIT dataset the Augmentation Strategy 2 produces a smaller mean and variance with respect to SC results. This results strengthens the assumption that implementing Augmentation Strategy 2 would perform better quality scores for audio reconstruction.

The PESQ results for Augmentation Strategy 2 when it is applied to TIMIT dataset are provided in Figure 5.14.

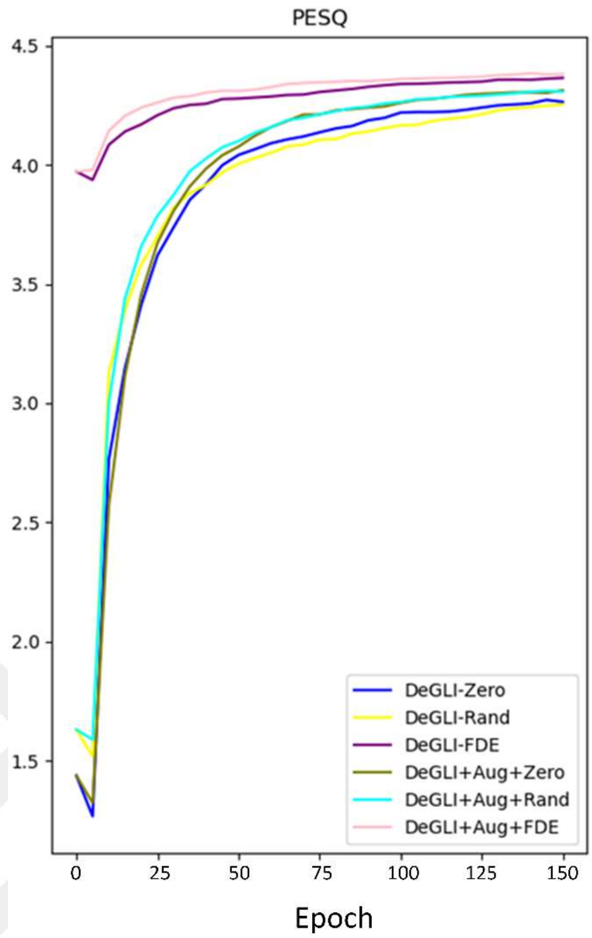


Figure 5.9 : Test 1 of Augmentation Strategy 2 with orders [-0.1, -0.05, 0.05, 0.1] on TIMIT.

From the results, it can be seen that augmenting dataset three times with Augmentation Strategy 2 slightly increases the reconstruction performance of DeGLI. It must be noted that data augmentation is not generally used in these types of problems. But the novel method proposed in this work successfully increases performance of neural network-based audio reconstruction model. It must be noted that the performance increased for all three phase initialization methods.

An additional test is concluded with LJSpeech dataset for sanity check. The result for this test is provided in Figure 5.15.

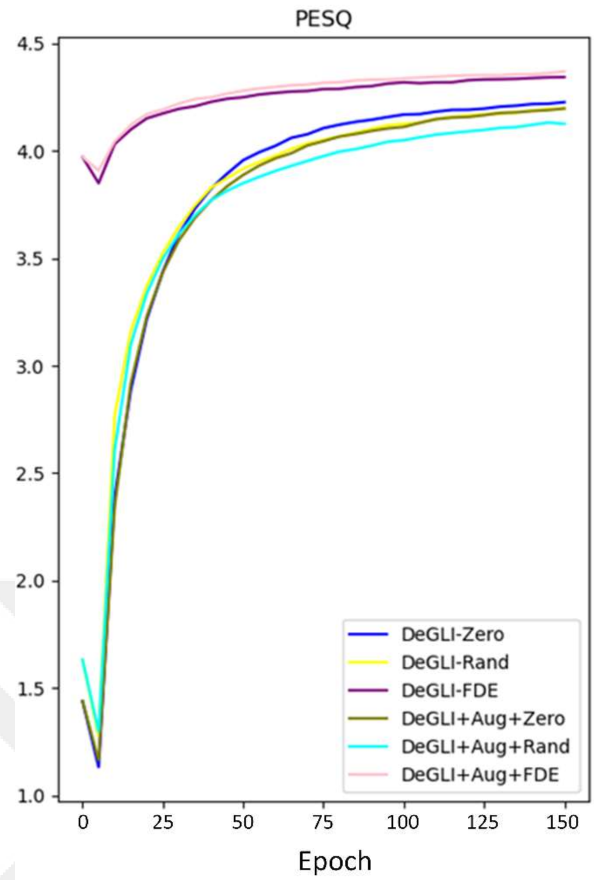


Figure 5.15 : Test 2 of Augmentation Strategy 2 with orders [-0.1, -0.05, 0.05, 0.1] on LJSpeech.

The results for LJSpeech is interesting. First, the best result is obtained with combination of phase initialization with Fractional Order Phase Estimator and model that was trained with Augmentation Strategy 2. This combination is successful for increasing DeGLI performance for two datasets and this outlines the applicability of methods that are proposed during thesis works.

Another result is that comparing to test with TIMIT dataset other phase initialization strategies produce better results for base Smaller DeGLI. For this result, it must be noted that implemented LJSpeech dataset in this experiment is a downsampled dataset. Additionally, TIMIT and LJSpeech datasets have different characteristics. But the most obvious answer should be that, although augmented, the DeGLI model that was trained on augmented dataset has learned the structure of augmented TIMIT samples. Each submodule in DeGLI has learned to denoise TIMIT data with estimated phase. That’s why the network performance is better with estimated phase but this quality does not necessarily translate to random or zero phase initializations.

In this NAS experiments, the DeGLI method is implemented. Due to the hardware constraints the convolutional layer filter numbers are rearranged to reduce parameter numbers. Firstly, the DeGLI method is compared with respect to standard GLA on TIMIT database. Secondly, the proposed FDE based phase estimation method is employed. Similarly to its application on GLA, this method increases performance for DeGLI. Thirdly, FOC based novel data augmentation method is applied to the training dataset to examine the effects of data augmentation on neural audio reconstruction. It is seen that the straightforward application of Fractional Order Frequency Scale method is not successful for audio reconstruction purposes. Additionally, another data augmentation strategy that combines data augmentation method that was explained in chapter 3 and phase estimation method that was explained in chapter 4 is proposed to produce augmented data. This combination increases audio synthesis performance, since estimating phase would create a more consistent complex STFT.

6. CONCLUSIONS

In this thesis dissertation, the provided introductory information outlines the purpose of the conducted experiments and research focus that are explained in the subsequent chapters. The purpose for the introductory chapter is the create an overview about the elementary concepts of the dissertation such as FOC and contemporary problems of audio signal processing. An important aspect of present signal processing approaches is due to that present inference models rely on large amounts of data. The direct result of this aspect is to provide tools for increasing amount of data when necessary. The secondary important aspect is based on the methods that produce inference models. The currently dominant neural network approaches contain very large number of parameters to optimize. Because of this, DSP tools are increasingly repurposed in a way to reduce the number of parameters of neural models. The introductory chapter discusses the validity of experimenting and designing novel methods with the unconventional FOC framework to answer the two important contemporary audio signal processing applications. Hence, the line of thought of this dissertation indicates the following question: "Is the fractional order calculus framework capable for providing novel methods to help the solve a number of problems of contemporary neural network-based audio signal classification and reconstruction approaches?"

In chapter 2, by providing the answer to the question: "What is fractional order calculus?", this framework is introduced. FOC enables both capabilities and complexity. As it is stated in this chapter, the complexities of the framework have been left it in sidelines for engineering applications. Especially in signal processing, increased complexity results in increased numerical calculation cost. Nevertheless, the capabilities of this framework continued to attract interest of many researchers. In literature, many engineering applications of fractional order calculus can be found. Especially, the provided literature for audio processing, time series estimation and image enhancement applications outlined the emphasized concepts of this thesis dissertation.

This dissertation is structured in a way that beginning with chapter 3, each chapter focuses on a specific type of application of audio signal processing. In each chapter, related literature background for the specific problem is provided. Following, the proposed FOC based methods are defined and experiments are conducted to show their capabilities.

In chapter 3, the problem at the hand is selected as Audio Classification Problem. As provided the literature background, this problem can be divided into subcategories based on the type of audio signal. The three types of audio signals are named as speech, music and environmental sound signal. Contrary to other two audio signal types, available data for environmental sound signals are relatively small. This aspect of environmental data provides a setback for applications of environmental sound processing but also increases the researches for on the domain of data augmentation methods. The chapter gives background about data augmentation methods in the literature. Because it employs a warping approach the VTLP method for augmenting audio data is outlined. In chapter three a total of three FOC based data augmentation methods are presented. The first method is called as Fractional Order Mask. This method employs the filtering parameters of the GL definition of fractional differintegral. In the literature, similar approaches are used for image enhancement and edge detection applications. The proposed augmentation method produces a 5x5 2D convolution kernel for α derivative order. The calculated filter parameters b_0, b_1, b_2 are placed in 8 directions, starting with the center of 5x5 convolution matrix. The proposed methods can be applied on all 2D representations of audio data. Since the most used representations for audio data are Mel Spectrogram and Log-Mel Spectrogram representations, the experiments with this method are restricted to these two audio data representations.

The other two presented methods are called Fractional Order Frequency Scale and Fractional Order Mel Scale methods. These two methods employ the RL definition of fractional order differintegral and numerical methods derived from this definition. These two methods essentially work as rescaling tools for the frequency scale or the Mel scale. By rescaling, the values in frequency bins of spectrogram representations are changed with respect to a derivation order α. Since the experiments in chapter 3 in is focused on ESC, Fractional Order Frequency Scale method is excluded. This method is employed to the problem in chapter 5. The experiments are conducted with the aim of inspecting accuracy result increases of neural network models. To this purpose, the experiments are conducted both on a 5-layer CNN model and a 18-layer ResNet. The results show that, a combined data augmentation procedure with proposed methods is capable of providing up to 7.7% increase for 5-layer CNN with Log-Mel Spectrograms as input and more than 9% increase for ResNet for the same augmented dataset. It must be noted that, the achieved accuracy results do not overcome the benchmark results in the literature. But since the scope of the experiments are related to demonstrating the capabilities of a mathematical framework, the achieved accuracy result increases provide validity for assumptions of this dissertation. The α derivation parameters for the implemented approaches are selected experimentally. A future work should focus on implementing procedures for automatic selection of this parameter.

Chapter 4 provides literature background of audio synthesis problem, its history and its contemporary importance. Since, audio signals are represented mostly in reduced feature spaces, reconstructing a time domain representation has a critical importance. The history of vocoding methods has provided such group of reconstruction approaches. Due to their relations of proposed method for signal reconstructions, in this chapter, information about phase reconstruction with SPSI method and spectral consistency-based iterative methods are deepened. This way, the significance of constructing a valid time domain audio signal from its amplitude spectrogram is underlined. Spectral consistency-based approaches lead to implementation iterative methods like GLA and its novel forms. The GLA employs successive projections of an amplitude spectrogram with initial phase spectrogram estimates. The success of a phase reconstruction algorithm is generally showed with respect to its SC and PESQ measure values. To prove the capabilities of the proposed FDE based method, the experimental results are conducted the provide the results of these measurements. The proposed method employs an approach of modelling a signals PSDF with fractional differential equations. This approach is widely used in ARFIMA time series estimations. By estimating the parameter β of FDE with least squares estimation, the long memory of property of fractional differential equations is linked to estimation of instantaneous frequency of a peak of a windowed audio spectrum. Expanding this relation to the amplitude spectrogram in a similar way like SPSI a novel phase reconstruction method is presented. The first group of experiments are provided the comparison of the proposed method and the SPSI by changing hop size, with respect to the SC and PESQ measures. The results show the usage of FOC framework provides up to 4% better quality than SPSI. Additional experiments look into the usage of these methods as initial phase estimators for spectral consistency based iterative methods, namely, GLA, FGLA, GLA-ADMM. The results of the second group of experiments underlines the capabilities of the proposed audio reconstruction method which implements FDE model for instantaneous phase estimation. The proposed method is focused on estimating frequency component of the phase gradient. A future work can be conducted to estimate time component of the phase gradient of a signal. Additionally, the proposed method can be extended to include all the values between two valleys around a peak to estimate instantaneous frequency in relation to the peak frequency bin.

As shown in the literature overview in chapter 5, NAS is the highly interested contemporary research and application topic. The advancements in DNN provides state of the art results for audio synthesis problem. As viewed in the elementary concepts of NAS, a substantial work in this area is focused on reducing the number of parameters of a neural network implementation to provide smaller sized models which are flexible for implementation of different hardware setups. For this purpose, filter-based speech production models are combined with neural networks. Recently, spectral consistency approach is implemented with a neural network. Abbreviated as DeGLI, this concept aims to provide a capable audio reconstruction procedure with smaller sized training parameters for neural network. This architecture depends on the design of smaller submodules that consist successive projections and a neural network. The important contribution of this architecture is its capability to be trained as a denoiser. A submodule that is trained as a denoiser than concatenated with its copies. This approach imitates the iteration procedure in standard GLA. The DeGLI provides flexibility in size. Depending on the hardware constraints, a select size of submodules can be concatenated. In chapter 5, the outcomes of chapters 3 and 4 are combined together to evaluate the applicability of FOC based methods to the NAS problem. In this sense, the proposed Fractional Order Frequency Scale data augmentation method in chapter 3 is used to increase data space. The proposed FDE based phase estimation approach is utilized for calculating initial phase. The hardware, on which the experiments of this dissertation conducted, dictated to implement a smaller sized neural network and a reduced sampling rate with less amount of data. The results of first experiment show that, the proposed phase initialization method has similar effect on the implemented DeGLI model to the standard GLA. The results of this experiment are shown with respect to the PESQ measures. The results also show that reducing parameter size for the implementation results in worse reconstruction quality. Following the standard GLA is excluded from the experiments. To improve the audio reconstruction quality, two data augmentation strategies that employ the proposed Fractional Order Frequency Scale method is tested. For the tests, dataset size is increased three times by augmenting it with different values of derivation order α. The important difference of these two strategies are as follows: Strategy 1 directly uses the complex spectrogram and augments it by applying the data augmentation method; Strategy 2 calculates the amplitude spectrogram of augmented data and by employing FDE based phase estimation method reproduces a new data. The experiments with Strategy 1 produced catastrophic results in terms of reconstruction quality. This is due to its effect on spectral consistency. Comparing two augmentation methods in terms of SC shows that Strategy 2 is capable of producing more spectrally consistent data. As a result, this novel strategy increases the audio spectrogram reconstruction quality of implemented DeGLI model. This result provides an intuition for capabilities of data augmentation methods that produce more spectrally consistent augmented data for training audio synthesis models.

In conclusion, the results of this dissertation can be grouped into three. Firstly, the FOC framework is a capable framework for designing methods of data augmentation, which addresses an important contemporary problem of present, highly demanded data driven approaches. Secondly, FDE based approaches provide quality estimators for a speech signal's phase component. This approach can also be used together with data driven audio synthesis methods. Although it needs a follow up research, a third result shows that a data augmentation tool that produces spectrally consistent spectrograms can provide increased performance for neural audio reconstruction models. These three results show that the validity of using FOC framework to address problems of contemporary audio classification and synthesis works.

REFERENCES

[1] **Ross, B.** (1977) The development of fractional calculus 1695-1900, *Historia Mathematica*, 4(1), pp. 75–89. doi: 10.1016/0315-0860(77)90039-8.

[2] **Podlubny, I.** (1999) *Fractional Differential Equations*. San Diego: Academic Press.

[3] **Petráš, I.** (2011) *Fractional Order Nonlinear Systems*. Higher Education Press and Springer-Verlag.

[4] **Das, S. and Pan, I.** (2012) *Fractional Order Signal Processing: Introductory Concepts and Applications*. Heidelberg: Springer. doi: 10.1007/978-3-642-23117-9.

[5] **Gutiérrez, R. E., Rosário, J. M. and Tenreiro Machado, J**. (2010) Fractional order calculus: Basic concepts and engineering applications, *Mathematical Problems in Engineering*, 2010, pp. 1–19. doi: 10.1155/2010/375858.

[6] **Valério, D., Ortigueira, M. D. and Lopes, A. M.** (2022) How many fractional derivatives are there?, *Mathematics*, 10(5), pp. 1–18. doi: 10.3390/math10050737.

[7] **De Oliveira, E. C. and Tenreiro Machado, J. A.** (2014) A review of definitions for fractional derivatives and integral, *Mathematical Problems in Engineering*, 2014(1940). doi: 10.1155/2014/238459.

[8] **Ortigueira, M. D. and Tenreiro Machado, J. A.** (2015) What is a fractional derivative?, *Journal of Computational Physics*. Elsevier Inc., 293, pp. 4–13. doi: 10.1016/j.jcp.2014.07.019.

[9] **Adams, M.** (2019) differint: A Python package for numerical fractional calculus. Available at: http://arxiv.org/abs/1912.05303.

[10] **Hosking, J. R. M.** (1981) Fractional differencing, *Biometrika*, pp. 165-175. doi: 10.2307/2335817.

[11] **Sabanal, S. and Nakagawa, M.** (1996) The fractal properties of vocal sounds and their application in the speech recognition model, *Chaos, Solitons & Fractals*, 7(11), pp. 1825–1843. doi: 10.1016/S0960-0779(96)00043-4.

[12] **Al-Akaidi, M.** (2004) *Fractal Speech Processing*. Cambridge University Press. doi: 10.1017/CBO9780511754548.

[13] **Lévy-Véhel, J.** (1995) Fractal approaches in signal processing, *Fractals*, 03(04), pp. 755–775. doi: 10.1142/S0218348X95000679.

[14] **Ortigueira, M. D. and Trujillo, J. J.** (2011) Generalized Grünwald–Letnikov fractional derivative and its Laplace and Fourier transforms, *Journal of Computational and Nonlinear Dynamics*, 6(3). doi: 10.1115/1.4003136.

[15] **Assaleh, K. and Ahmad, W. M.** (2007) Modeling of speech signals using fractional calculus, in *2007 9th International Symposium on Signal Processing and Its Applications*. IEEE, pp. 1–4. doi: 10.1109/ISSPA.2007.4555563.

[16] **Despotovic, V., Skovranek, T. and Peric, Z.** (2018) One-parameter fractional linear prediction, *Computers & Electrical Engineering*, 69, pp. 158–170. doi: 10.1016/j.compeleceng.2018.05.020.

[17] **Skovranek, T., Despotovic, V. and Peric, Z.** (2019) Optimal fractional linear prediction with restricted memory, *IEEE Signal Processing Letters*, 26(5), pp. 760–764. doi: 10.1109/LSP.2019.2908278.

[18] **Skovranek, T. and Despotovic, V.** (2019) Audio signal processing using fractional linear prediction, *Mathematics*, 7(7), p. 580. doi: 10.3390/math7070580.

[19] **Maragos, P. and Young, K. L.** (1990) Fractal excitation signals for CELP speech coders, in *International Conference on Acoustics, Speech, and Signal Processing*. IEEE, pp. 669–672. doi: 10.1109/ICASSP.1990.115838.

[20] **Maragos, P. and Potamianos, A.** (1999) Fractal dimensions of speech sounds: Computation and application to automatic speech recognition, *The Journal of the Acoustical Society of America*, 105(3), pp. 1925–1932. doi: 10.1121/1.426738.

[21] **Tamulevičius, G., Karbauskaitė, R. and Dzemyda, G.** (2019) Speech emotion classification using fractal dimension-based features', *Nonlinear Analysis: Modelling and Control*, 24(5). doi: 10.15388/NA.2019.5.1.

[22] **Pitsikalis, V. and Maragos, P.** (2009) Analysis and classification of speech signals by generalized fractal dimension features, *Speech Communication*, 51(12), pp. 1206–1223. doi: 10.1016/j.specom.2009.06.005.

[23] **Mathieu, B., Melchior, P., Oustaloup, A. and Ceyral, C.** (2003) Fractional differentiation for edge detection, *Signal Processing*, 83(11), pp. 2421–2432. doi: 10.1016/S0165-1684(03)00194-4.

[24] **Henriques, M., Valério, D., Gordo, P. and Melicio, R.** (2021) Fractional-order colour image processing, *Mathematics*, 9(5), pp. 1–15. doi: 10.3390/math9050457

[25] **Padlia, M. and Sharma, J.** (2018) Brain tumor segmentation from MRI using fractional sobel mask and watershed transform, *IEEE International Conference on Information, Communication, Instrumentation and Control, ICICIC 2017*, 2018-Janua, pp. 1–6. doi: 10.1109/ICOMICON.2017.8279121.

[26] **Gomolka, Z.** (2018) Backpropagation algorithm with fractional derivatives, *ITM Web of Conferences*, 21(4), pp. 1-10 doi: 10.1051/itmconf/20182100004.

[27] **Khan, S., Ahmad, J., Naseem, I. and Moinuddin, M.** (2018) A novel fractional gradient-based learning algorithm for recurrent neural networks, *Circuits, Systems, and Signal Processing*. Springer US, 37(2), pp. 593–612. doi: 10.1007/s00034-017-0572-z.

[28] **Bao, C., Pu, Y. and Zhang, Y.** (2018) Fractional-order deep backpropagation neural network, *Computational Intelligence and Neuroscience*, 2018. doi: 10.1155/2018/7361628.

[29] **Mannepalli, K., Sastry, P. N. and Suman, M.** (2017) A novel adaptive fractional deep belief networks for speaker emotion recognition, *Alexandria Engineering Journal*. Faculty of Engineering, Alexandria University, 56(4), pp. 485–497. doi: 10.1016/j.aej.2016.09.002.

[30] **Alidousti, J.** (2020) Stability and bifurcation analysis for a fractional prey–predator scavenger model, *Applied Mathematical Modelling*. Elsevier Inc., 81, pp. 342–355. doi: 10.1016/j.apm.2019.11.025.

[31] **Alidousti, J. and Ghafari, E.** (2020) Dynamic behavior of a fractional order prey-predator model with group defense, *Chaos, Solitons and Fractals*. Elsevier Ltd, 134. doi: 10.1016/j.chaos.2020.109688.

[32] **Li, P., Li, Y., Gao, R., Xu, C. and Shang, Y.** (2022) New exploration on bifurcation in fractional-order genetic regulatory networks incorporating both type delays, *European Physical Journal Plus*. Springer Berlin Heidelberg. doi: 10.1140/epjp/s13360-022-02726-3.

[33] **Li, P., Yan, J., Xu, C. and Shang, Y.** (2022) Dynamic analysis and bifurcation study on fractional-order tri-neuron neural networks incorporating delays, *Fractal and Fractional*, 6(3), pp. 1–25. doi: 10.3390/fractalfract6030161.

[34] **Huang, C., Wang, J., Chen, X. and Cao, J.** (2021) Bifurcations in a fractional-order BAM neural network with four different delays, *Neural Networks*. Elsevier Ltd, 141, pp. 344–354. doi: 10.1016/j.neunet.2021.04.005.

[35] **Huang, C., Liu, H., Shi, X., Chen, X., Xiao, M., Wang, Z. and Cao, J.** (2020) Bifurcations in a fractional-order neural network with multiple leakage delays, *Neural Networks*. Elsevier Ltd, 131, pp. 115–126. doi: 10.1016/j.neunet.2020.07.015.

[36] **Vaseghi, S. V.** (2007) *Multimedia Signal Processing: Theory and Applications in Speech, Music and Communications*, *Multimedia Signal Processing: Theory and Applications in Speech, Music and Communications*. doi: 10.1002/9780470066508.

[37] **Wu, Y. and Lee, T.** (2018) Reducing model complexity for DNN based large-scale audio classification, *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. IEEE, 2018-April, pp. 331–335. doi: 10.1109/ICASSP.2018.8462168.

[38] **Purwins, H., Li, B., Virtanen, T., Schluter, J., Chang, S.-Y. and Sainath, T.** (2019) Deep learning for audio signal processing, *IEEE Journal of Selected Topics in Signal Processing*, 13(2), pp. 206–219. doi: 10.1109/JSTSP.2019.2908700.

[39] **Sharma, G., Umapathy, K. and Krishnan, S.** (2020) Trends in audio signal feature extraction methods, *Applied Acoustics*. Elsevier Ltd, 158, p. 107020. doi: 10.1016/j.apacoust.2019.107020.

[40] **Alharbi, S., Alrazgan, M., Alrashed, A., Alnomasi, T., Almojel, R., Alharbi, R., Alharbi, S., Alturki, S., Alshehri, F. and Almojil, M.** (2021) Automatic speech recognition: Systematic literature review, *IEEE Access*. IEEE, 9, pp. 131858–131876. doi: 10.1109/ACCESS.2021.3112535.

[41] **Iliadis, L. A., Sotiroudis, S. P., Kokkinidis, K., Sarigiannidis, P., Nikolaidis, S. and Goudos, S. K.** (2022) Music deep learning: A survey on deep learning methods for music processing, *2022 11th International Conference on Modern Circuits and Systems Technologies, MOCAST 2022*. IEEE, pp. 22–25. doi: 10.1109/MOCAST54814.2022.9837541.

[42] **Bansal, A. and Garg, N. K.** (2022) Environmental sound classification: A descriptive review of the literature, *Intelligent Systems with Applications*. Elsevier Ltd, 16(April), p. 200115. doi: 10.1016/j.iswa.2022.200115.

[43] **Alías, F., Socoró, J. C. and Sevillano, X.** (2016) A review of physical and perceptual feature extraction techniques for speech, music and environmental sounds, *Applied Sciences*, 6(5). doi: 10.3390/app6050143.

[44] **Shannon, B. J. and Paliwal, K. K.** (2003) A comparative study of filter bank spacing for speech recognition, *Microelectronic Engineering Research Conference*, 41, pp. 310–312. Available at: https://maxwell.ict.griffith.edu.au/spl/publications/papers/merc03_ben.pdf.

[45] **Zbancioc, M. and Costin, M.** (2003) Using neural networks and LPCC to improve speech recognition, *SCS 2003 - International Symposium on Signals, Circuits and Systems, Proceedings*, 2, pp. 445–448. doi: 10.1109/SCS.2003.1227085.

[46] **Lu, L., Jiang, H. and Zhang, H. J.** (2001) A robust audio classification and segmentation method, *Proceedings of the ACM International Multimedia Conference and Exhibition*, (IV), pp. 203–211. doi: 10.1145/500170.500173.

[47] **Silva, D. F., De Souza, V. M. A., Batista, G. E. A. P. A. and Giusti, R.** (2012) Spoken digit recognition in portuguese using line spectral frequencies, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7637 LNAI, pp. 241–250. doi: 10.1007/978-3-642-34654-5_25.

[48] **Kleijn, W. B., Bäckström, T. and Alku, P.** (2003) On line spectral frequencies, *IEEE Signal Processing Letters*, 10(3), pp. 75–77. doi: 10.1109/LSP.2003.809035.

[49] **Deng, L., Li, J., Huang, J.T., Yao, K., Yu, D., Seide, F., Seltzer, M., Zweig, G., He, X., Williams, J. and Gong, Y.** (2013) Recent advances in deep learning for speech research at Microsoft, in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, pp. 8604–8608. doi: 10.1109/ICASSP.2013.6639345.

[50] **Giannakopoulos, T. and Pikrakis, A.** (2014) *Introduction to Audio Analysis: A MATLAB Approach*. 1st edn. Academic Press.

[51] **Ganchev, T., Fakotakis, N. and Kokkinakis, G.** (2005) Comparative evaluation of various MFCC implementations on the speaker verification task, *October*, 1(3), pp. 191–194. doi: 10.1.1.75.8303.

[52] **Cataltepe, Z., Yaslan, Y. and Sonmez, A.** (2007) Music genre classification using MIDI and audio features, *Eurasip Journal on Advances in Signal Processing*, 2007. doi: 10.1155/2007/36409.

[53] **Lu, L., Zhang, H. J. and Li, S. Z.** (2003) Content-based audio classification and segmentation by using support vector machines, *Multimedia Systems*, 8(6), pp. 482–492. doi: 10.1007/s00530-002-0065-0.

[54] **West, K. and Cox, S.** (2004) Features and classifiers for the automatic classification of musical audio signals, *Proceeding of the International Society for Music Information Retrieval Conference (ISMIR)*, pp. 1–6.

[55] **Dhanalakshmi, P., Palanivel, S. and Ramalingam, V.** (2011) Classification of audio signals using AANN and GMM, *Applied Soft Computing Journal*, 11(1), pp. 716–723. doi: 10.1016/j.asoc.2009.12.033.

[56] **Foote, J.** (1999) Overview of audio information retrieval, *Multimedia Systems*, 7(1), pp. 2–10. doi: 10.1007/s005300050106.

[57] **Humphrey, E. J., Bello, J. P. and LeCun, Y.** (2012) Moving beyond feature design: deep architectures and automatic feature learning in music informatics, *International Society for Music Information Retrieval Conference*, (Ismir), pp. 403–408. Available at: https://files.nyu.edu/jb2843/public/Publications.html.

[58] **Peeters, G. and Richard, G.** (2021) Deep Learning for audio and music, *Multi-faceted Deep Learning: Models and Data*. Springer International Publishing, pp. 231–266. doi: 10.1007/978-3-030-74478-6_10.

[59] **Bai, Z. and Zhang, X. L.** (2021) Speaker recognition based on deep learning: An overview, *Neural Networks*. Elsevier Ltd, 140, pp. 65–99. doi: 10.1016/j.neunet.2021.03.004.

[60] **Bahar, P., Zeyer, A., Schlüter, R. and Ney, H.** (2019) On using SpecAugment for end-to-end speech translation, pp. 2–9. Available at: http://arxiv.org/abs/1911.08876.

[61] **Hinton, G., Deng, L., Yu, D., Dahl, G.E., Mohamed, A.R., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T.N. and Kingsbury, B.** (2012) Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, *IEEE Signal Processing Magazine*, 29(6), pp. 82–97. doi: 10.1109/MSP.2012.2205597.

[62] **Abdel-Hamid, O., Mohamed, A. R., Jiang, H., Deng, L., Penn, G. and Yu, D.** (2014) Convolutional neural networks for speech recognition, *IEEE Transactions on Audio, Speech and Language Processing*. IEEE, 22(10), pp. 1533–1545. doi: 10.1109/TASLP.2014.2339736.

[63] **Graves, A., Mohamed, A. and Hinton, G.** (2013) Speech recognition with deep recurrent neural networks', in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, pp. 6645–6649. doi: 10.1109/ICASSP.2013.6638947.

[64] **Li, J., Mohamed, A., Zweig, G. and Gong, Y.** (2016) LSTM time and frequency recurrence for automatic speech recognition, *2015 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2015 - Proceedings*. IEEE, pp. 187–191. doi: 10.1109/ASRU.2015.7404793.

[65] **Hsu, W. N., Zhang, Y., Lee, A. and Glass, J.** (2016) Exploiting depth and highway connections in convolutional recurrent deep neural networks for speech recognition, *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 08-12-Sept, pp. 395–399. doi: 10.21437/Interspeech.2016-515.

[66] **Chan, W., Jaitly, N., Le, Q. and Vinyals, O.** (2016) Listen, attend and spell: A neural network for large vocabulary conversational speech recognition, in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 4960–4964. doi: 10.1109/ICASSP.2016.7472621.

[67] **Mtshali, P. and Khubisa, F.** (2019) A smart home appliance control system for physically disabled people, *2019 Conference on Information Communications Technology and Society, ICTAS 2019*. IEEE, pp. 1–5. doi: 10.1109/ICTAS.2019.8703637.

[68] **Tatman, R. and Kasten, C.** (2017) Effects of talker dialect, gender & race on accuracy of bing speech and youtube automatic captions, *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2017-Augus, pp. 934–938. doi: 10.21437/Interspeech.2017-1746.

[69] **Bello, J. P., Daudet, L., Abdallah, S., Duxbury, C., Davies, M. and Sandler, M. B.** (2005) A tutorial on onset detection in music signals, *IEEE Transactions on Speech and Audio Processing*, 13(5), pp. 1035–1046. doi: 10.1109/TSA.2005.851998.

[70] **Bay, M., Ehmann, A. F. and Stephen Downie, J.** (2009) 'Evaluation of multiple-F0 estimation and tracking systems', *Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009*. Kobe, Japan, pp. 315–320. doi: 10.5281/zenodo.1418241.

[71] **Bogdanov, D., Wack, N., Gómez Gutiérrez, E., Gulati, S., Boyer, H., Mayor, O., Roma Trepat, G., Salamon, J., Zapata González, J.R. and Serra, X.** (2013) Essentia: An audio analysis library for music information retrieval, *Proceedings of the 14th International Society for Music Information Retrieval Conference, ISMIR 2013*, pp. 493–498.

[72] **Zhu, Y., Kankanhalli, M. S. and Gao, S.** (2005) Music key detection for musical audio, *Proceedings of the 11th International Multimedia Modelling Conference, MMM 2005*, pp. 30–37. doi: 10.1109/MMMC.2005.56.

[73] **Papadopoulos, H. and Peeters, G.** (2007) Large-scale study of chord estimation algorithms based on chroma representation and HMM, in *2007 International Workshop on Content-Based Multimedia Indexing*. IEEE, pp. 53–60. doi: 10.1109/CBMI.2007.385392.

[74] **Salamon, J. and Bello, J. P.** (2017) Deep convolutional neural networks and data augmentation for environmental sound classification, *IEEE Signal Processing Letters*, 24(3), pp. 279–283. doi: 10.1109/LSP.2017.2657381.

[75] **Sailor, H. B., Agrawal, D. M. and Patil, H. A.** (2017) Unsupervised filterbank learning using convolutional restricted Boltzmann Machine for environmental sound classification, *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2017, pp. 3107–3111. doi: 10.21437/Interspeech.2017-831.

[76] **Salamon, J., Jacoby, C. and Bello, J. P.** (2014) A dataset and taxonomy for urban sound research, *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, (1), pp. 1041–1044. doi: 10.1145/2647868.2655045.

[77] **Salamon, J. and Bello, J. P.** (2015) 'Unsupervised feature learning for urban sound classification', in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 171–175. doi: 10.1109/ICASSP.2015.7177954.

[78] **Piczak, K. J.** (2015) Environmental sound classification with convolutional neural networks, in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, pp. 1–6. doi: 10.1109/MLSP.2015.7324337.

[79] **Chen, Y., Guo, Q., Liang, X., Wang, J. and Qian, Y.** (2019) Environmental sound classification with dilated convolutions, *Applied Acoustics*. Elsevier Ltd, 148, pp. 123–132. doi: 10.1016/j.apacoust.2018.12.019.

[80] **Mushtaq, Z. and Su, S. F.** (2020) Environmental sound classification using a regularized deep convolutional neural network with data augmentation, *Applied Acoustics*. Elsevier Ltd, 167, p. 107389. doi: 10.1016/j.apacoust.2020.107389.

[81] **Piczak, K. J.** (2015) ESC: Dataset for environmental sound classification, *MM 2015 - Proceedings of the 2015 ACM Multimedia Conference*, pp. 1015–1018. doi: 10.1145/2733373.2806390.

[82] **Halevy, A., Norvig, P. and Pereira, F.** (2009) The unreasonable effectiveness of data', *IEEE Intelligent Systems*, 24(2), pp. 8–12. doi: 10.1109/MIS.2009.36.

[83] **Shorten, C. and Khoshgoftaar, T. M.** (2019) A survey on image data augmentation for deep learning', *Journal of Big Data*. Springer International Publishing, 6(1). doi: 10.1186/s40537-019-0197-0.

[84] **Perez, L. and Wang, J.** (2017) The effectiveness of data augmentation in image classification using deep learning. Available at: http://arxiv.org/abs/1712.04621.

[85] **Taylor, L. and Nitschke, G.** (2018) Improving deep learning with generic data augmentation, in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, pp. 1542–1547. doi: 10.1109/SSCI.2018.8628742.

[86] **Ragni, A., Knill, K. M., Rath, S. P., and Gales, M. J. F.** (2014) Data augmentation for low resource languages, *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, (September), pp. 810–814. doi: 10.21437/interspeech.2014-207.

[87] **Rebai, I., Benayed, Y., Mahdi, W., and Lorré, J. P.** (2017) Improving speech recognition using data augmentation and acoustic model fusion', *Procedia Computer Science*, 112, pp. 316–322. doi: 10.1016/j.procs.2017.08.003.

[88] **Park, D.S., Chan, W., Zhang, Y., Chiu, C.C., Zoph, B., Cubuk, E.D. and Le, Q.V.** (2019) Specaugment: A simple data augmentation method for automatic speech recognition, *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2019-Septe, pp. 2613–2617. doi: 10.21437/Interspeech.2019-2680.

[89] **Fukuda, T., Fernandez, R., Rosenberg, A., Thomas, S., Ramabhadran, B., Sorin, A. and Kurata, G.** (2018) Data augmentation improves recognition of foreign accented speech, *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2018-Septe(September), pp. 2409–2413. doi: 10.21437/Interspeech.2018-1211.

[90] **Wei, S., Zou, S., Liao, F. and Lang, W**. (2020) A comparison on data augmentation methods based on deep learning for audio classification, *Journal of Physics: Conference Series*, 1453(1). doi: 10.1088/1742-6596/1453/1/012085.

[91] **Schlüter, J. and Grill, T.** (2015) Exploring data augmentation for improved singing voice detection with neural networks, *Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015*, pp. 121–126.

[92] **Sakai, A., Minoda, Y. and Morikawa, K.** (2017) Data augmentation methods for machine-learning-based classification of bio-signals, *10th Biomedical Engineering International Conference*, *BMEiCON 2017*, pp. 1–4. doi: 10.1109/BMEiCON.2017.8229109.

[93] **Yun, D. and Choi, S. H.** (2022) Deep learning-based estimation of reverberant environment for audio data augmentation, *Sensors*, 22(2), p. 592. doi: 10.3390/s22020592.

[94] **Ma, F., Li, Y., Ni, S., Huang, S.-L. and Zhang, L.** (2022) Data augmentation for audio-visual emotion recognition with an efficient multimodal conditional GAN, *Applied Sciences*, 12(1), p. 527. doi: 10.3390/app12010527.

[95] **Bahmei, B., Birmingham, E. and Arzanpour, S.** (2022) CNN-RNN and data augmentation using deep convolutional generative adversarial network for environmental sound classification, *IEEE Signal Processing Letters*, 29, pp. 682–686. doi: 10.1109/LSP.2022.3150258.

[96] **Jaitly, N. and Hinton, G. E.** (2013) Vocal Tract Length Perturbation (VTLP) improves speech recognition, *Proceedings of the 30 th International Conference on Machine Learning*, 90, pp. 42–51.

[97] **Nanni, L., Maguolo, G. and Paci, M.** (2020) Data augmentation approaches for improving animal audio classification, *Ecological Informatics*, 57(101084). doi: 10.1016/j.ecoinf.2020.101084.

[98] **Nanni, L., Maguolo, G., Brahnam, S. and Paci, M.** (2021) An ensemble of convolutional neural networks for audio classification, *Applied Sciences (Switzerland)*, 11(13). doi: 10.3390/app11135796.

[99] **Nam, H., Kim, S.-H. and Park, Y.-H.** (2022) Filteraugment: An acoustic environmental data augmentation method, in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 4308–4312. doi: 10.1109/ICASSP43922.2022.9747680.

[100] **Oldham, K. B. and Spanier, J.** (eds) (1974) *The Fractional Calculus Theory and Applications of Differentiation and Integration to Arbitrary Order*. 1st Editio. New York and London: Academic Press. doi: 10.1016/S0076-5392(09)60219-8.

[101] **Pu, Y. F., Zhou, J. L. and Yuan, X.** (2010) Fractional differential mask: A fractional differential-based approach for multiscale texture enhancement, *IEEE Transactions on Image Processing*, 19(2), pp. 491–511. doi: 10.1109/TIP.2009.2035980.

[102] **Chen, M., Pu, Y. F. and Bai, Y. C.** (2021) Low-dose CT image denoising using residual convolutional network with fractional TV loss, *Neurocomputing*. Elsevier B.V., 452, pp. 510–520. doi: 10.1016/j.neucom.2020.10.004.

[103] **Umesh, S., Cohen, L. and Nelson, D.** (2002) Frequency warping and the Mel scale, *IEEE Signal Processing Letters*, 9(3), pp. 104–107. doi: 10.1109/97.995829.

[104] **Diethelm, K.** (1997) An algorithm for the numerical solution of differential equations of fractional order, *Electronic Transactions on Numerical Analysis*, 5(January), pp. 1–6.

[105] **He, K., Zhang, X., Ren, S. and Sun, J.** (2016) Deep residual learning for image recognition, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[106] **Moorer, J. A.** (1977) Signal processing aspects of computer music: A survey, *Proceedings of the IEEE*. IEEE, 65(8), pp. 1108–1137. doi: 10.1109/PROC.1977.10660.

[107] **Serra, X. and Smith, J.** (1990) Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition, *Computer Music Journal*, 14(4), p. 12. doi: 10.2307/3680788.

[108] **de Poli, G.** (1983) A tutorial on digital sound synthesis techniques, *Computer Music Journal*, 7(4), p. 8. doi: 10.2307/3679529.

[109] **Chowning, J. M.** (1973) Synthesis of complex audio spectra by means of frequency modulation., *AES: Journal of the Audio Engineering Society*, 21(7), pp. 526–534.

[110] **Rabenstein, R. and Trautmann, L.** (2001) Digital sound synthesis by physical modelling, *International Symposium on Image and Signal Processing and Analysis, ISPA*, 2001, pp. 12–23. doi: 10.1109/ISPA.2001.938598.

[111] **Whorley, R. P. and Conklin, D.** (2016) Music generation from statistical models of harmony, *Journal of New Music Research*, 45(2), pp. 160–183. doi: 10.1080/09298215.2016.1173708.

[112] **Engel, J., Resnick, C., Roberts, A., Dieleman, S., Norouzi, M., Eck, D. and Simonyan, K.** (2017) Neural audio synthesis of musical notes with WaveNet autoencoders, *34th International Conference on Machine Learning, ICML 2017*, 3, pp. 1771–1780.

[113] **Briot, J. P. and Pachet, F.** (2020) Deep learning for music generation: challenges and directions, *Neural Computing and Applications*. Springer London, 32(4), pp. 981–993. doi: 10.1007/s00521-018-3813-6.

[114] **Wagner, P. *et al.*** (2019) Speech synthesis evaluation — state-of-the-art assessment and suggestion for a novel research program, in *10th ISCA Workshop on Speech Synthesis (SSW 10)*. *ISCA*, pp. 105–110. doi: 10.21437/SSW.2019-19.

[115] **Kaur, N. and Singh, P.** (2023) Conventional and contemporary approaches used in text to speech synthesis: a review, *Artificial Intelligence Review*. Springer Netherlands, 56(7), pp. 5837–5880. doi: 10.1007/s10462-022-10315-0.

[116] **Shi, Z.** (2021) A survey on audio synthesis and audio-visual multimodal processing. Available at: http://arxiv.org/abs/2108.00443.

[117] **Wang, Y., Skerry-Ryan, R.J., Stanton, D., Wu, Y., Weiss, R.J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S. and Le, Q.** (2017) Tacotron: Towards end-to-end speech synthesis, in *Interspeech 2017*. *ISCA*, pp. 4006–4010. doi: 10.21437/Interspeech.2017-1452.

[118] **Kawahara, H.** (2006) STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds, *Acoustical Science and Technology*, 27(6), pp. 349–353. doi: 10.1250/ast.27.349.

[119] **Dudley, H.** (1939) Remaking speech, *The Journal of the Acoustical Society of America*, 11(2), pp. 169–177. doi: 10.1121/1.1916020.

[120] **Gold, B. and Rader, C.** (1967) The channel vocoder, *IEEE Transactions on Audio and Electroacoustics*, 15(4), pp. 148–161. doi: 10.1109/TAU.1967.1161919.

[121] **Dolson, M.** (1986) The Phase vocoder: A tutorial, *Computer Music Journal*, 10(4), p. 14. doi: 10.2307/3680093.

[122] **Spanias, A. S.** (1994) Speech coding: a tutorial review, *Proceedings of the IEEE*, 82(10), pp. 1541–1582. doi: 10.1109/5.326413.

[123] **Quatieri, T. F.** (2002) *Discrete-Time Speech Signal Processing: Principles and Practice*. 1st edn. Prentice Hall.

[124] **Oord, A.V.D., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. and Kavukcuoglu, K.** (2016) WaveNet: A generative model for raw audio, pp. 1–15. Available at: http://arxiv.org/abs/1609.03499.

[125] **Griffin, D. and Lim, J.** (1983) Signal estimation from modified short-time Fourier transform, *IEEE International Conference on Acoustics, Speech, and Signal Processing*, *ICASSP '83*, IEEE, pp. 804–807. doi: 10.1109/ICASSP.1983.1172092.

[126] **Prusa, Z., Balazs, P. and Sondergaard, P. L.** (2017) A noniterative method for reconstruction of phase from STFT magnitude, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(5), pp. 1154–1164. doi: 10.1109/TASLP.2017.2678166.

[127] **Kleczkowski, P.** (1989) Group additive synthesis, *Computer Music Journal*, 13(1), p. 12. doi: 10.2307/3679851.

[128] **Krawczyk, M. and Gerkmann, T.** (2014) STFT phase reconstruction in voiced speech for an improved single-channel speech enhancement, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. IEEE, 22(12), pp. 1931–1940. doi: 10.1109/TASLP.2014.2354236.

[129] **Perraudin, N., Balazs, P. and Sondergaard, P. L.** (2013) A fast Griffin-Lim algorithm, in *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, pp. 1–4. doi: 10.1109/WASPAA.2013.6701851.

[130] **Le Roux, J., Kameoka, H., Ono, N. and Sagayama, S.** (2010) Fast signal reconstruction from magnitude STFT spectrogram based on spectrogram consistency, *Proceedings of the 13th International Conference on Digital Audio Effects (DAFx-10)*. Graz,Austria, pp. 397–403.

[131] **Masuyama, Y., Yatabe, K. and Oikawa, Y.** (2019) Griffin–Lim like phase recovery via Alternating Direction Method of Multipliers, *IEEE Signal Processing Letters*, 26(1), pp. 184–188. doi: 10.1109/LSP.2018.2884026.

[132] **Beauregard, G. T., Harish, M. and Wyse, L.** (2015) Single pass spectrogram inversion, in *2015 IEEE International Conference on Digital Signal Processing (DSP)*. IEEE, pp. 427–431. doi: 10.1109/ICDSP.2015.7251907.

[133] **Prusa, Z. and Søndergaard, P. L.** (2016) Real-time spectrogram inversion using phase gradient heap integration, *Proceedings of the 19th International Conference on Digital Audio Effects (DAFx-13)*. Brno, Czech Republic, pp. 17–21.

[134] **Kalchbrenner, N., Elsen, E., Simonyan, K., Noury, S., Casagrande, N., Lockhart, E., Stimberg, F., Oord, A., Dieleman, S. and Kavukcuoglu, K.** (2018) Efficient neural audio synthesis, *Proceedings of the 35th International Conference on Machine Learning*. Available at: http://arxiv.org/abs/1802.08435.

[135] **Valin, J. M. and Skoglund, J.** (2019) LPCNET: Improving neural speech synthesis through linear prediction, *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2019-May, pp. 5891–5895. doi: 10.1109/ICASSP.2019.8682804.

[136] **Masuyama, Y., Yatabe, K., Koizumi, Y., Oikawa, Y. and Harada, N.,** (2021) Deep Griffin–Lim Iteration: Trainable iterative phase reconstruction using neural network, *IEEE Journal of Selected Topics in Signal Processing*, 15(1), pp. 37–50. doi: 10.1109/JSTSP.2020.3034486.

[137] **Govalkar, P., Fischer, J., Zalkow, F. and Dittmar, C.** (2019) A comparison of recent neural vocoders for speech signal reconstruction, in *10th ISCA Workshop on Speech Synthesis (SSW 10)*. ISCA: ISCA, pp. 7–12. doi: 10.21437/SSW.2019-2.

[138] **Liu, Y.-W. and Smith, J. O.** (2007) Audio watermarking through deterministic plus stochastic signal decomposition, *EURASIP Journal on Information Security*, 2007, pp. 1–12. doi: 10.1155/2007/75961.

[139] **Puckette, M.** (1995) Phase-locked vocoder, *Proceedings of 1995 Workshop on Applications of Signal Processing to Audio and Accoustics*. IEEE, pp. 222–225. doi: 10.1109/ASPAA.1995.482995.

[140] **Le Roux, J., Ono, N. and Sagayama, S.** (2008) Explicit consistency constraints for STFT spectrograms and their application to phase reconstruction, in *ITRW on Statistical and Perceptual Audio Processing, SAPA 2008*, pp. 23–28.

[141] **Boyd, S., Parikh, N., Chu, E., Peleato, B. and Eckstein, J.** (2010) Distributed optimization and statistical learning via the Alternating Direction Method of Multipliers, *Foundations and Trends in Machine Learning*, 3(1), pp. 1–122. doi: 10.1561/2200000016.

[142] **Parikh, N. and Boyd, S.** (2014) Proximal algorithms, *Foundations and Trends in Optimization*, 1(3), pp. 127–239. doi: 10.1561/2400000003.

[143] **Tarasov, V. E. and Tarasova, V. V.** (2016) Long and short memory in economics: Fractional-order difference and differentiation, *IRA-International Journal of Management & Social Sciences (ISSN 2455-2267)*, 5(2), p. 327. doi: 10.21013/jmss.v5.n2.p10.

[144] **Geweke, J. and Porter-Hudak, S.** (1983) The estimation and application of long memory time series models, *Journal of Time Series Analysis*, 4(4), pp. 221–238. doi: 10.1111/j.1467-9892.1983.tb00371.x.

[145] **Garofolo, J. S., Lamel, L. F., Fisher, W. M., Fiscus, J. G. and Pallett, D. S.** (1993) *DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM. NIST speech disc 1-1.1.*

[146] **Loizou, P. C.** (2017) *Speech Enhancement: Theory and Practice*. Second Edition, CRC Press.

[147] **Donahue, C., McAuley, J. and Puckette, M.** (2019) Adversarial audio synthesis, in *International Conference on Learning Representations (ICLR) 2019*. New Orleans, Louisiana, United States, pp. 1–16. Available at: http://arxiv.org/abs/1802.04208.

[148] **Li, N., Liu, S., Liu, Y., Zhao, S. and Liu, M.** (2019) Neural speech synthesis with transformer network, *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01), pp. 6706–6713. doi: 10.1609/aaai.v33i01.33016706.

[149] **Albadawy, E. A., Gibiansky, A., He, Q., Wu, J., Chang, M.-C. and Lyu, S.** (2022) Vocbench: A neural vocoder benchmark for speech synthesis, in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 881–885. doi: 10.1109/ICASSP43922.2022.9746698.

[150] **Shen, J., Pang, R., Weiss, R.J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R. and Saurous, R.A.** (2018) Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 4779–4783. doi: 10.1109/ICASSP.2018.8461368.

[151] **Juvela, L., Bollepalli, B., Tsiaras, V. and Alku, P.** (2019) GlotNet—A raw waveform model for the glottal excitation in statistical parametric speech synthesis, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. IEEE, 27(6), pp. 1019–1030. doi: 10.1109/TASLP.2019.2906484.

[152] **Ping, W., Peng, K., Zhao, K. and Song, Z.** (2019) ‘WaveFlow: A compact flow-based model for raw audio’, pp. 7662–7672. Available at: http://arxiv.org/abs/1912.01219.

[153] **Song, E., Byun, K. and Kang, H.-G.** (2019) ExcitNet vocoder: A neural excitation model for parametric speech synthesis systems, in *27th European Signal Processing Conference (EUSIPCO)*. IEEE, pp. 1–5. doi: 10.23919/EUSIPCO.2019.8902701.

[154] **Airaksinen, M., Bollepalli, B., Juvela, L., Wu, Z., King, S. and Alku, P.** (2016) GlottDNN — A full-band glottal vocoder for statistical parametric speech synthesis, in *Interspeech 2016*. ISCA: ISCA, pp. 2473–2477. doi: 10.21437/Interspeech.2016-342.

[155] **Popov, V., Kudinov, M. and Sadekova, T.** (2020) Gaussian LPCNet for multisample speech synthesis, in *ICASSP 2020 - IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 6204–6208. doi: 10.1109/ICASSP40776.2020.9053337.

[156] **Hershey, J. R., Le Roux, J. and Weninger, F.** (2014) Deep unfolding: model-based inspiration of novel deep architectures, pp. 1–27. Available at: http://arxiv.org/abs/1409.2574.

[157] **Venkatakrishnan, S. V., Bouman, C. A. and Wohlberg, B.** (2013) Plug-and-Play priors for model based reconstruction, in *2013 IEEE Global Conference on Signal and Information Processing*. IEEE, pp. 945–948. doi: 10.1109/GlobalSIP.2013.6737048.

[158] **Trabelsi, C., Bilaniuk, O., Ying, Z., Serdyuk, D., Subramanian, S , Santos, J. F., Mehri, S., Rostamzadeh, N., Bengio, Y. and Pal, C.** (2018) Deep complex networks, *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, (2016), pp. 1–19.

CURRICULUM VITAE

Name Surname **:** Bilgi Görkem YAZGAÇ

EDUCATION **:**

- **B.Sc.** **:** 2014, Istanbul Technical University, Faculty of Electrical and Electronics Engineering, , Department of Electronics Engineering
- **M.Sc.** **:** 2017, Istanbul Technical University, Science Engineering and Technology Institute, Electronics Engineering Program

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2015-2023 Istanbul Technical University, Türkiye, Research Assistant at Department of Electronics and Communication Engineering.

  Conferences:

  - AgroGeoinformatics

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Yazgaç, B. G.** and Kırcı, M. (2022) Fractional-order calculus-based data augmentation methods for environmental sound classification with deep learning', *Fractal and Fractional*, 6(10). doi: 10.3390/fractalfract6100555.
- **Yazgaç, B. G.** and Kırcı, M. (2021) Fractional differential equation-based instantaneous frequency estimation for signal reconstruction, *Fractal and Fractional*, 5(3). doi: 10.3390/fractalfract5030083.

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:

- **Yazgac, B. G.** and Kirci, M. (2019) Fractional order calculus based fruit detection, *8th International Conference on Agro-Geoinformatics, Agro-Geoinformatics 2019*, pp. 14–17. doi: 10.1109/Agro-Geoinformatics.2019.8820545.
- **Yazgac, B. G.,** Durmus, H., Kirci, M., Gunes, E. O. and Karli, H. B (2019) Petri nets based procedure of hardware/software codesign of an urban agriculture monitoring system, *8th International Conference on Agro-Geoinformatics, Agro-Geoinformatics 2019*. doi: 10.1109/Agro-Geoinformatics.2019.8820255.
- **Yazgac, B. G.** and Kirci, M. (2017) Embedded system application for sunn pest detection, *6th International Conference on Agro-Geoinformatics, Agro-Geoinformatics 2017*. doi: 10.1109/Agro-Geoinformatics.2017.8047027.

- **Yazgaç, B. G.** and Kırcı, M. (2016) Detection of sunn pests using sound signal processing methods, in Agro-Geoinformatics (Agro-Geoinformatics), *5th International Conference on Agro-Geoinformatics, Agro-Geoinformatics 2016*. Tianjin, China: IEEE. doi: 10.1109/Agro-Geoinformatics.2016.7577694.
- Aygun, S., **Yazgac, B. G.,** Kirci, M. and Gunes, E. O. (2016) Design of an embedded system for automatically determination of the varieties of wheat seeds grown in Turkey based on image analysis, *5th International Conference on Agro-Geoinformatics, Agro-Geoinformatics 2016*. doi: 10.1109/Agro-Geoinformatics.2016.7577679.