

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YAZILIM TABANLI TELSİZ HABERLEŞME SİSTEMLERİNDE DERİN
ÖĞRENME YAKLAŞIMI İLE RADYO SİNYALLERİNİN
SINIFLANDIRILMASI

Adem GÜL

YÜKSEK LİSANS TEZİ

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Haberleşme Programı

Danışman

Doç. Dr. Bülent BOLAT

Haziran, 2022

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YAZILIM TABANLI TELSİZ HABERLEŞME SİSTEMLERİNDE DERİN
ÖĞRENME YAKLAŞIMI İLE RADYO SİNYALLERİNİN
SINIFLANDIRILMASI

Adem GÜL tarafından hazırlanan tez çalışması 14.06.2022 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Haberleşme Mühendisliği Anabilim Dalı, Haberleşme Programı **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Doç. Dr. Bülent BOLAT
Yıldız Teknik Üniversitesi
Danışman

Jüri Üyeleri

Doç. Dr. Bülent BOLAT, Danışman
Yıldız Teknik Üniversitesi

Prof. Dr. Mehmet Serdar Ufuk TÜRELİ, Üye
Yıldız Teknik Üniversitesi

Dr. Öğr. Üyesi Gökalg TULUM, Üye
Nişantaşı Üniversitesi

Danışmanım Doç. Dr. Bülent BOLAT sorumluluğunda tarafımca hazırlanan “Yazılım Tabanlı Telsiz Haberleşme Sistemlerinde Derin Öğrenme Yaklaşımı ile Radyo Sinyallerinin Sınıflandırılması” başlıklı çalışmada veri toplama ve veri kullanımında gerekli yasal izinleri aldığımı, diğer kaynaklardan aldığım bilgileri ana metin ve referanslarda eksiksiz gösterdiğimi, araştırma verilerine ve sonuçlarına ilişkin çarpıtma ve/veya sahtecilik yapmadığımı, çalışmam süresince bilimsel araştırma ve etik ilkelerine uygun davrandığımı beyan ederim. Beyanımın aksinin ispatı halinde her türlü yasal sonucu kabul ederim.

Adem GÜL

İmza



Bu alıřma, Yıldız Teknik Üniversitesi Bilimsel Arařtırma Proje Koordinatörlüğü'nün FYL-2021-4368 numaralı projesi ile desteklenmiřtir.



Aileme
ve
biricik eşime

TEŞEKKÜR

Yüksek Lisans eğitimim ve nihai çalışmam olan tez sürecinde yardımlarını ve desteklerini esirgemeyen, sürecin işleyişi ve takibi konusunda her daim aktif olarak yol almamı sağlayan değerli Hocam Doç. Dr. Bülent BOLAT'a,

Tüm hayatım boyunca yaptıkları fedakarlıklarla ve maddi/manevi desteklerini esirgmeden arkamda duran sevgili annem Adile GÜL, babam Rasim GÜL ve kardeşim Berkan GÜL'e,

Meslek ve eğitim hayatımı içeren kariyerim de dahil olmak üzere yaşamımın her alanında, Hukukçu olmasına rağmen mühendislik kavramlarını duymaktan sıkılmadan beni her zaman dinleyerek ve destekleyerek başarıya ulaşmamı sağlayan biricik eşim Merve GÜL'e,

Güçlü kişiliğiyle ve kariyerindeki başarılarıyla bana yol gösteren ve örnek teşkil eden çok değerli yöneticim ve abim Erdinç ÖZTÜRK'e,

Değerli dostlarım, yakın arkadaşlarım Nureddin YILDIRIM ve Okan SARAÇOĞLU'na,

En içten teşekkürlerimi sunarım.

Adem GÜL

İÇİNDEKİLER

SİMGE LİSTESİ	vii
KISALTMA LİSTESİ	viii
ŞEKİL LİSTESİ	ix
TABLO LİSTESİ	x
ÖZET	xi
ABSTRACT	xiii
1 GİRİŞ	1
1.1 Literatür Özeti.....	1
1.2 Tezin Amacı ve Önemi	3
1.3 Hipotez	3
2 YÖNTEM	5
2.1 Derin Öğrenme	5
2.1.1 Evrişimli Sinir Ağları (CNN).....	7
2.1.2 Yinelemeli Sinir Ağları (RNN) ve LSTM Ağları.....	9
2.2 Yazılım Tabanlı Radyo	13
2.3 Modülasyon Türleri.....	17
3 UYGULAMA	23
3.1 Uygulama Senaryosu.....	23
3.2 Veri Setinin Toplanması	24
3.3 Derin Öğrenme için Ön İşlem.....	31
3.4 Verinin Sınıflandırılması	32
3.5 Ağların Başarımları	36
4 SONUÇ VE ÖNERİLER	41
KAYNAKÇA	43
TEZDEN ÜRETİLMİŞ YAYINLAR	46

SİMGE LİSTESİ

p	Olasılık fonksiyonu
X_{-}	Giriş veri boyutu
U_{-}	Filtre boyutu
Y_{-}	Çıktı boyutu
N_{-}	Tekrarlanan çerçeve sayısı
θ_t	t 'ninci iterasyonda ağın tüm ağırlıkları (vektör)
$\Delta(\theta)_t$	Öğrenme süreci değeri
E	Geri yayılım maliyet fonksiyonu
α	Öğrenme oranı
W	Giriş ile nöron arasındaki ağırlıklar
U	Geri besleme ağırlıkları
E_i	Karesel hata
f_c	Taşıyıcı sinyal frekansı
$s(t)$	Modüle sinyal
ϕ_m	Modülasyonda kullanılan faz değeri
a_k	İletilen sembol
$h(t)$	Gauss darbe yanıtı fonksiyonu
$n(t)$	Gürültü fonksiyonu
$f(\tau)$	Saat yenilemede faz düzeltme miktarı
z_k	Zamanlama fonksiyonu
$z^{(i)}$	Softmax fonksiyonunda eğitim seti
$X^{(t)}$	ConvLSTM ağ girişleri
$C^{(t)}$	ConvLSTM ağdaki hücre çıkış durumları
$h^{(t)}$	ConvLSTM ağında gizli katman durumları

KISALTMA LİSTESİ

ASK	Amplitude Shift Keying (Genlik Kaydırmalı Anahtarlama)
BB	Baseband (Temel Bant)
BPTT	Back-Propagation Through Time (Zaman İçinde Geri Yayılım)
CNN	Convolutional Neural Network (Evrişimsel Sinir Ağı)
dB	Desibel
FCNs	Fully Connected Neural Networks (Tamamen Bağlı Sinir Ağları)
FSK	Frequency Shift Keying (Frekans Kaydırmalı Anahtarlama)
IF	Intermediate Frequency (Orta Frekans)
IQ	In-Quadrature (Karelemeye)
kNN	k-Nearest Neighbors (k-En Yakın Komşular)
LSTM	Long-Short Term Memory (Uzun-Kısa Dönem Hafıza)
MCS	Modulation and Coding Scheme (Modülasyon ve Kodlama Şeması)
OOK	On-Off Keying (Aç-Kapa Anahtarlama)
PSA	Phase Sensitive Amplifier (Faz Hassasiyetli Yükselteç)
PSK	Phase Shift Keying (Faz Kaydırmalı Anahtarlama)
PSF	Pulse Shaping Filter (Darbe Şekillendirme Filtresi)
QAM	Quadrature Amplitude Modulation (Karesel Genlik Modülasyonu)
RF	Radio Frequency (Radyo Frekans)
RNN	Recurrent Neural Network (Yinelemeli Sinir Ağı)
SDR	Software Defined Radio (Yazılım Tabanlı Radyo)
SVN	Support Vector Machine (Destek Vektör Makinesi)
SNR	Signal to Noise Ratio (Sinyal Gürültü Oranı)
VHF	Very High Frequency (Çok Yüksek Frekans)
UHF	Ultra High Frequency (Ultra Yüksek Frekans)
YSA	Yapay Sinir Ağları

ŞEKİL LİSTESİ

Şekil 2.1 Yapay Sinir Ağları katman yapısı.....	6
Şekil 2.2 Derin sinir ağları	6
Şekil 2.3 CNN Ağları gösterimi	9
Şekil 2.4 Yinelemeli Ağ – RNN modeli gösterimi	10
Şekil 2.5 LSTM blok şeması	12
Şekil 2.6 LSTM nöronlarının güncellenmesi.....	13
Şekil 2.7 Temel bir SDR uygulamasının blok diyagramı.....	13
Şekil 2.8 HackRF One blok diyagramı.....	14
Şekil 2.9 GNU Radio FM Radyo Alıcı blok diyagramı.....	15
Şekil 2.10 Uygulama sistem blok diyagramı.....	16
Şekil 2.11 Modülasyon türleri.....	18
Şekil 2.12 Sinyal örnekleme işlemi ve I/Q demodülasyonu	19
Şekil 2.13 FM odülasyon süreci	20
Şekil 2.14 ASK/FSK modülasyon süreci	21
Şekil 2.15 QAM modülasyon süreci	22
Şekil 3.1 Çalışmada kullanılan sınıflandırma modeli	24
Şekil 3.2 Sistemin veri haberleşmesinin temsili gösterimi	25
Şekil 3.3 Uygulamada kullanılan GNU Radio şematığı.....	26
Şekil 3.4 PSF filtresi Gauss darbe yanıtı.....	27
Şekil 3.5 Frekans spektrumunda pasif durum gözlemlenmesi sonucu.....	29
Şekil 3.6 Frekans spektrumunda aktivite olması durumu.....	29
Şekil 3.7 GNU Radio WX GUI Constellation bloğu sonuçları (aktif spektrum) ...	31
Şekil 3.8 Tasarlanan CNN Modelinin Eğitim Algoritması	33
Şekil 3.9 ConvLSTM hücre mimarisi	35
Şekil 3.10 Model-1 CNN yapısının eğitime ve doğrulama başarımlar eğrisi	37
Şekil 3.11 Model-2 CNN yapısının eğitime ve doğrulama başarımlar eğrisi	37
Şekil 3.12 ConvLSTM Sinir Ağı Modeli'nin başarımlar çıktısı.....	38
Şekil 3.13 ConvLSTM Sinir Ağı başarımlar oranı düşük modellerin başarımlar çıktısı...38	

TABLO LİSTESİ

Tablo 2.1 CNN Modeli katman sayısı hesaplama yöntemi.....	9
Tablo 2.2 Veri toplamada kullanılan anten modelleri ve özellikleri	14
Tablo 2.3 Uygulama frekanslarında sık kullanılan modülasyon türleri.....	18
Tablo 3.1 Algılanan frekanslar ve sınıf gösterimleri	36
Tablo 3.2 CNN ağ modellerinin sonuçları	39
Tablo 3.3 ConvLSTM ağ modellerinin sonuçları	40



Yazılım Tabanlı Telsiz Haberleşme Sistemlerinde Derin Öğrenme Yaklaşımı İle Radyo Sinyallerinin Sınıflandırılması

Adem GÜL

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Danışman: Doç.Dr. Bülent BOLAT

Günümüzde kablosuz frekans bandının kullanımının yaygınlaşmasıyla birlikte birçok güvenlik açığı ortaya çıkmıştır. Bu durum çeşitli güvenlik önlemlerinin araştırılması ve geliştirilmesini bir zorunluluk olarak getirmektedir. Özellikle ulusal güvenlikten sorumlu devlet kurumlarının bu yöndeki ihtiyaçları her geçen gün değişmekte ve artmaktadır. Bu nedenle ortamdaki kablosuz sinyallerin ve vericilerin doğru bir şekilde tanımlanması gereksinimi önem kazanmaktadır. Bu sinyal tanımlama sürecinin doğrudan uç cihazlarda verimli bir şekilde gerçekleştirilmesi, işlemin hızlı olması için de gerekli hale gelmiştir. Bu projenin amacı, yazılım tabanlı radyo cihazlarını (SDR) kullanarak kablosuz sinyallerin sınıflandırılmasıdır. En iyi sınıflandırma başarısını elde etmek için derin öğrenme yöntemleri kullanılmış ve başarımları kıyaslanmıştır. Önerilen yöntemin gerçek dünyada uygulanabilir olduğunu göstermek amacıyla radyo sinyalleri yazılım tabanlı radyo platformları kullanılarak üretilmiştir. Verici cihazlardan gönderilen sinyaller alıcı cihaz tarafından I/Q veri formatında kaydedilmiş ve derin öğrenme algoritmaları ile yapılan sınıflandırmanın başarımı araştırılmıştır. Çalışmanın deneysel bir sonucu olarak, özellikle görüntü işleme ve obje tanımlama

uygulamalarında verimli sonuçları olan CNN modelinin radyo sinyallerinin tespitinde %80'e varan başarımlar sunduğu görülmüştür. Bunun yanı sıra alternatif ve modern bir yöntem olarak kullanılan LSTM ağı modeli CNN'e göre çok daha düşük seviyelerde başarımlar göstermiştir. LSTM modeli tek başına bu uygulamada tamamen başarısız olduğundan hibrit bir model olarak ConvLSTM modeli kullanılmış ve daha başarılı sonuçlar elde edilmiştir. İki farklı model yapısı arasındaki karşılaştırmanın yanı sıra başarı oranı yüksek olan CNN ağı modelinde katman sayısındaki artışın ağın başarımlarını ile doğrudan ilişkili olmadığı da pratik bir sonuç olarak görülmüştür.

Anahtar Kelimeler: Yazılım Tabanlı Radyo, Derin Öğrenme, Yapay Sinir Ağları, IQ Diagramları, Spektrum Analizi, Kablosuz Ağ Güvenliği

The Classification of Radio Signals with Deep Learning Approach in Software Based Wireless Communication Systems

Adem GÜL

Department of Electronics and Communication Engineering

Master Thesis

Supervisor: Assoc. Prof. Dr. Bülent BOLAT

The appearance of many security vulnerabilities is also increasing today with the widespread use of the wireless frequency band. This situation brings with it the research and development of various security measures as a necessity. Identification of wireless signals and transmitters properly in the environment becomes a crucial situation. The efficient achievement of this process directly on the end devices also becomes necessary for the process to be fast. The purpose of this paper is to show the performance of deep learning methods for classification of wireless signals by making use of the software-defined radio devices. In particular, it is aimed to implement real-world applications and evaluate the results by utilizing software-based radio platforms. We investigated the performance of the classification made by deep learning algorithms on the I/Q data samples as a wireless signal representation that obtained on the receiver device by collected from transmitter devices. As an experimental result of the study, it has been seen that successful results are seen up to %80 in the detection of radio signals of the CNN model , which is especially efficient in image processing and identification applications. On the other hand, the LSTM network

model, which is used as an alternative and modern method, has shown much less success than CNN with %30 learning success ratio. In addition to the comparison in terms of two different models, the one with a rich experience in the layer of the network model with a high success rate has resulted in a practical way.

Keywords: Software Defined Radio, Deep Learning, Artificial Neural Network, IQ Constellation, Spectrum Analysis, Wireless Network Security



1.1 Literatür Özeti

Günümüzde kablosuz ağlara olan ihtiyaç ve kablosuz spektrum kullanımı hızla artmaktadır. Ancak yaygın olarak kullanılan kablosuz cihazların teknolojik altyapıları da birbirinden farklıdır. Kablosuz iletişim sistemlerinin davranışı incelendiğinde, çeşitli yasal veya yasadışı senaryolarla karşılaşmak mümkündür. Özellikle siber güvenlik uygulamalarına ihtiyaç duyan devlet kurumlarının sinyal istihbaratına olan ihtiyacının yanı sıra sivil-kablosuz haberleşme alanında uygulama yapan operatörler ve diğer kurum ve toplulukların da ihtiyaçları gün geçtikçe artmaktadır.

Kablosuz spektrumda aktif olan sinyaller zamanla değişebilir, bazı sinyallerin özellikleri tespit edilemediğinden sınıflandırılmayabilir, sinyal tekrarlayıcı görevi gören bozucuların olduğu yerlerde de alıcılar bazı tekrarlanan sinyaller tarafından yanlış yönlendirilebilir. Bu vericilerin türleri, eşzamanlı iletimlerin neden olduğu parazit nedeniyle çakışabilir. Bu nedenle öncelikle frekans spektrumunda sinyallerin spektrumdaki konumunun ve hangi frekansta yayın yaptıklarının doğru ve hızlı bir şekilde tespit edilmesi kritik öneme sahiptir. Bu projede bir denetimli derin öğrenme metodu kullanılarak çeşitli kablosuz sinyallerin sınıflandırılması ve başarımların analizi üzerine bir çalışmanın süreçlerinden ve sonuçlarından bahsedilmiştir.

Kablosuz sinyal sınıflandırma çalışmalarında geleneksel makine öğrenmesi yöntemlerinin, istatistiksel yöntemlerin veya bunların kombinasyon yöntemlerinin (SVM, k-NN vb.) tasarım aşamasında ciddi düzeyde mühendislik bilgisi ve/veya disiplinler arası karmaşık uzmanlık gerektirdiği önceki çalışmalarda kanıtlanmıştır [1]. Derin öğrenme yöntemlerinden biri olan Evrişimli Sinir Ağları'nın (CNN) I/Q verileriyle yapılan eğitim sonucunda elde ettiği başarımın geleneksel yöntemlere göre önemli ölçüde daha iyi olduğu [1]'deki çalışma ile de ortaya konmuştur. CNN yapısının tercih edilmesinin bir diğer nedeni, görüntü tanıma ve konuşma tanıma

gibi çok boyutlu verilerin işlendiği uygulamalardaki başarısıdır [2]. Ayrıca, [36]'daki çalışmada, giriş verisi olarak frekans verilerini kullanan eğitilmiş bir ağ (CNN), lisanssız frekans bantlarında girişim tanımlaması için bir yöntem önermektedir.

Geçmiş çalışmalar [10][11][12][13][14][32] gözden geçirildiğinde, kablosuz frekans tespiti için kullanılan geleneksel yöntemlerin karmaşık ve manuel özellikler kullanılarak gerçekleştirildiği anlaşılmaktadır. 2012 yılında Krizhevsky, Sutskever ve Hinton [2] tarafından yeniden canlandırıldığından beri, derin öğrenme algoritmalarının karmaşık veri ve davranışlar üzerindeki analiz yeteneklerinin çoğu durumda geleneksel sınıflandırma yöntemlerinden daha avantajlı olduğu bulunmuştur. Derin öğrenme yöntemleriyle tasarlanan sinir ağları, eski kısıtlı sinir ağı modellerinden daha üstündür. Ayrıca, gerçekleştirilen uygulama türü için özel ihtiyaçlara gerek duymadan karmaşık özelliklerin çıkarılmasında da rol oynar [16][17][18][37]. Derin öğrenme yaklaşımının kullanıldığı bu yenilikçi sınıflandırma yöntemi sayesinde kablosuz spektrumun kullanıldığı ortamlarda iletişim uygulamaları açısından çevre bilincinin artacağı öngörülmektedir. Bu durumun gerekliliği daha önce bahsedilen nedenlerden dolayı da önem kazanmaktadır. Ayrıca bu yaklaşım sayesinde sadece frekans spektrumundaki durumu algılamakla kalmayıp modülasyon tanımlama gibi işlevler de gerçekleştirilebilmektedir. Bu şekilde, spektrum kullanımını izlemek ve vericilerin davranışlarını buna göre yeniden düzenlemek mümkündür. [1], derin öğrenme yöntemlerinden CNN'yi kullanarak fiziksel katmandaki farklı modülasyon türlerini tanımlar ve sınıflandırır. Ancak bu çalışmada bir cihazın frekans spektrumunu tanımlamak yerine sadece modülasyon tipi belirlenmiştir. [11], SVM ve kNN algoritmalarını kullanır ve modülasyona özgü hatalara dayalı olarak ve %99 doğru sonuçlarla Wi-Fi cihazları için parmak izlerini tanımlar. Bu çalışmalar değerlendirildiğinde, en temel problemin sinyali en doğru modelleyecek olan öznitelik kümesini seçmek olduğu görülmektedir. Bu çalışmada, bu zorluğa bir alternatif çözüm olarak derin ağlar önerilmektedir. Öznitelik seçimini öğrenme sürecinin bir parçası haline getiren derin ağların kullanılması ile bu problemin üstesinden gelinebileceği düşünülmüştür.

1.2 Tezin Amacı ve Önemi

Bu çalışmada derin öğrenme yöntemlerinden CNN [19] ve LSTM [27] yönteminin SDR platformu üzerinde uygulaması gerçekleştirilmiştir. Modelin eğitimi sırasında denetimli öğrenme kullanılmıştır. Derin öğrenme, denetimli öğrenme yaklaşımı için güçlü bir altyapı sağlar. Ağdaki katman sayısını ve dolayısıyla ağın derinliğini büyük ölçüde artırarak karmaşık yapıları ve işlevleri öğrenebilir. Ayrıca bu derin ağın öğrenme katsayısı, büyük veri kümelerinden yararlanılarak artar. Karmaşık sayı türünde elde edilen I/Q verileri, bu yöntemin girdi verilerini farklı boyutlarda işleyebilme özelliğinden yararlanılarak işlenir. Varsayımsal veya simülasyon analizleri yerine gerçek dünya uygulamaları referans alınarak farklı tipte kablosuz vericilerden dinleme ve farklı karakteristik yapıları temsil eden I/Q örnekleri toplanmıştır. Bu girdi veri örneklerini toplamak için endüstriyel SDR platformlarından biri olan HACKRF kartı kullanılmıştır. Bu kartın kullanılma sebepleri arasında dinleme aralığının geniş bir spektrumda olması (1MHz-6GHz), örnekleme hızının yüksek olması, sinyal işleme kabiliyeti ve kullanılan arayüz programlarının iyi sonuç vermesi sayılabilir. Veri toplama, toplanan I/Q örneklerinin miktarı mümkün olduğu kadar yüksek olacak şekilde gerçekleştirilmiş ve ağın eğitiminde kullanılmıştır. Yapay sinir ağını eğitirken overfitting yapmamak için ağın hiperparametreleri sürekli güncellenmekte ve eğitim süreci en başarılı şekilde gerçekleştirilmektedir. Bu uygulamada başarıyı incelenen metot sayesinde güvenli bir kablosuz iletişim ortamı sağlamak amaçlanmıştır. Frekans türleri çeşitlendirilerek bu kapsamı artırılabilceği de öngörülmektedir.

1.3 Hipotez

Bu çalışmanın temelinde yer alan hipotez, demodülasyon süreçleri için gereken zaman ve işlem maliyetinin yüksekliği göz önünde bulundurularak bu süreçlere gerek kalmadan bir çözüm ortaya koymaktır. Ayrıca modülasyon işlemlerinde dikkat edilmesi gereken bir diğer husus da modülasyon sırasında kullanılan kodlama yöntemlerinin bilindik yöntemler dışında olması durumunda zaten demodülasyon işleminin başarılı bir şekilde gerçekleştirilemeyeceğidir. Çeşitli çalışmalarda da fiziksel katman güvenliğinin sağlanabilmesi adına kullanılan

adaptif modülasyon yöntemleri sunulmaktadır [28][29]. Spektrum ve sinyal modelinin algılanma sürecinin hızlandırılması adına bu çalışmada temel bant seviyesinde sinyal işleme süreçleri gerçekleştirilmeden I ve Q verilerinin değerleri 2 boyutlu bir şekilde ele alınarak sinyalin karakterini temsil eden ilk dijital verilerin örüntüsel olarak analiz edilmesi ve sinyallerin sınıflandırılması yöntemi sunulmuştur.

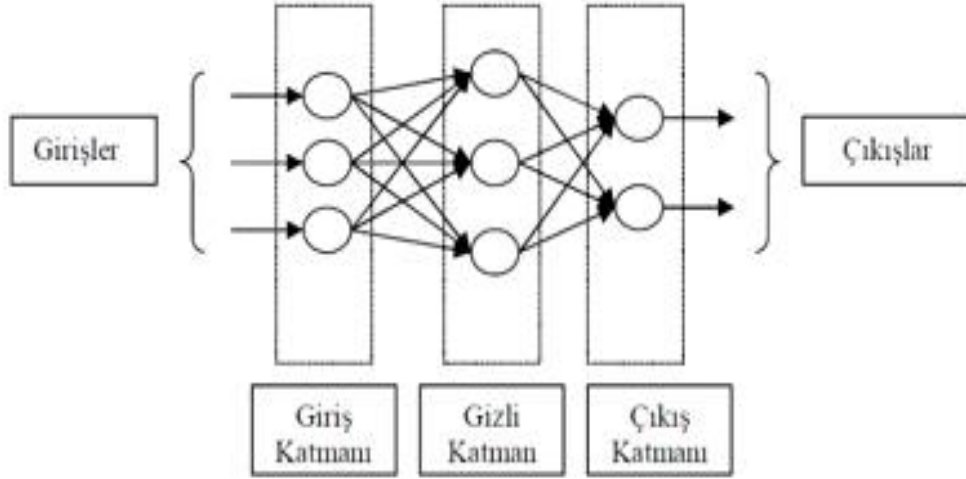


2.1 Derin Öğrenme

Bu çalışmada derin öğrenme yöntemleri kullanılarak çeşitli kablosuz haberleşme sinyallerinin sınıflandırılması üzerinde durulmuştur. İşlenen sinyaller farklı kablosuz vericilerden üretilmiş ve bir SDR cihazı ile I/Q formatında kaydedildikten sonra sınıflandırılmıştır. Sınıflandırma için Evrişimli Sinir Ağı (CNN) ve Uzun Kısa Süreli Bellek (LSTM) yapıları kullanılmıştır.

Derin öğrenmede asıl hedeflenen şey, bir problemin çözümüne yönelik senaryonun veya probleminden bağımsız olarak bir senaryonun öğrenilmesine yönelik kurgunun tasarlanması süreçlerini bilgisayar modelinin yapabilmesi için imkan sağlamaktır. Bilgisayar modeli bu avantajı kullanarak alternatif senaryolar karşısında da çözüm üretebilecektir. Bu öğrenme yönteminde bilgisayara yüklenen büyük veri setleri ve bunların katmanlar halinde tasarlanan öğrenme yapısından geçirilmesi senaryosu düşünüldüğünde insan beynindeki etkileşim halinde olan kalabalık sinir hücreleri akla getirilebilir. Bir örnek vermek gerekirse bir bilgisayara araba, kırmızı, beyaz, pantolon, kedi, çocuk kavramları giriş verisi olarak tanıtılıp, “Kırmızı pantolonlu çocuk ile beyaz kedi arabanın yanında oyun oynuyor” yorumu bilgisayar tarafından çıktı olarak elde edilebilir. Bu basit örneğin ötesinde özellikle yüz ve ses tanıma uygulamalarında derin öğrenmenin rolü günümüzde çok büyüktür. Bir yapay zekayı derin öğrenme ile eğitirken hem denetimli hem de denetimsiz öğrenme yöntemleri kullanılabilir. Bununla birlikte Derin Öğrenme kavramı, Derin Sinir Ağları olarak da anılır. Yapay Sinir Ağları (YSA) biyolojik nöronların basitleştirilmiş bir modelinin işlevsel bir taklididir. Bunların amacı gerçek dünya problemlerine yönelik çözümler üreten makinelerin geliştirilmesidir [3]. YSA’lar insan beynindekine benzer nöronlara sahiptir ve bu nöronların hepsi bir şekilde birbirine bağlıdır. Şekil 2.1’de görüldüğü üzere yapay sinir ağları üç temel katmandan oluşurlar. Bunlardan kısaca bahsetmek gerekirse, giriş katmanı; giriş verilerinin alındığı ve ağı beslediği katmandır. Gizli katman;

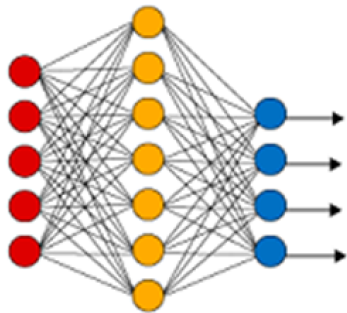
giriş verilerinin matematiksel hesaplamalara tabi tutulduğu ve bu hesaplamalar sonucunda çıktıya giden süreçte hangi nöronların aktif edip edilmeyeceği kararının verildiği katmandır. Çıktı katmanı ise ağıın çıktıların oluşturulduğu katmandır.



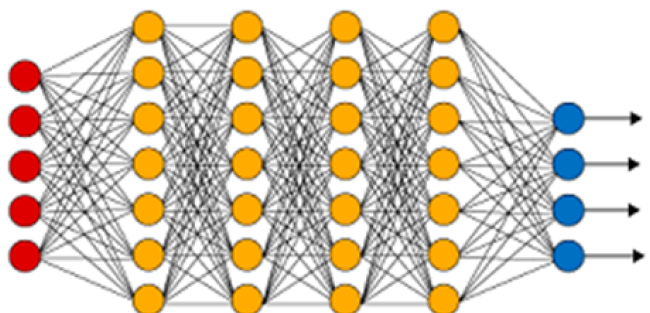
Şekil 2.1 Yapay Sinir Ağları katman yapısı

Derin Öğrenme Sinir Ağları'nda çok sayıda gizli katman kullanılmasından ötürü bu öğrenme çeşidi “Derin” olarak anılmaktadır (Şekil 2.2). Ağıın tasarımı aşamasında önemli olan işlemlerden biri de gizli katmanların ve her gizli katmanda yer alan nöronların sayısının belirlenmesidir [4].

Temel Sinir Ağları Gösterimi



Derin Öğrenme Sinir Ağları Gösterimi



● Giriş Katmanı ● Gizli Katman ● Çıkış Katmanı

Şekil 2.2 Derin sinir ağları [15]

Son yıllarda makine öğrenmesi uygulamalarında kullanılan algoritma ve araçlarda artış olmuş ve çeşitli ücretsiz kütüphaneler yaygınlaşmıştır [33]. GPU

mimarilerinden yararlanılarak bu yazılım kitaplıklarının hızlı bir şekilde çalıştırılması ve eğitim uygulamalarının hayata geçirilmesi kolaylaşmıştır. Keras-Tensorflow, Theano, Torch, MXNet, bu kitaplık ve araçlardan bazıları olup aktif olarak kullanılmaya ve desteklenmeye devam etmektedir. Bu araçlar sayesinde makine öğrenmesi veya derin öğrenme uygulamalarında ağı parametrelerini güncellemek ve tasarımı esnek bir şekilde değiştirmek mümkündür. [34]'de bu makine öğrenme araçlarının başarımlarını analizleri FCNs, CNN ve RNN ağı eğitilerek özellikle eğitim süreleri açısından ve bulut sistemlerde gerçekleştirilmiştir. [35]'de ise bu benzer şekilde farklı makine öğrenme araçları (Caffee, Mxnet, Pytorch, Tensorflow) bulut sistemler yerine uç cihazlarda gerçekleştirilmiştir. Her iki çalışma da farklı platformlarda olmak üzere kapsamlı nitelikte bu araçların başarımlarını teşhis etme amacıyla tamamlanmıştır.

Ağın başarımlarını artırmanın yollarından biri olarak da kütüphanenin desteklediği ölçüde çalışmalar yapılabilir. Gradyan hızlandırma ve derin öğrenme 2016 ve 2017 yılları ile birlikte makine öğrenmesi uygulamalarında geniş kullanım alanına sahip olmuştur. Yapısal verilerde gradyan hızlandırma ve görüntü tanıma gibi bilişsel problemlerde daha çok derin öğrenme tercih edilmektedir. Eski metotları tercih edenler programlama dili olarak XGBoost kütüphanesini ve Python ile R'yi tercih ederken; modern ve yaygın kullanıma bakıldığında kullanım kolaylığı sebebiyle ve farklı uygulama alanlarına uygunluğu nedeniyle Google tarafından geliştirilmiş olan Keras-Tensorflow kütüphanesi [6] ve Python programlama dili kullanılmaktadır. Bu çalışmada, CNN ve LSTM modelleri Keras-Tensorflow kütüphanesi kullanılarak gerçekleştirilmiştir. Hızlı tasarım kabiliyeti ve diğer araçlara göre kullanıcı dostu arayüzü nedeniyle bu çalışmada Keras-Tensorflow kütüphanesi tercih edilmiştir.

2.1.1 Evrişimli Sinir Ağları (CNN)

Evrişimli Sinir Ağları (CNN) iki boyutlu bir giriş sinyali, sinyali oluşturan yönlere/nesnelere ayıran ve bu basit parçacıklara önem dereceleri atayabilen bir derin öğrenme algoritmasıdır. Bir CNN modelinde gereken veri ön işleme süresi, diğer sınıflandırma algoritmalarına kıyasla çok daha kısadır. Filtreler ilkel

yöntemlerle elle tasarlanırken, CNN modelleri yeterince eğitilirse bu filtreleri/özellikleri öğrenebilme kabiliyetine de sahiptir.

Giriş verileri, görünür katmanda sunulan, tensör adı verilen çok boyutlu bir veri dizisidir. Bu çalışmada giriş verileri 2 boyutlu (2B) tensör adı verilen 3 boyutlu (3B) matris tipi verilerdir. Sunulan CNN ağ modelindeki temel işlevler ve bloklar Şekil 2.3'te gösterilmiştir. Katmanların işlevleri ise şu şekildedir.

Evrişim Katmanı: Bu katman, veri setindeki özellikleri algılamak için kullanılır. Bu katmanda, giriş verisi olarak kabul edilen giriş veri çerçevesinden düşük ve yüksek seviyeli özellikleri çıkarmak için görüntüye bir tür filtre uygulanır. Örneğin, bu filtre bir görüntüdeki kenarları algılamak için olabilir. Bu filtreler genellikle çok boyutludur. Bu çalışmadaki filtremiz de 3x3 boyutlarındadır. Filtre, veri setine uygulanırken giriş matrisinin sol üst köşesine konumlandırılır ve giriş matrisi ile filtre matrisleri arasındaki indeksler çarpılarak tüm sonuçlar toplanır. Son olarak, bu sonuçlar çıktı matrisinde saklanır. Daha sonra bu filtre bir piksel kaydırılır ve işlem tekrarlanır. Bu katmanın çalışması sonucunda elde edilen çıktı matrisi genellikle “Özellik Haritası” olarak bilinir.

Max-Pooling Katmanı: Ağırlık sayısının azaltıldığı ve ağırlıkların uygunluğunun kontrol edildiği katmandır. Bu katmanın ana görevi, ağdaki gösterimin ve parametrelerin yer değiştirme boyutunu ve hesaplama adımlarının sayısını azaltmaktır. Bu şekilde ağdaki uyumsuzluklar kontrol edilir. Bu tez çalışmasında tasarlanan CNN mimarisinin max-pooling işlemi için 2x2 filtre kullanılmıştır.

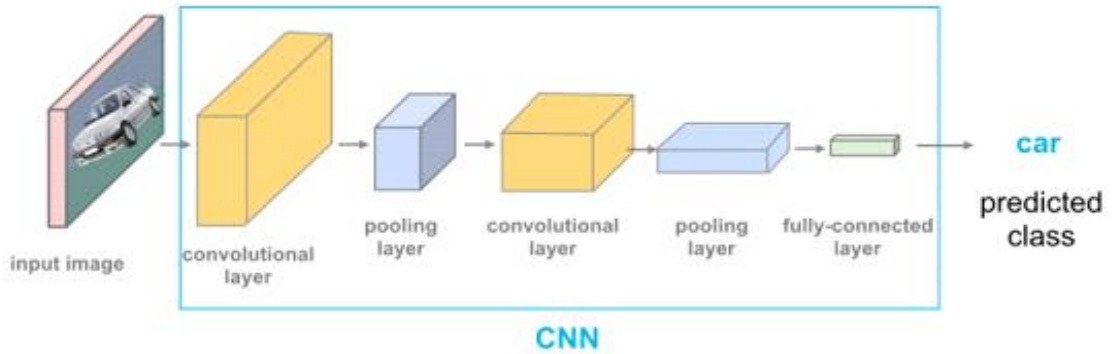
Düzleştirme Katmanı: Sonraki adımlarda oluşturulan klasik sinir ağı için verilerin hazırlandığı katmandır. Genel olarak, sinir ağları, tek boyutlu dizilerden girdi verilerini alır. Bu katman sayesinde Evrişimli ve Alt Örnekleme katmanlarından gelen veriler tek boyutlu bir diziye dönüştürülür ve bir sonraki adımda oluşturulacak olan Tam Bağlantılı Katman'a aktarılır.

Yoğun Katman: Sınıflandırmada kullanılan klasik sinir ağını içeren katmandır. CNN ağının son katmanıdır. Düzleştirme Katmanı'ndan veri alır ve sinir ağı yöntemi ile öğrenme işlemini gerçekleştirir.

Ağın parametreleri seçilirken parametre sayısı; giriş veri boyutu X_- , filtre boyutu U_- , çıktı boyutu Y_- , evrişimli katman derinliği M_- ve tekrarlanan çerçeve sayısı N_- olmak üzere, Tablo 2.1.'de özetlendiği şekilde belirlenmiştir.

Tablo 2.1 CNN Modeli katman sayısı hesaplama yöntemi

Size	Katman	
	<i>Evrişimli Katman</i>	<i>Maxpooling Katmanı</i>
Giriş Boyutu	$X_- * X_- * M_-$	$X_- * X_- * M_-$
Çıkış Boyutu	$Y_- * Y_- * N_-$	$Y_- * Y_- * M_-$
Parametre Sayısı	$(U_- * U_- * M_- + 1) * N_-$	0

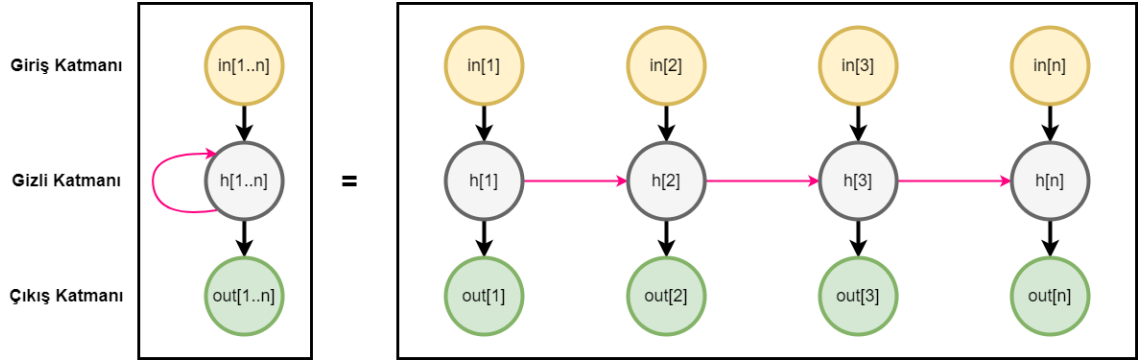


Şekil 2.3. CNN ağları gösterimi [31]

2.1.2 Yinelemeli Sinir Ağları (RNN) ve LSTM Ağları

Yinelemeli ağlar, sıralı ya da zaman serisi türü giriş verilerini sınıflandırmak için tasarlanmış ve geleneksel çok katmanlı almaç mimarisine benzeyen bir derin sinir ağı türüdür (Şekil 2.4). Ağın eğitiminde hatanın zamanla geri yayılımı ve gradyen iniş algoritması kullanılmaktadır. Gradyenleri hesaplamak için kullanılan hızlı bir algoritma olarak genelde “Geri Yayılım Algoritması” tercih edilir. Hesaplanan çıktı ile beklenen çıktı değeri arasındaki fark, ağ hatası olarak adlandırılır. Geri yayılım algoritması çıkış katmanında hesaplanan ağ hatasını alır ve giriş doğru adım

adım iletir. Bu hatalar her katmanda sırayla nöronların ağırlıklarını güncellemek için kullanılır [20]. Bu işlem, ağ hatası minimum olana kadar tekrarlanır.



Şekil 2.4 Yinelemeli Ağ – RNN modeli gösterimi

Geri yayılım metodunda maliyet fonksiyonu E olmak üzere maliyetin, ağırlıklara bağlı şekilde değişim oranı Denklem (2.1)'de ifade edildiği gibidir.

$$\nabla E(\theta) = \frac{\partial E}{\partial \theta_t} \quad (2.1)$$

Burada θ_t ifadesi, t 'nci iterasyonda ağın tüm ağırlıklarını ifade eden vektördür. Geri yayılım metodu uygulandığında bilgi, öğrenme süreci boyunca biriktirilir (toplanır). Ağ ağırlıkları,

$$\theta_{t+1} = \theta_t + \Delta(\theta)_t \quad (2.2)$$

eşitliği ile hesaplanır. Burada θ_t ifadesi, t 'nci iterasyonda ağın ağırlıklarını; θ_{t+1} ise $(t+1)$ 'inci iterasyondaki ağırlıkları temsil eder. $\Delta(\theta)_t$ ise öğrenme sürecidir. Yapay sinir ağı öğrenme süreci, her bir periyotta yapılan hatayı azaltmak için $\Delta(\theta)_t$ 'nin güncellenmesini gerektirir. $\Delta(\theta)_t$ fonksiyonu şu şekilde yazılabilir:

$$\Delta(\theta)_t = -\alpha \nabla E(\theta) \quad (2.3)$$

Denklem (2.3)'de α çarpanı, pozitif sabit değer olan öğrenme oranıdır. Geri yayılım metodu; temelde öğrenme oranı α ve hedeflenen çıktı ile hesaplanan çıktı arası fark olan E 'ye dayanır. Bu geri yayılım metodunun, ileri beslemeli sinir ağlarına uygulanmasıyla aslında Yinelemeli Yapay Sinir Ağı kurulmuş olur.

Şekil 2.4.'te gösterilen yapıda 3. nöron için geri yayılım algoritması işletilecek olursa, çıkıştaki hata E_3 sırasıyla 3., 2. ve 1. nöronlar arasındaki ağırlıkların güncellenmesi için şu şekilde yayılır:

$$S_1 = W.x_1 + U.S_0, S_2 = W.x_2 + U.S_1, S_3 = W.x_3 + U.S_2 \quad (2.4)$$

$$E_1 = \frac{1}{2} (S_3 - Y)^2 \quad (2.5)$$

$$\frac{\partial E}{\partial S_3} = S_3 - Y \quad (2.6)$$

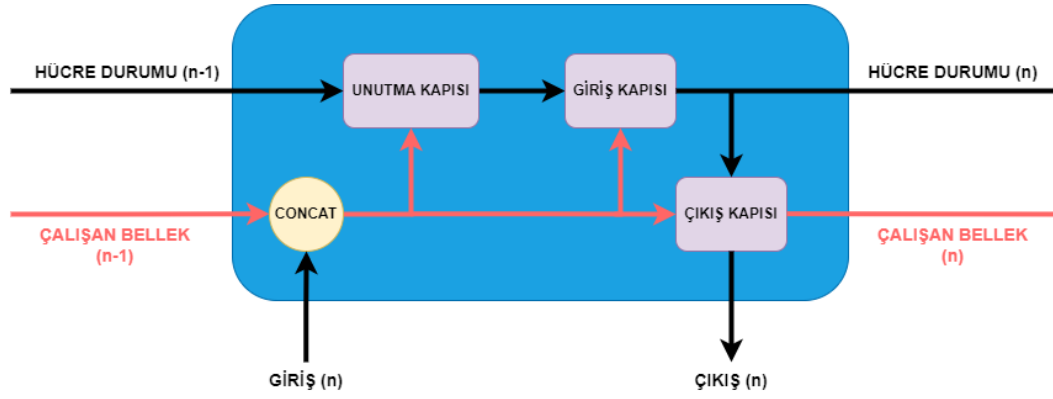
$$\frac{\partial E}{\partial S_2} = (S_3 - Y) \cdot U = \frac{\partial E}{\partial S_3} \cdot \frac{\partial S_3}{\partial S_2} \quad (2.7)$$

$$\frac{\partial E}{\partial S_1} = (S_3 - Y) \cdot U \cdot U = \frac{\partial E}{\partial S_3} \cdot \frac{\partial S_3}{\partial S_2} \cdot \frac{\partial S_2}{\partial S_1} \quad (2.8)$$

Burada E : karesel hatayı, W : giriş ile nöron arasındaki ağırlıkları, U : geri besleme ağırlıklarını, X : giriş verisini, Y : istenen çıkışı, S_i : ise nöronun çıkışının ifade etmektedir.

Hatanın geri yayılması işlemi boyunca son katmandan ilk katmana ilerlenirken gradyanın genliği azalır ve sifıra gidebilir. Bu da ağırlıkların güncellenmesi ve ağırlık verimliliği açısından dezavantajdır. Bu duruma Kaybolan Gradyan adı verilir. Bu problemin aşılması için bu çalışmada olduğu gibi Yinelemeli Sinir Ağı türü olan LSTM metodu kullanılabilir. Standart RNN'lerde olduğu gibi, LSTM'ler geri beslemeli yapılardır; çalışma belleğini birden fazla yinelemede sürdürür ve bir araya getirir. LSTM'ler ayrıca zaman içinde geri yayılım aracılığıyla optimize edilen ağırlıklarla yinelemeler arasında ağırlıkları ve aktivasyon işlevlerini paylaşır.

LSTM, geri yayılım esnasında farklı zaman ve katmanlardan gelen hata değerini korumaya çalışır. LSTM yapısında temel yapıtaşı tek bir sinir ağı katmanı yerine, birbirine belirli bir kuralla bağlı dört katmandan oluşur. Bu katmanlara kapı adı verilmektedir (Şekil 2.5).



Şekil 2.5 LSTM Blok Şeması

LSTM düğümüne gelen bilgiler şunları içerir [21]:

Önceki Yineleme'den Gelen Hücre Durumu: Bu LSTM'e özel bir bileşendir.

Önceki Yineleme'den Gelen Çalışma Belleği: Geleneksel RNN'lerdeki ile aynıdır.

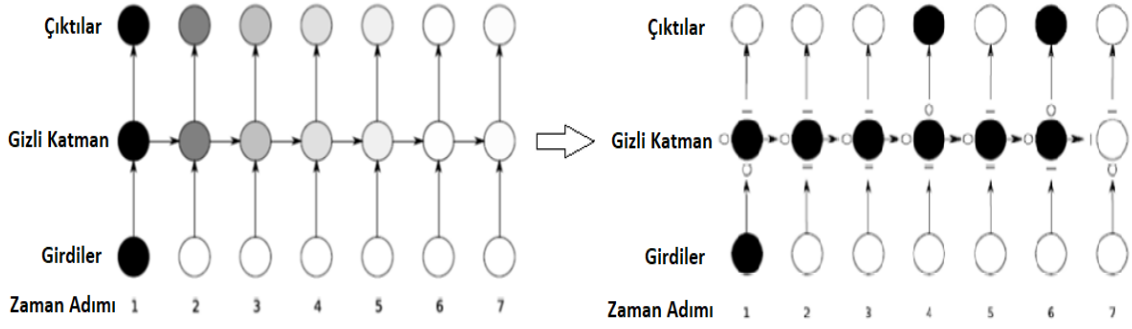
Geçerli Yineleme'den Gelen Giriş Değeri: Geleneksel RNN'lerde olduğu gibidir.

Hücre Durumu: LSTM'in en önemli bileşenidir denilebilir. Görevi, düğümün tüm yinelemeleri boyunca bilgiyi (gerekirse) kalıcı olarak uzun süreli bir bellek olarak hareket ettirmektir. Hücre Durumu, gereksiz bilgileri kaldırmak veya önemli bağlamsal bilgileri saklamak için değiştirilebilir. Bu nedenle, erken yinelemelerden gelen önemli bilgilerin uzun diziler boyunca kaybolmamasını sağlar.

Unutma Kapısı: Unutma Kapısı'nın tek rolü, Hücre Durumu'ndan hangi bilgilerin çıkarılması gerektiğine karar vermektir. Bu, birleştirilmiş çalışma belleği / Giriş Değeri üzerinde hesaplamalar yaparak ve sonra bunları Hücre Durumu'na uygulayarak yapılır.

Giriş Kapısı: Giriş Kapısı, birleştirilmiş çalışma belleğine / Giriş Değeri'ne bağlı olarak Hücre Durumu'na hangi bilgilerin eklenmesi gerektiğine karar verir.

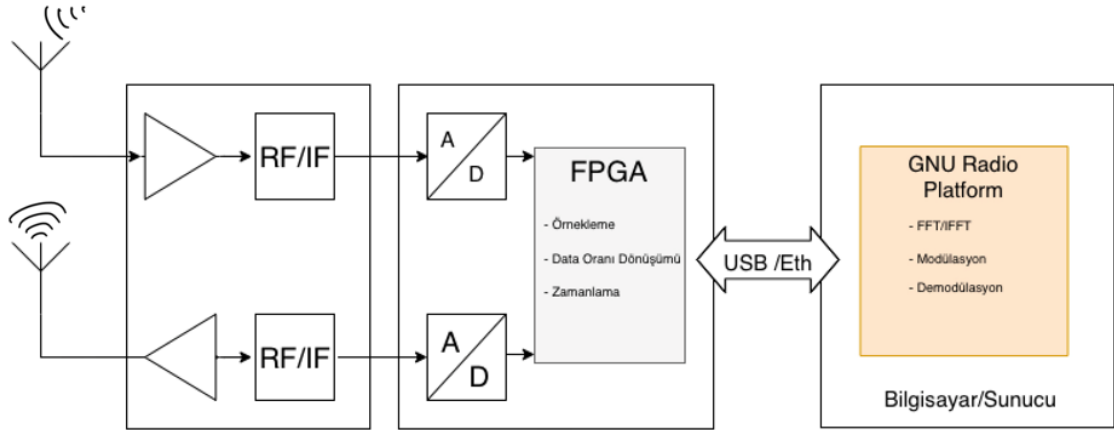
Çıkış Kapısı: Çıkış Kapısı, mevcut Hücre Durumu ve birleştirilmiş çalışma belleği / Giriş Değeri'ne ilişkin hesaplamalara dayanarak bu düğümün hangi çalışma belleğini çıkaracağına karar verir.



Şekil 2.6 LSTM nöronlarının güncellenmesi

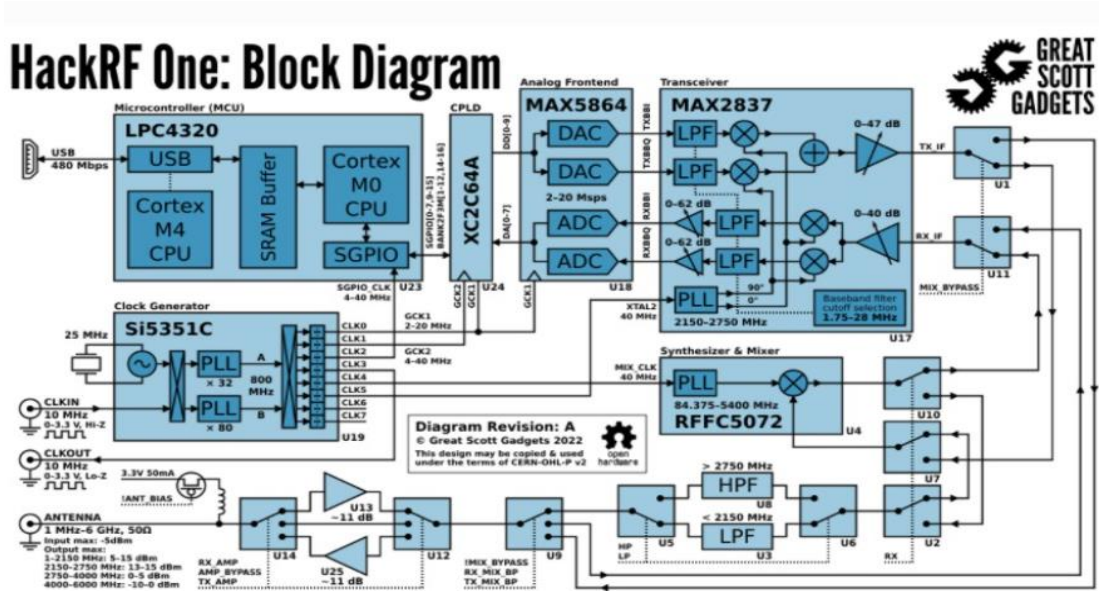
2.2 Yazılım Tabanlı Radyo

Yazılım tabanlı radyo, yazılım ile konfigüre edilebilen, farklı modülasyon türlerinde haberleşebilen bir yapıdır. SDR'nin yazılım ile konfigüre edilebilmesi, endüstriyel kullanımda tüm işlem yığınlarının veya modülasyon türlerinin tek bir güncelleme ile değiştirilebilme olanağını sağlamaktadır. Bu da yeni donanım geliştirmeleri için zaman ve maliyet tasarrufu sağlamaktadır. Açık kaynak olarak geliştirilmiş olan bir program (GNURadio) ve donanım (SDR) sayesinde gerçekleştirilen yazılım tabanlı radyo sisteminin blok diyagramı Şekil 2.7.'da gösterilmiştir.



Şekil 2.7 Temel bir SDR uygulamasının blok diyagramı

Bu çalışmada ortamdaki kablosuz sinyallerin toplanması için kullanılan HackRF One modeli, açık kaynak donanım ve yazılım altyapısına sahip olan bir SDR'dir. HackRF One blok şeması Şekil 2.8'de verilmiştir.



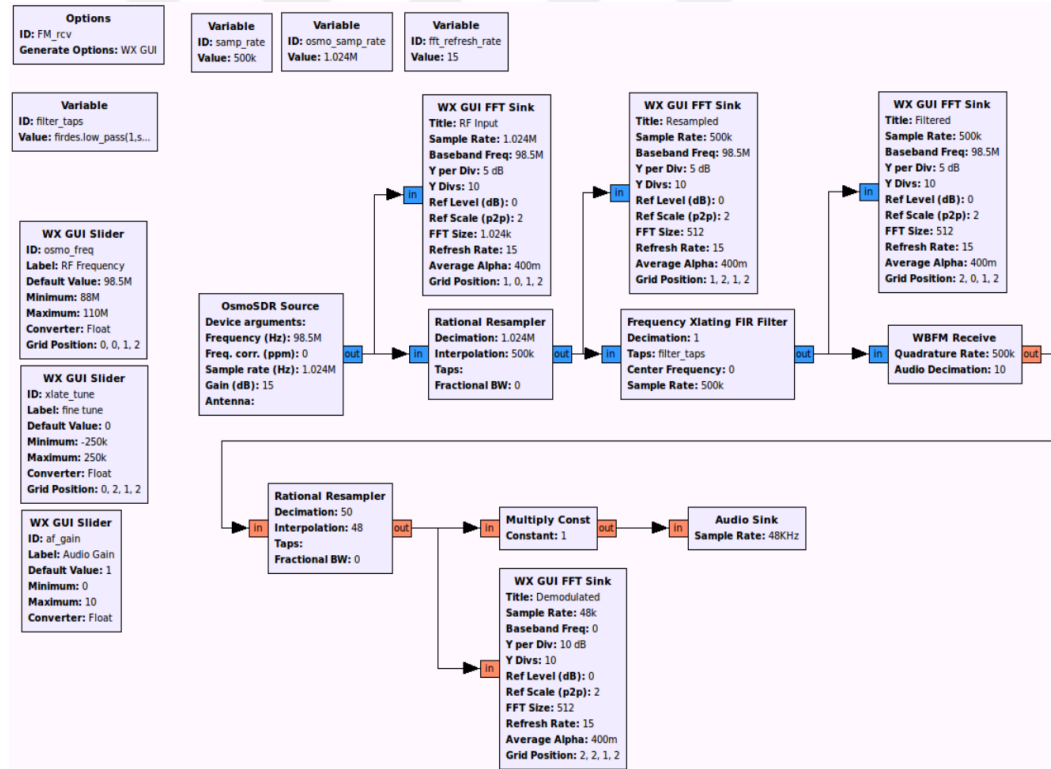
Şekil 2.8 HackRF One blok diyagramı [7]

SDR ile veri toplanırken aranan sinyallerin frekanslarının düzgün algılanabilmesi için bu spektrum değerlerine uygun antenler seçilmiştir. Her frekans için manuel olarak antenler değiştirilerek uygulama gerçekleştirilmiştir. Tablo 2.2’de kullanılan antenlerin özellikleri verilmiştir.

Tablo 2.2 Veri toplamada kullanılan anten modelleri ve özellikleri

Anten Modeli	Frekans	Kazanç	Polarizasyon
LPWA/ISM Anten	433-960 MHz	3 dBi	Lineer
Dipol LTE Anten	700-2700 MHz	9 dBi	Lineer Dikey
Multi-band Anten	2.4/5.1/5.8 GHz	4/4.5/5dBi	Omni Directional
4G LTE GSM Blade Anten	698-960 MHz, 1710-2690 MHz	5.5 dBi	Dikey
FM Teleskobik	30-1000 MHz	5 dBi	Lineer

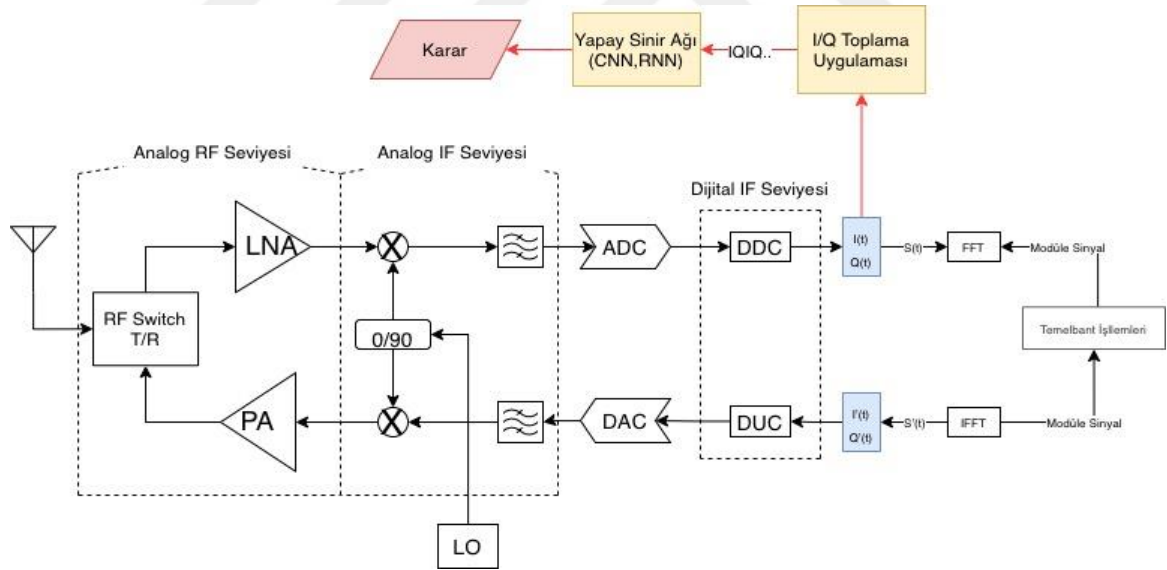
SDR ile yapılan uygulamalardaki başarımlar, SDR ile iletişim halinde olan bilgisayar arabiriminin başarımına bağlıdır. Bu bilgisayar gömülü sistem barındıran bir çip seviyesinde de olabilir veya bir kişisel bilgisayar da olabilir. Bilgisayarın işlem kapasitesi, uygulamada geliştirilecek ve kullanılacak olan sinyal/veri işleme algoritmalarının başarımını etkileyecektir ve bu durum gerçek zamanlı sistemlerde önemli bir faktördür. GNURadio, SDR uygulamalarında sinyal işleme sürecinde kullanılabilecek açık kaynak bir radyo geliştirme platformudur. Çoğunlukla Linux tabanlı işletim sistemlerinde kullanılır ve grafiksel bir arayüz sağladığı için geliştirici için kolaylık sunmaktadır. Bir radyo uygulamasında yazılımsal veya donanımsal olarak yer alabilecek bazı bloklar: filtreler, modülatör ve demodülatörler, sinyal üreticileri ve bunların yanı sıra grafik çizdirme bloklarını içeren bir uygulamadır. Şekil 2.9’da GNU Radio ile belirli bir frekansta ve kullanıcı tarafından kolaylıkla RF parametreleri girilen bir uygulama şeması görülmektedir.



Şekil 2.9 GNU Radio FM Radyo Alıcı blok diyagramı

HackRF One ile yapılan bu çalışmadaki uygulamada da Ubuntu 18.04.6 LTS versiyon Linux tabanlı sistemde uygulama gerçekleştirilmiştir. GNURadio derleme komut dosyasını kullanarak GNURadio ve OsmoSDR sürücüsü yüklenmiştir. GNURadio’da tasarlanan ve derlenen proje, SDR platformu içerisine aktarılırken

ve SDR'da uygulama çalıştıktan sonra dijital veri toplanırken SDR ile Linux PC arasında USB fiziksel arayüzü üzerinden iletişim gerçekleşmiştir. Projenin başlarında GNURadio ile I/Q verisi üretilip bir dosyaya istenen formatta aktarılmıştır. Bu süreçte [26]'daki çalışma referans alınarak bu uygulama gerçekleştirilmiştir. Bu yöntemle oluşturulan ham veri, yapay sinir ağı modelinde ön işlem sürecinden geçtikten sonra ağı beslemiştir. Öncelikle bu yöntem ile ilerlendikten sonra verinin toplanması, depolanması ve yapay zeka uygulamasına dahil edilmesi süreçlerinin hantallığı farkedilmiştir. Bunun üzerine GNURadio gibi bir GUI arayüzü kullanmak yerine Python dili kullanılarak yeni yazılım senaryoları hazırlanmıştır. Bunun avantajı yapay sinir ağı eğitilirken oluşturulan SDR veri toplama yazılımı ile yapay zeka yazılımı aynı proje altında çalıştırılmış ve daha hızlı bir uygulama olduğu görülmüştür. Bu uygulama gerçekleştirilirken ortam olarak Intel i7 işlemci ve NVIDIA Geforce ekran kartı altyapısına sahip bir bilgisayarda Linux platform üzerinde Pycharm Community derleyicisi kullanılmıştır.



Şekil 2.10 Uygulama sistem blok diyagramı

Şekil 2.10'da görülen diyagramda giriş verisinin hangi seviyede elde edildiği görülmektedir. Temelbant seviyesindeki işlemlere gerek duymadan, henüz bu seviyede demodüle edilmemiş olan sinyalin I/Q görüntüsü ele alınarak yapay sinir ağına işlenmiştir. Bu sayede zaman domeninden frekans domenine geçmeden ve frekans domenindeki sinyal işleme süreçlerini gerçekleştirilmeden toplanan

sinyalin türü anlaşılmasına çalışılmıştır. Bu şekilde algılanan sinyalin gerçek frekans değeri olan RF sinyal seviyesinde olunmadığından sinyalin iletim ortamındaki spektrumunu algılamak mümkün değildir. Çünkü gelen sinyal anten ile algılandıktan sonra analog ve dijital bir takım frekans dönüşüm işlemlerine maruz kalmıştır. Bu süreçte kullanılan Mixer komponentleri ile sinyalin seviyesi aşağı veya yukarı seviyede değiştirilmiştir. Ancak zaten uygulamada sinyali topladığımız frekans değeri bilindiği için o seviyedeki I/Q görüntüsünü tespit ederek frekans değeriyle eşleştirip bir sonuç elde edilmiştir. Ayrıca eğitim aşamasında hangi frekansta hangi sinyalin ortamda etkin olduğu da gerçek ortam verilerinin farkında olunmasından ötürü eğitim uygulamasının en etkili şekilde gerçekleştirildiği söylenebilir. Yapay sinir ağı bu aşamada toplanan veriden bir örüntü elde etmiş ve bunu verilen sınıflarla etiketlemiştir. Karar sürecinde de benzer örüntüyü belirtilen frekansta elde etmesi doğru sınıflandırmayı yapmasında etkili olmuştur.

2.3 Modülasyon Türleri

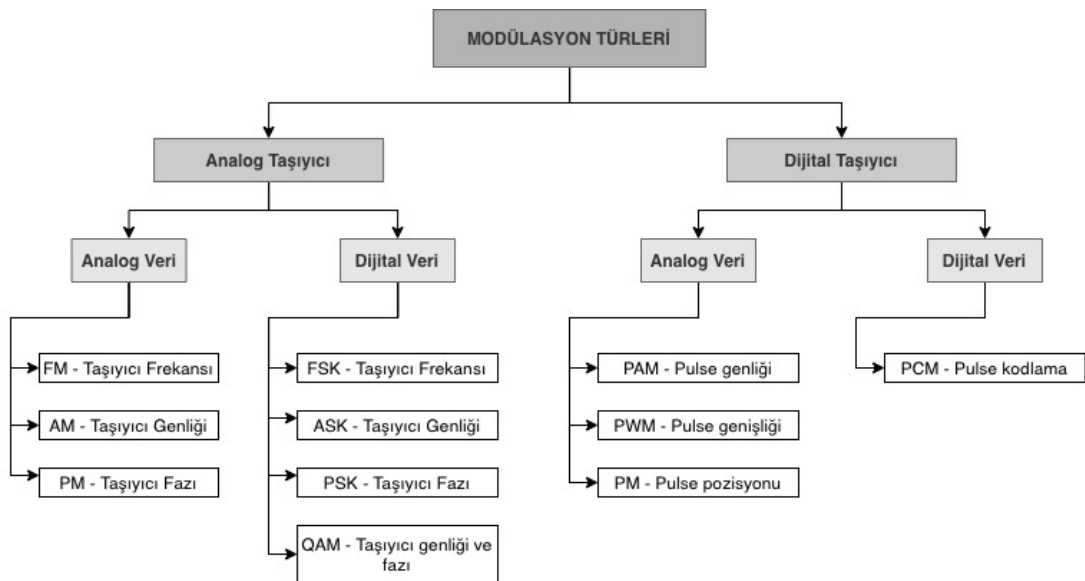
Modern haberleşme sistemlerinde dijital ve analog olmak üzere iki çeşit modülasyon türü bulunmaktadır ve frekans, genlik, faz gibi temel sinyal parametrelerinin manüplasyonu ile modülasyon süreci gerçekleştirilmektedir. Çalışmada dinleme aralığında olan sinyallerden yalnızca FM sinyallerinin kodlanma türü analog modülasyon türünden, diğer sinyallerin kodlanma türü ise dijital modülasyon türündendir. Analog ve dijital modülasyon teknikleri doğrusal veya doğrusal olmayan türde olabilir. Doğrusal modülasyonda, iletilen sinyalin genliği ve/veya fazı, dijital modülasyon sinyali ile doğrusal olarak değişirken, iletilen sinyal genliği, doğrusal olmayan teknikler için sabittir [38].

Bu çalışmada ele alınan ve sınıflandırılan sinyal türleri ve bu sinyallerin elde edilmesi için kullanılan çeşitli modülasyonlar Tablo 2.3'te verilmiştir. Buradaki haberleşme modlarının kullandığı yaygın modülasyon türleri saptanmış ve bunların tamamı olmasa da bazıları belirlenerek bu modülasyonların I/Q diyagramları referans alınmıştır. Bu sayede dinlenen spektrumdaki frekans değeri ile karesel modülasyon görüntüsü (I/Q diyagramı) eşleştirilerek sinir ağı eğitilmiştir.

Tablo 2.3 Uygulama frekanslarında sık kullanılan modülasyon türleri

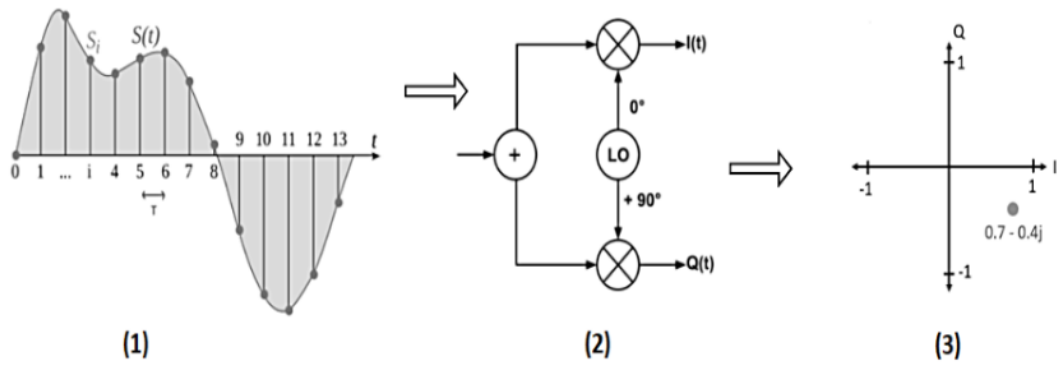
Sinyal Türü	Modülasyon Türü
WiFi 2.4G/5G	QPSK / 16-QAM /64-QAM
LTE	16-QAM / 64-QAM / 256-QAM
FM Radyo	FM - Analog Mod.
ISM Bant	ASK/FSK

Bu çalışmada algılanan sinyal sınıfları analog ve digital modülasyon türleri ile kodlanmış olan işaretlerdir. Şekil 2.11’de bu modülasyon türleri kategorize edilmiştir. Bunlardan FM, ASK, FSK, QPSK, QAM türleri günümüzde radyo haberleşmesinde en yaygın kullanılan kodlama yöntemlerindendir. Bu çalışmada da bu modülasyon türlerine odaklanılmıştır ancak hiçbir şekilde temel bant düzeyinde bir işlem gerçekleştirilmemiştir. Modülasyon işleminin sonucunda elde edilen ve faz/genlik bilgisiyle kodlanmış olan I ve Q değerleri, alıcı cihaz tarafında demodülasyon işlemi gerçekleştirilmeden önce işlenmiş ve sınıflandırma algoritmasına giriş verisi olarak sunulurken sinir ağı modeli eğitilmiştir.



Şekil 2.11 Modülasyon türleri

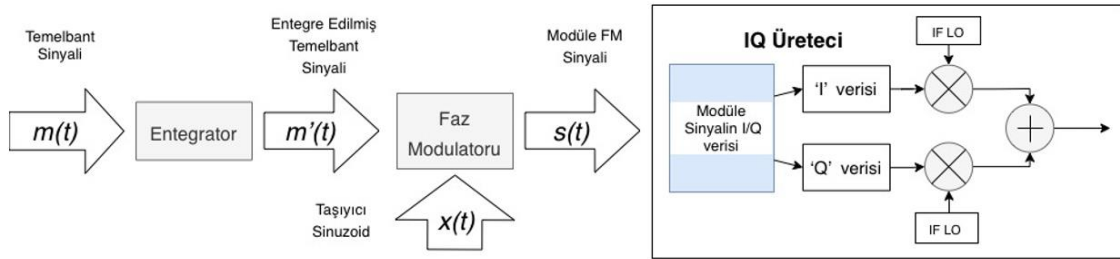
Modülasyon işlemi sonucunda oluşan faz ve genlik bilgilerini taşıyan iki bileşen olan I ve Q sembolleri zaman (t) domeninde birleştirilerek RF sinyali olarak iletim ortamında yayılmak üzere bir Dijital-Analog dönüşüm işlemine tabi tutulur. DAC yapıları sayesinde bu dijital veri, analog sinyal biçimine dönüştürülür. Aynı şekilde alıcı tarafında da I ve Q bileşenlerini elde etmek için önce analog formda olan RF sinyali, ADC birimi tarafından dijital forma dönüştürülerek $I+jQ$ verisi elde edilir. Sonrasında bu iki veriyi ayırmak için bir Kareleme Demodülatörü kullanılır ve I/Q verileri ayrı ayrı elde edilmiş olurlar. Şekil 2.12’de bu süreci gösteren bir görsel yer almaktadır. Bu işlem sırasında kullanılan Lokal Osilatör elemanı yüzünden algılanmak istenen RF sinyal seviyesinin sahip olduğu taşıyıcı frekansı algılanamaz. Yani iletim ortamında verici ve alıcı cihaz arasında yol almış olan RF sinyalinin iletim ortamında sahip olduğu asıl frekans bilinmemektedir. Bu yüzden bu çalışmada da I ve Q verilerinin takımyıldız diyagramı üzerindeki koordinatları algılanarak frekans değeri de sınıflandırma algoritması tarafından bilinecek şekilde ağa giriş verisi olarak sunulmuştur. Bu sayede frekans değeri ile çok boyutlu I/Q verileri ilişkilendirilerek ağın eğitimi gerçekleştirilmiştir.



Şekil 2.12 Sinyal örnekleme işlemi ve I/Q demodülasyonu

FM Modülasyon: Frekans Modülasyonu işlemindeki temel prensip temel bant seviyesinde ve analog formda olan bir bilgi sinyalinin sahip olduğu genlik değerinin taşıyıcı sinyalin farklı bir frekansı ile temsil ediliyor olmasıdır. Dünyada kullanılan FM radyo bandı , radyo frekans spektrumunun VHF (30-300MHz) olarak bilinen aralığında sınıflandırılır. FM radyo yayını için yayın olarak kullanılan frekans aralığı 87.5 MHz – 108 MHz aralığıdır [22]. Eski Sovyet

cumhuriyetlerinde ve bazı eski Doğu Bloku ülkelerinde, eski 65.8-74 MHz bandı da kullanılmaktadır. Atanan frekanslar 30 kHz aralıklardır. Bazen OIRT bandı olarak anılan bu bant, yavaş yavaş aşamalı olarak kullanımdan kaldırılmaktadır. OIRT bandının kullanıldığı yerlerde, 87.5–108.0 MHz bandı, CCIR bandı olarak adlandırılır. Bir diğer farklı frekans aralığı olarak Japonya'da 76-95 MHz bandı kullanılır. Bu çalışmada ise Türkiye'deki FM sinyallerinin yaygın kullanıldığı ve BTK [30]'nın belirlediği aralıkta FM radyo yayınından toplanan sinyaller kullanıldığından 87 – 108 MHz aralığında veri toplanmıştır. Ayrıca bu bant ülkemizde de en kolay şekilde manipüle edilen ve illegal bir şekilde özellikle reklam amaçlı işgal edilen bir frekans bandıdır. Bu durumunu engellemek adına RTÜK kurumunun teftiş ilkeleri doğrultusunda çalışmaları sürekli olarak devam etmektedir.

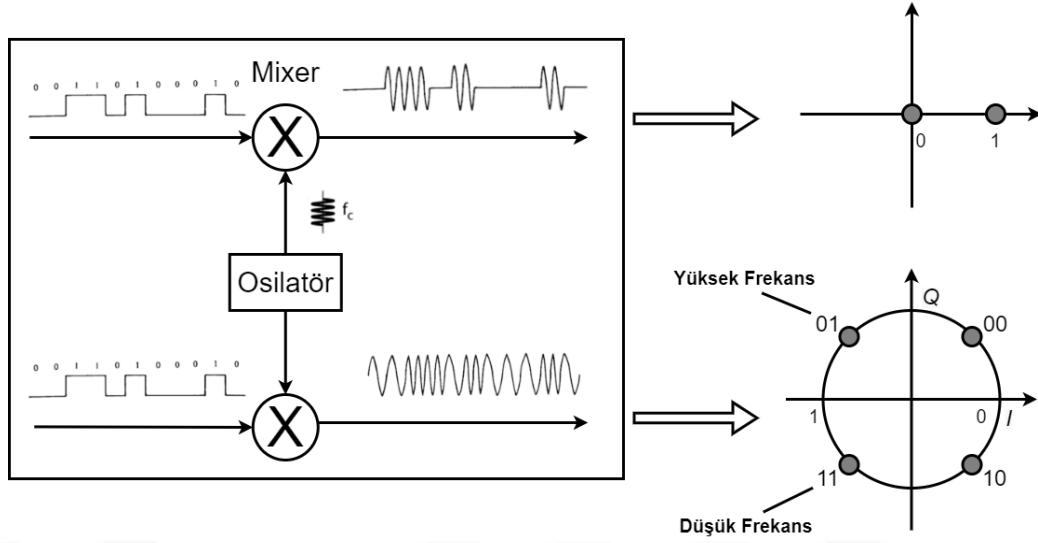


Şekil 2.13 FM Modülasyon süreci

ASK ve FSK Modülasyonları: Daha önce bahsedildiği üzere bu çalışmada ISM bantlarında uygulama frekansları olarak 433 MHz ve 866 MHz frekansları ve dolayları algılanmıştır. Özellikle yayın algılanan frekans değerleri 433-437 ve 863-870 MHz aralıklarıdır. 433/868 MHz gibi 1GHz altı ISM bant uygulamalarında ASK ve FSK modülasyonları yoğun bir şekilde kullanılır. Bu bantlardaki haberleşmede genel olarak 868 MHz bandı için FSK, 433 MHz bandı için ASK modülasyon kullanılmaktadır. ASK modülasyonu sinyal ve veri iletimi için genlik, FSK ise frekans domenini kullanır. Denklem (2.9)'da f_c taşıyıcı sinyalin frekansı olmak üzere A_0 ve A_1 değerleri modülasyon aşamasında kullanılan genlikleri ifade etmektedirler.

$$s(t) = \begin{cases} A_0 \cos(2\pi f_c t), & 0 \\ A_1 \cos(2\pi f_c t), & 1 \end{cases} \quad (2.9)$$

Örnek bir $s(t)$ modüle sinyalinin takımyıldız diyagramı aşağıdaki gibidir.



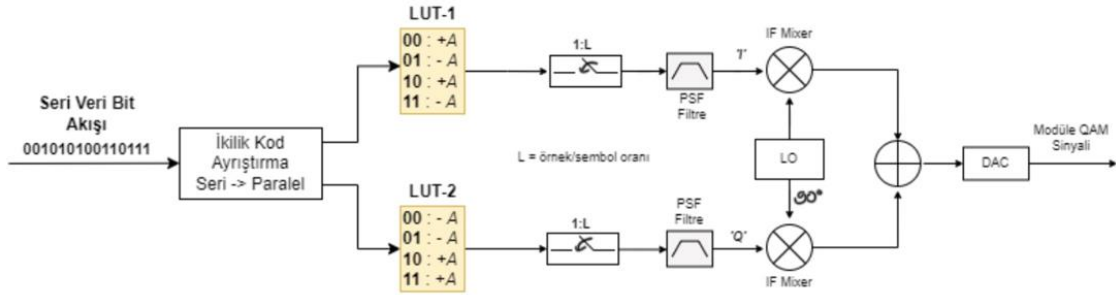
Şekil 2.14 ASK/FSK modülasyon süreci

FSK modülasyon işlemi için formül aşağıdaki gibidir. Burada da A değeri sabit bir genlik değerini ifade ederken f_1 ve f_2 değerleri ise modülasyonda kullanılacak frekans değerlerini temsil etmektedir.

$$s(t) = \begin{cases} A \cos(2\pi f_1 t), & 0 \\ A \cos(2\pi f_2 t), & 1 \end{cases} \quad (2.10)$$

PSK ve QAM Modülasyonları: Bu çalışmada özellikle geniş bant uygulamaları olan LTE ve WiFi sinyal sınıflarının kullandığı yaygın modülasyon türleri PSK ve QAM modülasyonlarıdır. QAM; PSK ve ASK'nın bir kombinasyonu olarak düşünülebilir. ASK gibi sadece genlik anahtarlama PSK gibi sadece faz anahtarlama olmamakla birlikte hem faz hem genlik bilgisinden faydalanan bir dijital modülasyon türüdür. Bu yüzden I/Q grafiğinde mutlak genlik değeri farklılıklar gösterebilmektedir. Gigabit seviyesinde kablosuz ağlar yaygın bir şekilde QAM modülasyon tekniğini kullanmaktadır. Genel kullanıma göz atıldığında geniş bant yayınlarda sıklıkla kullanılanlar 16QAM, 64QAM ve 256QAM metotlarıdır. Bunlar ülkelere, bölgelere ve sağlanan mobil haberleşme altyapısına göre değişiklik gösterebilir. Örneğin en bilinen kablosuz standartlardan biri olan WiFi IEEE 802.11 modelinde yüksek kapasiteli veri haberleşmesini sağlamak için QAM modülasyonunun türevlerinden (BPSK, QPSK, 16-QAM, 64 QAM gibi) faydalanılmaktadır. Bu QAM modellerinin her biri farklı sembol oranları ve sembol başına bit değerlerine sahiptir. Çalışmada

WiFi ve LTE sinyalleri için özellikle QPSK, 16-QAM, 256-QAM modülasyonları beklenmiştir.



Şekil 2.15 QAM modülasyon süreci

QAM modülasyonunun algılanması sırasında takımyıldız diyagramından yalnızca I/Q görüntüleri incelense de bu modülasyonun verici tarafında nasıl kodlandığını bilmekte fayda olduğu düşünülmüştür. $s(t)$ modüle sinyal olmak üzere:

$$s(t) = A_m \cos(2\pi Ft + \phi_m), \cos(A + B) = \cos A \cdot \cos B - \sin A \cdot \sin B \quad (2.11)$$

eşitliği ile ifade edilmektedir.

$$s(t) = A_m \cdot \cos(2\pi Ft) \cdot \cos(\phi_m) - A_m \cdot \sin(2\pi Ft) \cdot \sin(\phi_m) \\ [A_m \cos(\phi_m)] \cdot \cos(2\pi Ft) - [A_m \sin(\phi_m)] \cdot \sin(2\pi Ft) \quad (2.12)$$

Denklem (2.11) ve Denklem (2.12) sonucunda elde edilen I ve Q sinyalleri Denklem (2.13)'teki gibi ifade edilir.

$$\begin{cases} I = A_m \cos(\phi_m) \\ Q = A_m \sin(\phi_m) \end{cases} \quad (2.13)$$

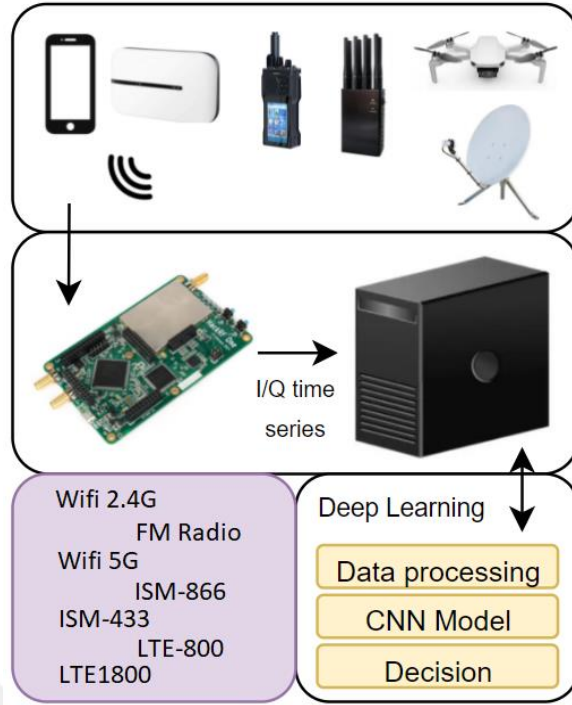
QAM modülasyonu sonucunda I ve Q komponentleri üretilmektedir. Bunlar modüle edilen sinyalin sembollerini temsil ederek genlik ve faz bilgisini taşımaktadırlar. Bu çalışmada alıcıda toplanan modüle sinyalin genlik ve faz bilgilerini içeren I ve Q değerlerinin konumlarından faydalanılarak bir algılama düzeyi geliştirilmiştir.

3.1 Uygulama Senaryosu

Bu çalışmada, SDR platformu üzerinde, derin öğrenme ağları yardımıyla ham I/Q verilerini anlamlandırmayı ve en iyi başarıyı elde etmeyi amaçlayan, temel bant seviyesinde gerçekleştirilen kapsamlı sinyal işlemeden mümkün olduğunca kaçınan bir uygulama gerçekleştirilmiştir. Bu sayede endüstriyel anlamda uygulanabilir bir yapı elde edilmeye çalışılmıştır.

Sinir ağlarını toplanan verilerle beslemeden önce giriş verilerinin ve etiketlerinin hazırlanması zorunludur. Derin öğrenmede bu süreç uygulamanın içeriğine göre değişiklik gösterebilir [9]. Her derin öğrenme modeli eğitiminde, girdi verilerinin yapısına göre bir veri ön işleme süreci zorunludur. Böylece ham veriler sinir ağlarına sunulur. Özellik çıkarma, makine öğrenmesi ve elimizdeki verilerle ilgili bilgileri kullanarak modeli daha iyi çalıştırmak için verilere bazı dönüşümler uygulayarak modeli besleme süreci olarak da düşünülebilir. İyi bir özellik çıkarımı, sorunu çözmek için daha az kaynak harcanmasını mümkün kılmaktadır.

Çalışmada, gerçek dünyada sıklıkla kullanılan FM Radyo, LTE-1800, ISM-433, ISM-866, LTE-800, WiFi-2.4G, Wifi-5G frekans bantları kullanılmıştır. Sinyal örnekleri toplanırken mümkün mertebede bu frekanslarda yayın yapan, haberleşen gerçek cihazlar kullanılmasına özen gösterilmiştir. SDR platformu olarak YTÜ BAP Koordinatörlüğü tarafından finanse edilmiş olan HackRF One SDR'dan faydalanılmıştır. Bu şekilde hibrit veri toplama yöntemi kullanılmış ve ağın eğitimi için kullanılacak veri setinin çeşitliliği ve örneklem sayısı olabildiğince yüksek tutulmuştur. Zaman serisi verileri şeklinde toplanan I/Q verileri Şekil 3.1'de gösterildiği gibi eğitim birimine aktarılır ve ağın eğitilmesi için kullanılır.

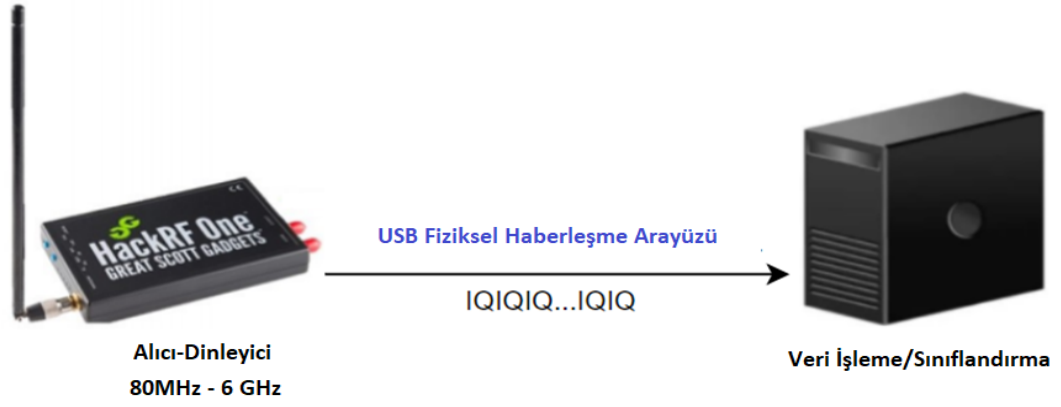


Şekil 3.1 Çalışmada kullanılan sınıflandırma modeli

3.2 Veri Setinin Toplanması

Ağı beslemek için ve sorgulama sırasında kullanmak üzere giriş verisi toplanırken kablosuz alıcı platformu olarak HackRF One SDR cihazı kullanılmıştır. Projede kullanılan HackRF One SDR özellikleri aşağıdaki gibidir:

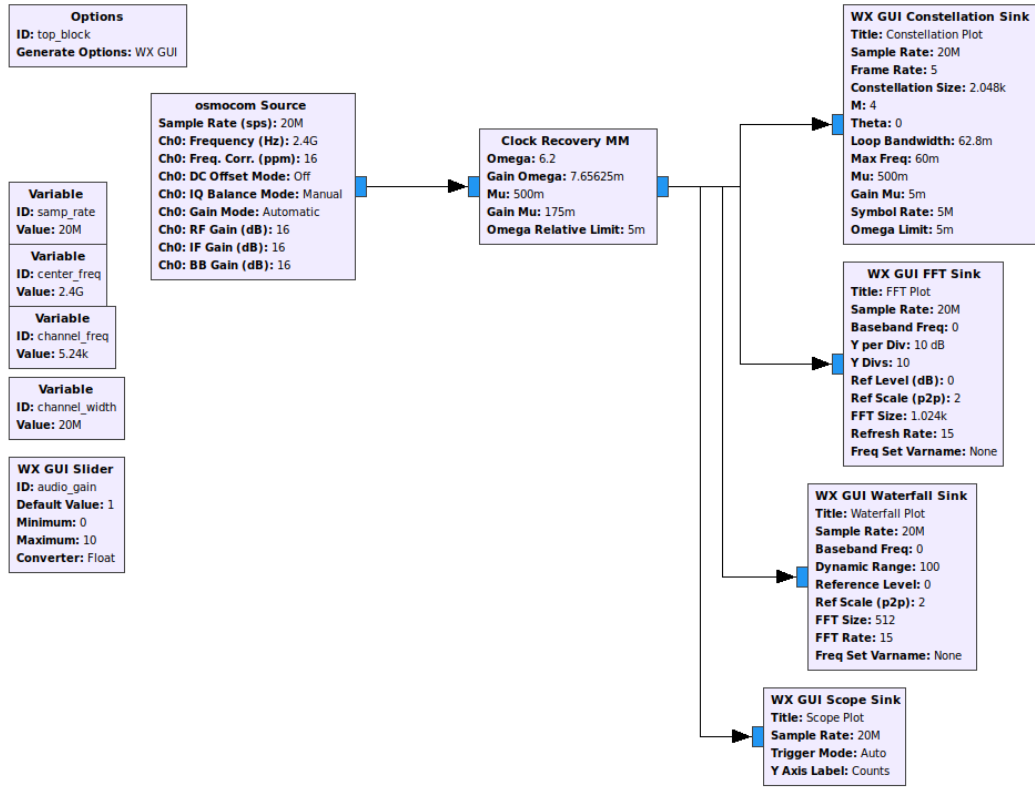
- 1 MHz'den 6000 MHz'e kadar herhangi bir radyo sinyali iletimi veya alımı
- Yarım çift yönlü alıcı-verici
- Maksimum örnekleme hızı: 20MSps
- 8 bit kareleme örnekleri
- Yüksek Hızlı USB arayüzü



Şekil 3.2 Sistemin veri haberleşmesinin temsili gösterimi

Çalışmada önceden planlanmış frekanslarda belirli niteliklere göre HackRF One ile dinleme yapılmış ve temel bant frekansında işleme tabi tutulmamış olan ham I/Q verileri elde edilmiştir. Örnekleme frekansı 16MSPs, RF Kazanç değeri 16 dB ve IF Kazanç değeri 32 dB olarak belirlenerek uygulama gerçekleştirilmiştir. Bu nihai değerler, farklı değerler iteratif olarak uygulanarak ve sonuçlar elde edilerek deneysel olarak belirlenmiştir. Alıcıdan toplanan I/Q verilerinin kayıt etme işlemi IQIQIQIQ...IQ formatında gerçekleştirilmiştir.

Toplanan ham veriler giriş veri vektörlerinin n adet örneğinden oluşur. Bu veri vektörleri, kablosuz sinyallerin varlığı ve davranışı hakkında çıkarımlarda bulunabilen sinir ağı modelleri oluşturmak için kullanılan girdilerdir.

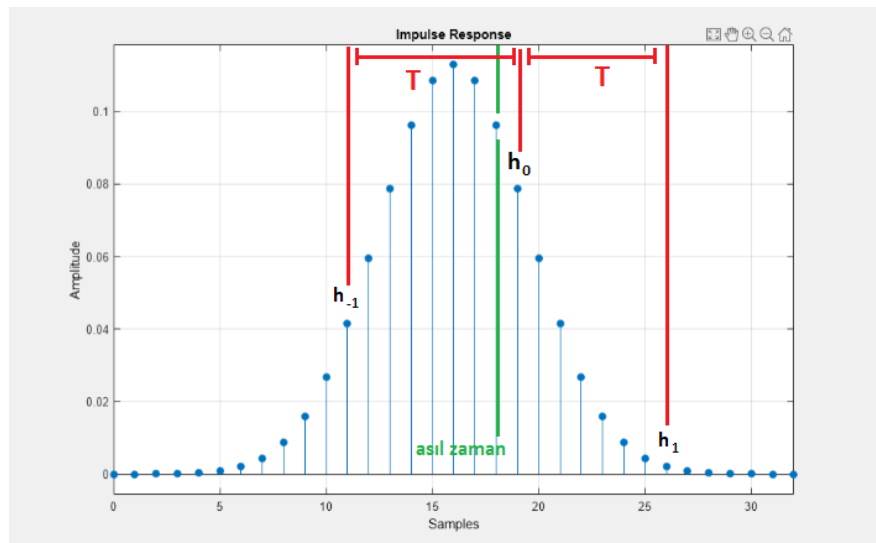


Şekil 3.3 Uygulamada kullanılan GNU Radio şematiği

Şekil 3.3'te uygulamada kullanılan GNU Radio şeması görülmektedir. Burada görülen Osmocom bloğu HackRF cihazının donanım sürücülerini kullanarak radyo haberleşmesinin cihaz üzerinde gerçekleşmesini sağlayan yapıdır. Alıcı modunda toplanması hedeflenen sinyalin frekans değeri ve bu işlem için donanımın yapılandırılması için gereken parametreler (IF kazancı, BB kazancı, RF kazancı, Frekans Düzeltme-ppm vs.) kullanılarak konfigüre edilir. Bu temel bloğun çıkışında ise I/Q değerleri elde edilmektedir. WX GUI Constellation bloğu sayesinde Şekil 3.7'de görülen figürler elde edilmektedir. Yine bu blokta da örnekleme değeri, constellation büyüklüğü gibi parametreler kritiktir. Bu çalışmada constellation büyüklüğünün iteratif olarak en uygun değeri bulunmuştur. Bu parametrenin değeri ne kadar yüksek olursa anlık grafiksel görünen I/Q örnek değeri o kadar artmaktadır. Maksimum 4096 ile sınırlandırılmıştır. WX GUI Waterfall Sink bloğu istenen merkez frekansta istenen bant genişliğinde frekans spektrumunun görüntüsünü sunar. Bu sayede dinleme yapılan spektrumdaki haberleşmenin aktivitesi takip edilebilmektedir. Şekil 3.5 ve

Şekil 3.6'da bu bloğun çıktıları görülebilmektedir. FFT Sink ve Scope blokları da benzer şekilde dinamik bir şekilde sinyalin BB seviyesine indirgenerek işlenmiş haliyle görüntüsünü ve aktivasyonunu sunar. Bunlar dinleme işleminin doğrulama yöntemi olarak ele alınmış ve kullanılmıştır. En kritik ve zorunlu bloğumuz WX GUI Constellation bloğudur. GNU Radio veya herhangi bir IF veya BB seviyesi sinyal işleme ve görüntüleme uygulamasında donanımın doğru şekilde yapılandırılması ve minimum hata ile I/Q verisinin toplanması önemlidir. Bunun için deneysel olarak elde edilen tecrübeler sonrasında Clock Recovery bloğunun gerekliliği de anlaşılmıştır. Clock Recovery, GNU Radio'daki sinyal işleme bloklarından biridir ve görevi, verici tarafından kullanılanlarla aynı frekans ve faza sahip bir sinyalden örnekleri kurtarmaktır. Bu, örneğin asenkron bir dijital sinyalden semboller çıkarmak istenildiğinde gereklidir. Alıcının, sinyalde bulunan birlerin ve sıfırların merkezleriyle senkronize edilmesini sağlar. Bunu yaparken de ADC bloğunun saat sinyali üzerinde bir takım güncellemeler yapmaktadır. Bu metodun ilk çalışmaları 1976 yılında yayınlanan bir makalede anlatılmıştır [25]. Çalışmayı yapanlar, ADC bloğunun çalışma hızının dönüştürülen sinyalin sembol hızında çalışacağı farklı bir zamanlama kurtarma metodu önermişlerdir. Bir PSF filtresinde Gauss darbe yanıtı $h(t)$, iletilen sembol $a(k)$, gürültü $n(t)$ olacak şekilde Denklem (3.1)'de alınan sinyal modellenmiştir.

$$x(t) = \sum_k (a_k h(t - kT) + n(t)) = \sum_k (a_k h_k + n(t)) \quad (3.1)$$



Şekil 3.4 PSF filtresi Gauss darbe yanıtı

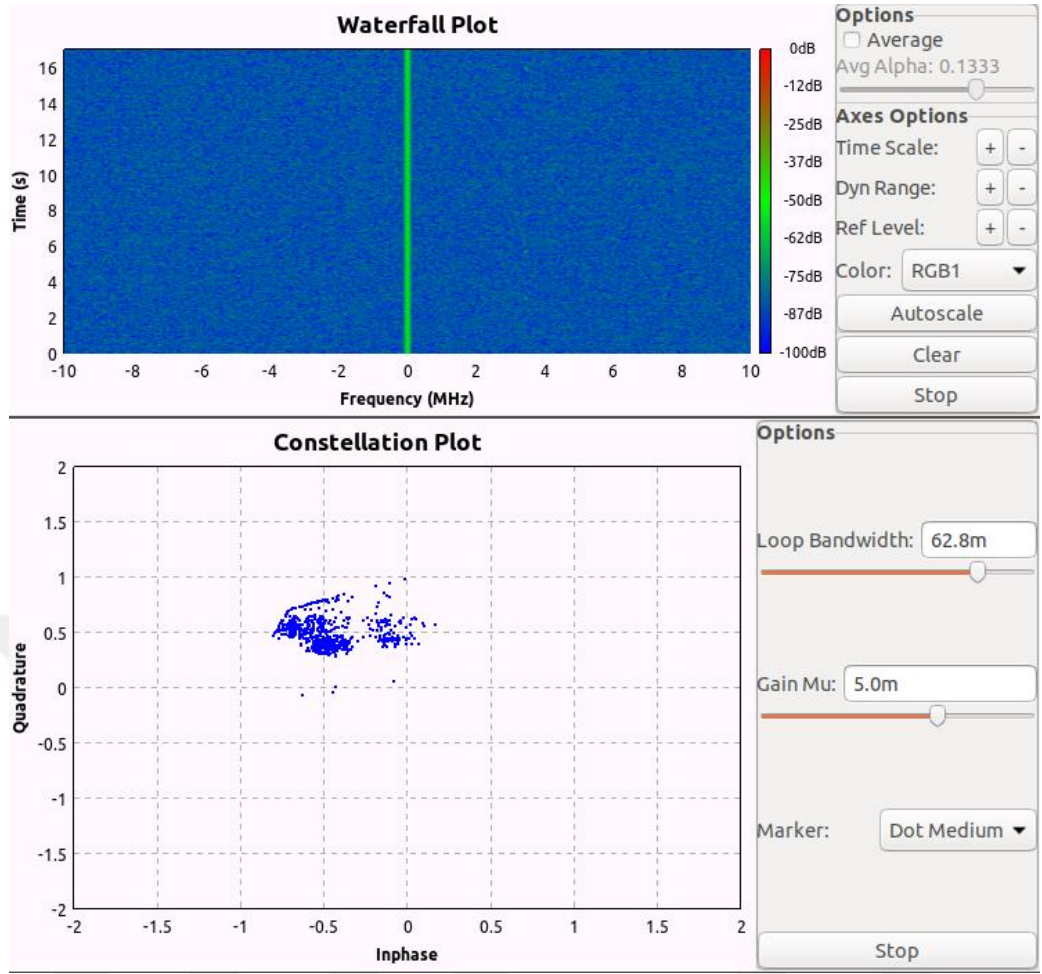
Şekil 3.4 'te PSF filtresinin yanıtına bakacak olursak h_0 : okunan değer, h_{-1} : T kadar önceki değer, h_1 : T kadar sonraki değer olmak üzere eğer ADC'te geç okuma yapıldıysa h_1-h_{-1} : negatif, erken okuma yapıldıysa h_1-h_{-1} : pozitif olacaktır. Erken okumada zamanlama fazı artırılmalı, geç okumada azaltılmalıdır. Yöntemin sunulduğu çalışmada [25] çözüm Denklem (3.2)'de ifade edilmiştir:

$$f(\tau) = \frac{1}{2} (h_1 - h_{-1}) = \frac{1}{2} (h(\tau + T) - h(\tau - T)) \quad (3.2)$$

h_1 ve h_{-1} değerlerinin bilinmesi ile $f(\tau)$ hesaplanır ve böylece örnekleme fazı artırılır veya azaltılır. $z(k)$, zamanlama fonksiyonu olmak üzere bu fonksiyonun beklenen değeri, $f(\tau)$ ile aynı şeyi temsil etmektedir. z_k 'nın formülasyonu Denklem (3.3)'te verilmiştir.

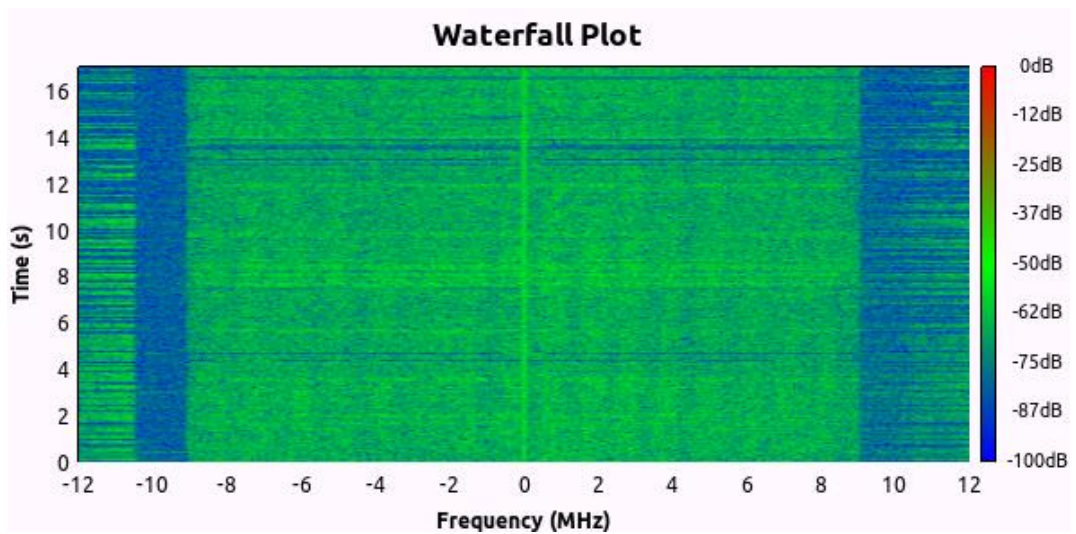
$$E\{z_k\} = f(\tau), \quad z_k = \frac{1}{2} (x_k a_{k-1} - x_{k-1} a_k) / E\{a_k^2\} \quad (3.3)$$

Dinleme yapılan frekans bandında herhangi bir haberleşme olup olmadığı GNU Radio akışında Waterfall Plot isimli blok ile algılanmıştır. Proje şeması çalıştırılmadan önce bu blok için dinlenecek frekans girilmek zorundadır ve anlık tek bir frekansta dinleme yapılmaktadır. Belirlenen frekansta haberleşme olmaması Şekil 3.5'te görüleceği gibi sinyal gücü (dB) renklendirmesine göre mavi renkte olacaktır ve Constellation grafiği aşağıdaki gibi her seferinde tek bir tarafta biriken değerler göstermektedir. Bu veri birikmesinin farklı frekanslarda oluştuğu da deneysel olarak gözlemlenmiştir.



Şekil 3.5 Frekans spektrumunda pasif durum gözlemlenmesi sonucu

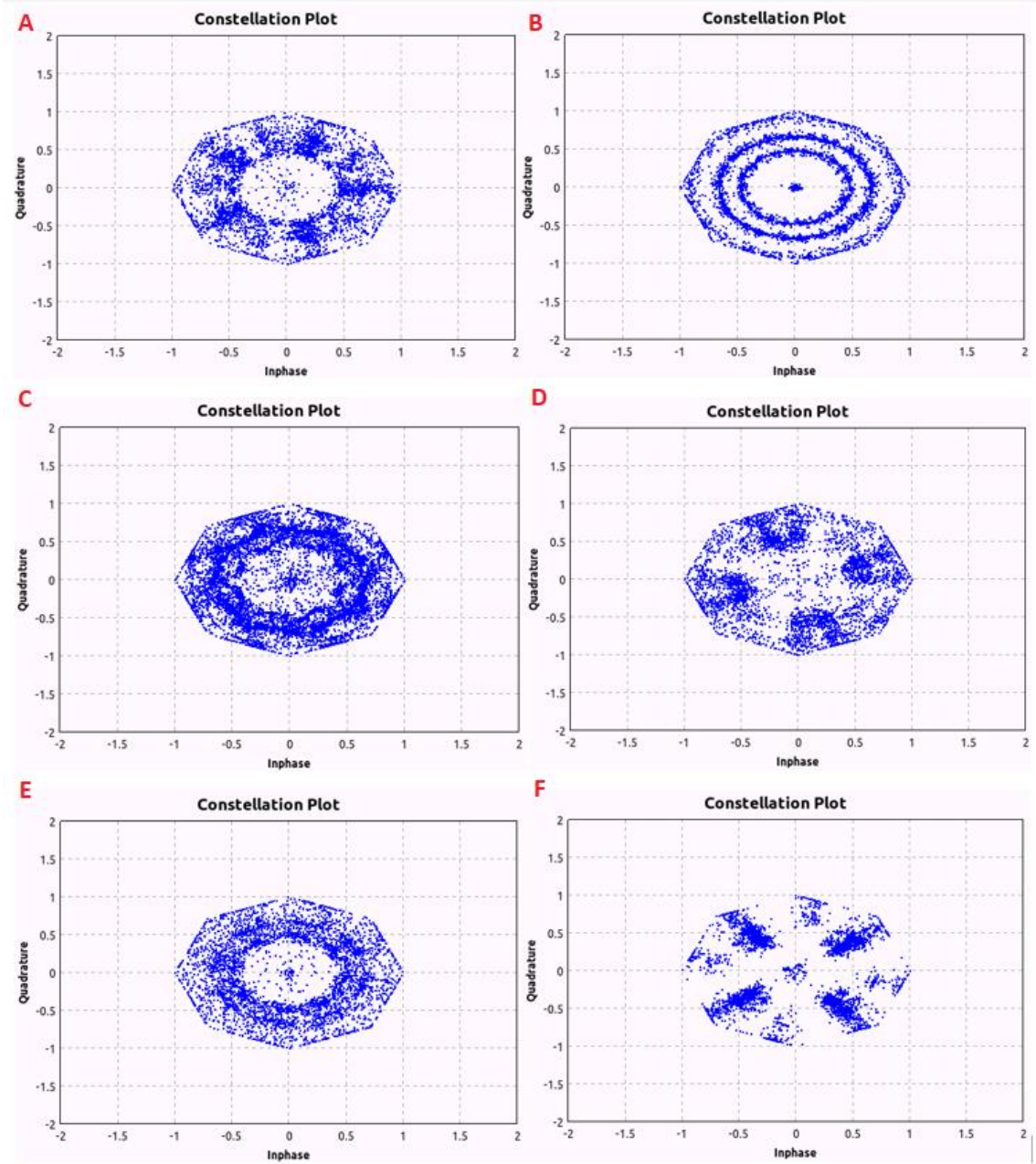
Aktif bir bant görüntüsü ise aşağıdaki Şekil 3.6'da görülmektedir. Burada LTE 1800 MHz frekansında dinleme yaparken frekans spektrumundaki hareketlilik net bir şekilde belirgindir.



Şekil 3.6 Frekans spektrumunda aktivite olması durumu

Şekil 3.7’de farklı frekanslarda yapılan dinlemelerin I/Q diagramları görülmektedir. Çalışmadaki mevcut donanım ve radyo yazılımları ile gerçekleştirilen uygulamada elde edilen sonuçlarda teorik görüntülerin net bir şekilde görülememesi ancak mantıklı bir örüntünün elde edilmiş olduğu çeşitli çalışmalarda [23][24] elde edilen sonuçlarla kıyaslanarak doğrulanmıştır. Bu sonuca göre IF frekans seviyesinde I ve Q değerleri elde edilmeden önce faz gürültüsünü azaltma çalışması , tasarlanacak uygun PSA yapıları ile gerçekleştirilebilir. Bunun sonucunda daha temiz bir I/Q diyagramı elde etme imkanı yüksek olacaktır. Bu çalışmada bu tür yapılardan faydalanılmamıştır.

Şekil 3.7’de elde edilen görüntülerin temsil ettiği sinyallerin kesim frekansları (f_c) , A,C: 5.24 GHz (Wifi-5G), B: 2.433 GHz (Wifi-2.4), D,F:1732.3 MHz(LTE Band3) , E: 840.1 Mhz (LTE Band20) şeklindedir. Bu frekanslarda özellikle dijital modülasyon tekniklerinden olan QAM ve PSK türü modülasyonlar kodlama tekniği olarak kullanılmaktadır. Burada bu görüntülerin sayısal karşılıkları olan I ve Q değerlerinin oluşturacağı sayı dizilerinde tespit edilmesi hedeflenmiş örüntüler sayesinde demodülasyon işlemine gerek kalmadan sinyal yapıları birbirlerinden ayrıştırılmıştır.



Şekil 3.7 GNU Radio WX GUI Constellation bloğu sonuçları (aktif spektrum)

3.3 Derin Öğrenme için Ön İşlem

Derin öğrenmedeki en kritik evrelerden biri veri ön işleme sürecidir. Çoğu zaman bu süreç, bir ağ modeli seçmekten bile daha kritiktir. Bu çalışmada, veri ön işleme, daha iyi kablosuz veri temsilleri elde etme olasılığını güçlendirmek için toplanan spektrum verilerinin analizini ve manipülasyonunu içerir. Kablosuz alıcıdan toplanan veriler aslında karmaşık veri tipindedir: $I + jQ$. Ancak alıcıda yüksek örnekleme yapıldığından bu hızda verinin bir dosyaya yazılması sadece $IQIQIQ \dots IQIQIQ$ formatında mümkün olmuştur.

Bu nedenle veriler $[I+jQ, I+jQ, I+jQ, I+jQ, I+jQ]$ olarak yeniden biçimlendirilmesi ve ağıın girdisi olarak ele alınması gerektiği düşünülmüştür. Veri ön işleme sürecinde, ham verinin düzenlenmesinden ağa girişe kadar olan süreçte yer alan adımlar şu şekildedir: ham I/Q verisini karmaşık sayı vektörüne dönüştürmek, CNN giriş veri ölçeğine uyarlamak , hedef çıktı verilerinin sınıflandırılması ile kodlanması ve eğitim-test veri seti oranının belirlenmesi [5]. Biçimlendirilmiş şekilde olan yeni giriş veri seti 28×28 boyutlarında matrisler halinde olup Denklem (3.4)'te ifade edilmiştir.

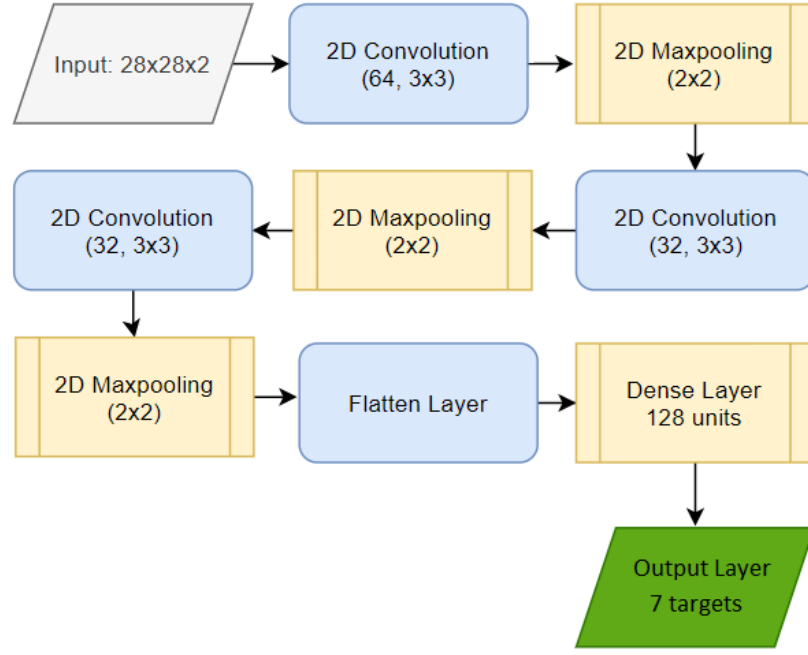
$$z^{(i)} = \begin{bmatrix} I_{1,1} & Q_{1,1} \\ I_{1,2} & Q_{1,2} \\ I_{1,3} & Q_{1,3} \\ \vdots & \vdots \\ \vdots & \vdots \\ I_{1,28} & Q_{1,28} \end{bmatrix}, \begin{bmatrix} I_{2,1} & Q_{2,1} \\ I_{2,2} & Q_{2,2} \\ I_{2,3} & Q_{2,3} \\ \vdots & \vdots \\ \vdots & \vdots \\ I_{2,28} & Q_{2,28} \end{bmatrix}, \dots, \begin{bmatrix} I_{n,1} & Q_{n,1} \\ I_{n,2} & Q_{n,2} \\ I_{n,3} & Q_{n,3} \\ \vdots & \vdots \\ \vdots & \vdots \\ I_{n,28} & Q_{n,28} \end{bmatrix}, \quad i \in \{1, 2, \dots, n\} \quad (3.4)$$

Girdi veri setinin işlenmesinde CNN ağıının pencere mantığından yararlanmak için $28 \times 28 \times 2$ boyutlarında bir üç boyutlu matris modeli kullanılmıştır. Bu durumda ağ modelinin eğitiminde giriş veri setinden 1568 değer kullanılmıştır. Eğitim ve test verilerinin veri kümesini doğru bir şekilde temsil etmesini sağlamak için eğitim ve test verilerinin oranını doğru şekilde seçmek önemlidir.

Genel olarak bir eğitim veri seti %70 ile %80 arasında belirlenir. Kalan kısımda ise test verileri belirlenir. Ayrıca veri setinin ağı daha verimli eğitebilmesi için verilerin rastgele karıştırılması işlemi de veri seti üzerinde gerçekleştirilmiştir.

3.4 Verinin Sınıflandırılması

Tasarlanan CNN modelin şeması Şekil 3.2'de blok diyagram olarak gösterilmiştir. Buradan da görüleceği üzere giriş verileri $28 \times 28 \times 2$ boyutlarında seçilmiştir. Bir sonraki evrişim katmanı, katman derinliği 64 ve giriş verisi (3×3) boyutlarında olacak şekilde tasarlandı. Sonraki evrişim katmanlarında bu değer (32, 3×3) olarak ayarlanmıştır. İki boyutlu maksimum havuzlama katmanlarının filtrelerinin boyutları (2×2) olarak seçilmiştir. Evrişim modelinin son aşamasında, tam bağlı sinir ağı modelinde giriş için tek bir katmanda 128 nöron kullanılır.



Şekil 3.8 Tasarlanan CNN Modelinin Eğitim Algoritması

Çıkış katmanında ise yedi hedef belirlenmiş olup bu hedefler 7 bit ile kodlanmıştır. Evrişim katmanlarında ReLU aktivasyon fonksiyonu kullanılmıştır. Tam bağlantılı eğitim katmanında, giriş için aktivasyon işlevi ReLU ve çıkış katmanı için Softmax işlevi kullanılmıştır. Sinir ağı eğitilirken $D = \{(z^{(1)}, y^{(1)}), \dots, (z^{(n)}, y^{(n)})\}$; eğitim seti, $z^{(i)} (i \in \{1, 2, \dots, n\})$; eğitim setinin etiket verisi ise $y^{(j)} (i \in \{1, 2, \dots, n\})$ olarak ele alındığında belirlenen bir z numune verisi için olasılık hesabı $p = (y = k | z)$ olarak ele alınmıştır. Bu durumda her nöronda softmax regresyon modeli Denklem (3.5)'teki gibi hesaplanmaktadır.

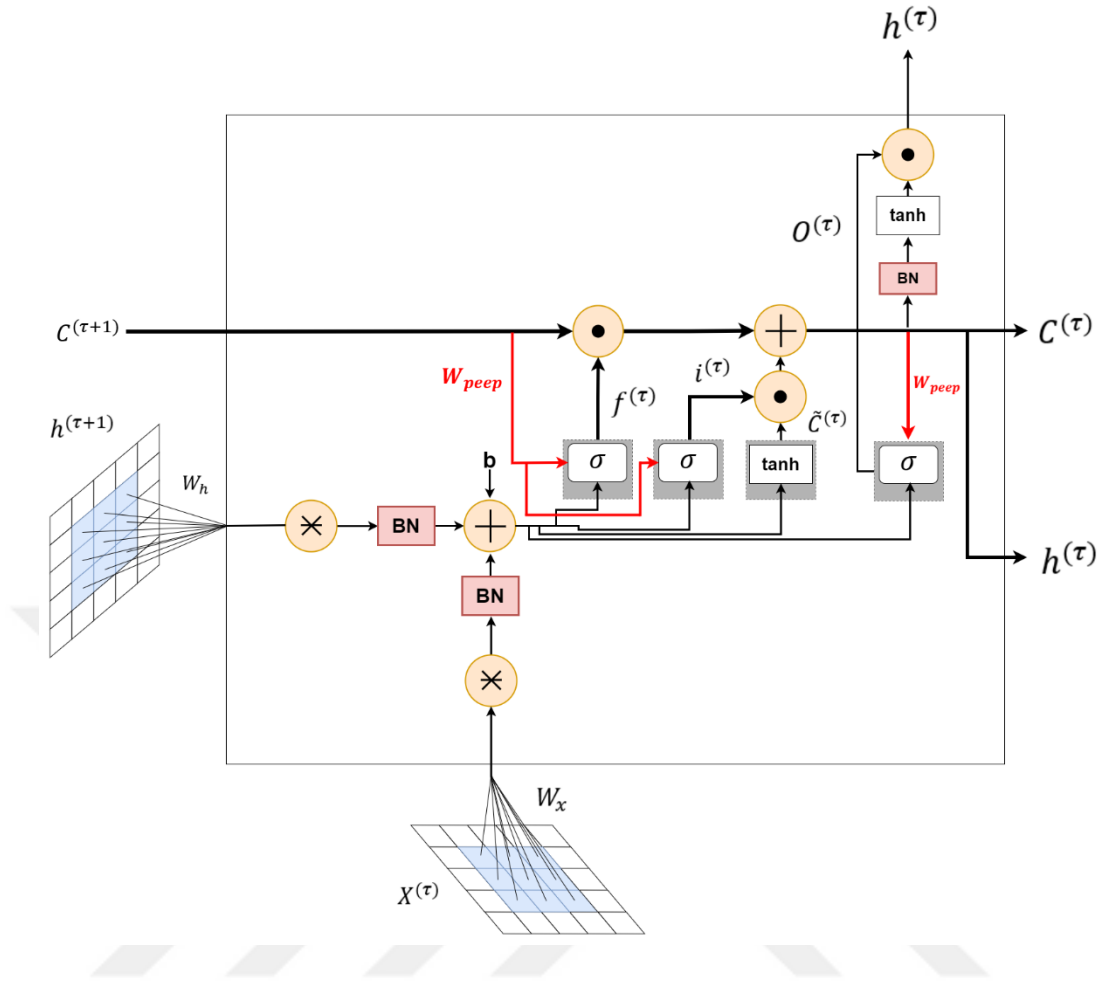
$$\begin{aligned}
 p_1(y^{(1)} = 1 | z) \\
 p_2(y^{(2)} = 2 | z) \\
 \dots \\
 p_m(y^{(m)} = m | z)
 \end{aligned} \tag{3.5}$$

CNN ağıının çıktı katmanında ise yedi hedef belirlenmiş olup bu hedefler Python uygulamasında 2'lik sistemde 7 bit ile kodlanmıştır. Evrişim katmanlarında matematiksel ifadesi $\max(0,x)$ olan ReLU aktivasyon fonksiyonu kullanılmıştır. Tam bağlantılı eğitim katmanında, giriş için aktivasyon işlevi ReLU ve çıkış katmanı için Softmax işlevi kullanılmıştır. Yaygın olarak kullanılan lojistik regresyon işlevi, 0 ile 1.0 arasında bir ondalık sayı döndürmektedir. Softmax, bu fikri çok sınıflı bir dünyaya genişleten bir yaklaşımdır. Softmax, çok sınıflı bir problemin çözümü için her bir sınıfı ondalık olasılık değeriyle temsil eder [8]. Bu ondalık olasılıkların toplamı 1.0'ı geçmemelidir. Bu sınırlama, eğitimin başka türlü olacağından daha hızlı yakınsamasına neden olur.

Optimize edici, ağı eğitiminde ve modelin derlenmesinde daha doğru sonuçlar elde etmek için kayıp fonksiyonunu en aza indirecek ağırlıkları güncellemek için kullanılır. Optimize edici olarak ADAM modeli seçilmiştir. Bu optimize edici türünde, degradelerin katlanarak azalan bir ortalaması tutulur. Adam Optimize Edicisi'ni seçmenin ana nedenlerinden biri, düşük bellek gereksinimi ve yüksek hesaplama verimliliğidir. Denklem (3.6)'da görülen m_t ve v_t , sırasıyla gradyanların ortalaması olan birinci moment ve merkezlenmemiş varyans olan ikinci moment değerleridir.

$$\widehat{m}_t = \frac{m_t}{1 - \beta_1^t}, \quad \widehat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (3.6)$$

Çalışmada kullanılan ikinci derin öğrenme yöntemi olan LSTM ağı modeli ile yapılan erken denemelerde beklenen başarımlar sağlanamamıştır. Bunun ana nedenlerinden biri, tek boyutlu zaman serileri için tasarlanan LSTM yönteminin matris verilerini düzgün bir şekilde işleyememesidir. Bu nedenle, I/Q verisi gibi çok boyutlu bir veri seti üzerinde de uygulanabilen ve bir LSTM türevidir olan ConvLSTM yönteminden faydalanılmıştır. ConvLSTM, LSTM hücresi içinde evrişim modeli gerçekleştirilen, LSTM hücre modelinin bir varyasyonudur. Giriş verisi üzerinde matris çarpım işlemini evrişim işlemiyle değiştirir. Evrişim uygulayarak çok boyutlu matris verisi üzerinde uzamsal özellikleri yakalar. ConvLSTM, hem girdiden duruma hem de durumdan duruma geçişlerde evrişimli yapılara sahiptir (Şekil 3.9).



Şekil 3.9 ConvLSTM hücre mimarisi

ConvLSTM şemasında $X^{(t)}$ ağıın girişlerini , $C^{(t)}$ hücre çıkışlarını, $h^{(t)}$ gizli katman durumlarını ve $i^{(t)}, f^{(t)}, o^{(t)}$ kavramları ise ConvLSTM'deki kapıları temsil eder [39]. Bu kavramlar ayrıca ConvLSTM yapısında satır ve sütun ile sunulan 3B tensör yapısındadırlar. Şekil 3.9'da da belirtilen ConvLSTM mimarisinin matematiksel denklemleri Denklem (3.7)'de yer almaktadır.

$$\begin{aligned}
 i^{(t)} &= \sigma(W_{xi} * X^{(t)} + W_{hi} * h^{(t-1)} + W_{ci} \circ C^{(t-1)} + b_i) \\
 f^{(t)} &= \sigma(W_{xf} * X^{(t)} + W_{hf} * h^{(t-1)} + W_{cf} \circ C^{(t-1)} + b_f) \\
 C^{(t)} &= f^{(t)} \circ C^{(t-1)} + i^{(t)} \circ \tanh(W_{xc} * h^{(t-1)} + b_c) \\
 o^{(t)} &= \sigma(W_{xo} * X^{(t)} + W_{ho} * h^{(t-1)} + W_{co} \circ C^{(t)} + b_o) \\
 h^{(t)} &= o^{(t)} \circ \tanh(C^{(t)})
 \end{aligned} \tag{3.7}$$

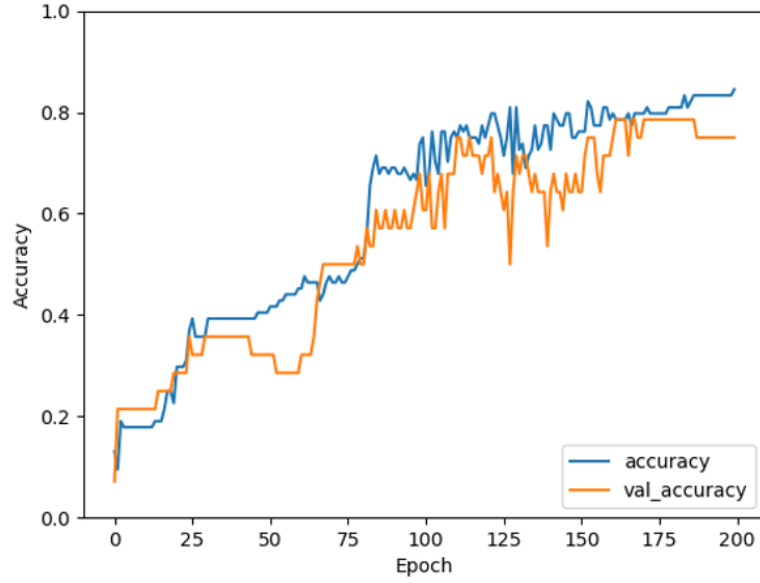
3.5 Ağların Başarımları

Çalışma kapsamında gerçek hayatta en yoğun ve sık kullanılan sinyal türlerinden yedi farklı sinyal türü ele alınmıştır. Tablo 3.1, çalışmada kullanılan sinyal türlerini ve çalışma frekanslarını göstermektedir. Her frekans bandında uygun bir anten yardımıyla alınan sinyaller rastgele olarak %75 eğitim, %25 test verisi olarak kullanılmak üzere ayrılmıştır.

Tablo 3.1 Algılanan frekanslar ve sınıf gösterimleri

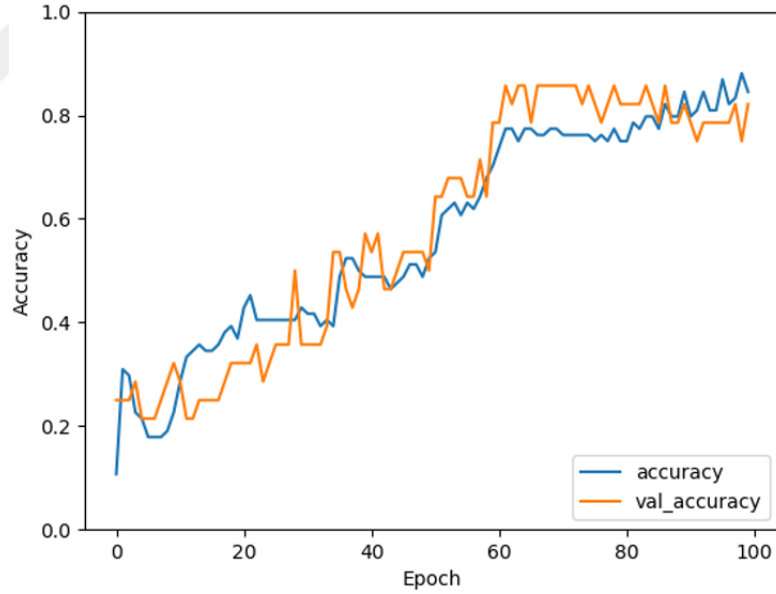
Sinyal Adı	Frekans Aralığı	Sınıf
FM Radio	87 – 108 MHz	1
ISM Band-1	433.50 – 434.790 MHz	2
ISM Band-2	863 – 870 MHz	3
LTE – Band 20	832 – 862 MHz	4
LTE – Band 3	1710 – 1785 MHz	5
Wifi 2.4 – Channel 1	2412 MHz	6
Wifi 5 – Channel 6	5180 MHz	7

Çalışmanın ilk aşamasında yoğun katman boyutu 128, yığın sayısı 256 olan bir CNN ağı 200 adım boyunca eğitilmiştir. Ağın eğitime ve doğrulama başarımlarının adım sayısına göre değişimi Şekil 3.4'te sunulmuştur. Ağın doğrulama başarımı %75 olarak ölçülmüştür. Şekil 3.4 incelendiğinde yaklaşık 100. adımdan sonra adım sayısının artmasına rağmen ağın başarımında olumlu bir gelişme olmadığı gözlenmiştir. Bu nedenle ağ, farklı parametrelerle daha kısa süreyle eğitilerek başarımının arttırılmasına çalışılmıştır.



Şekil 3.10 Model-1 CNN yapısının eğitim ve doğrulama başarımlar eğrisi

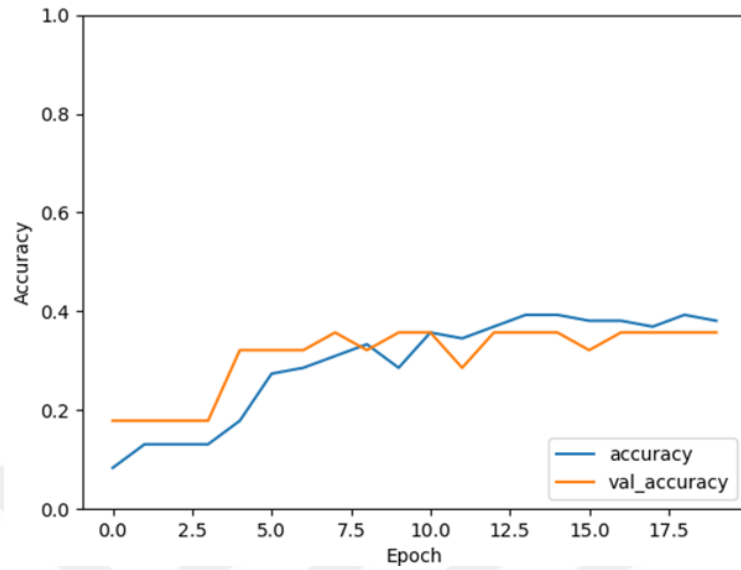
Yoğun katman boyutu 64, yığın sayısı 128 olan bir CNN ağı, 100 adım boyunca eğitildiğinde başarımının %81'e yükseldiği görülmüştür (Şekil 3.10).



Şekil 3.11 Model-2 CNN yapısının eğitim ve doğrulama başarımlar eğrisi

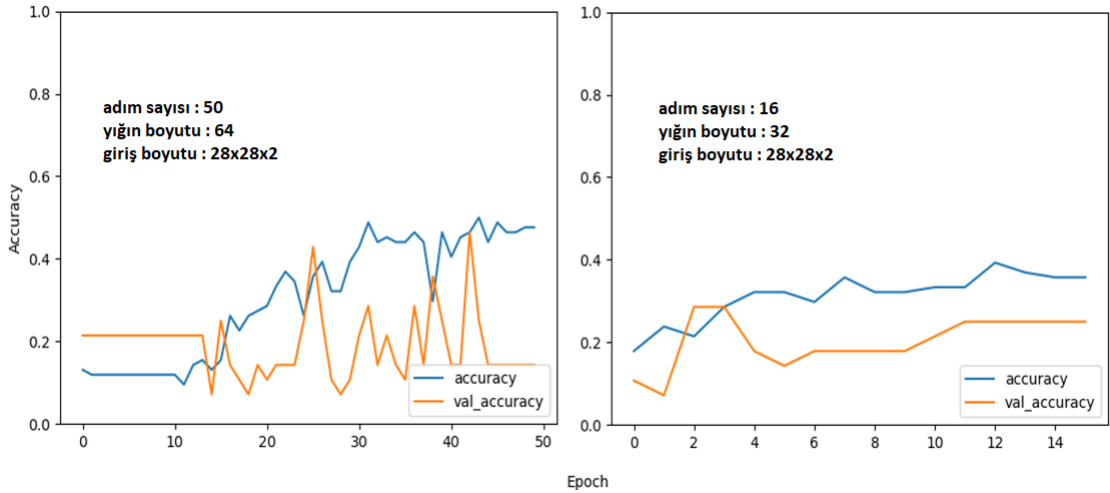
Çalışmanın son kısmında aynı veri, farklı parametreler için ConvLSTM modeli ile denenmiştir. Yapılan denemeler sonucunda elde edilen en yüksek başarımlar %40'ın altında kalmıştır. En uygun ağ parametreleri olarak adım sayısı 20, yığın boyutu 64, şekillendirilmiş giriş boyutu 28x28x2 olarak tespit edilmiştir (Şekil 3.12).

Farklı özelliklerle eğitilen CNN ağı modellerinin başarımlarını da içeren karşılaştırma verisi Tablo 3.2’de verilmiştir.



Şekil 3.12 ConvLSTM Sinir Ağı Modeli’nin başarımlar çıktısı

Bir karşılaştırma yapılabilmesi amacıyla farklı parametrelerle yapılan bazı denemelere ait sonuçlar Şekil 3.13’te sunulmuştur. Yine farklı model ağların karşılaştırılması için Tablo 3.3’e başvurulabilir.



Şekil 3.13 ConvLSTM Sinir Ağı başarımları düşük modellerin başarımlar çıktısı

Şekil 3.13 incelendiğinde ConvLSTM ağının eğitme başarımının hızlı bir şekilde doyuma ulaştığı görülmektedir. Bu durum, ağın çok çabuk bir şekilde öğrenmekten vazgeçip ezberlemeye yöneldiği şeklinde yorumlanmıştır.

Tablo 3.2 CNN ađ modellerinin sonuçları

MODEL	GİRİŞ BOYUTU	EVRIŞİM KATMANI	YOĞUN KATMAN	PARAMETRE SAYISI	BAŞARIM ORANI
CNN Model 1 Adım: 200 Yığın: 256	28x28x2	K1 – 32x3x3 K2 – 32x3x3 K3 - 64x3x3	128	160,842	0.744194
CNN Model 2 Adım: 100 Yığın: 128	28x28x2	K1 – 32x3x3 K2 – 32x3x3 K3 - 64x3x3	64	94,602	0.815162
CNN Model 3 Adım: 200 Yığın: 128	28x28x2	K1 – 32x3x3 K2 – 32x3x3 K3 - 128x3x3	64	178,634	0.618387
CNN Model 4 Adım: 200 Yığın: 128	28x28x2	K1 – 64x3x3 K2 – 64x3x3 K3 - 64x3x3	64	141,322	0.677419

Tablo 3.3 ConvLSTM ađ modellerinin sonuçları

MODEL	GİRİŞ BOYUTU	LSTM KATMANI	YOĞUN KATMAN	PARAMETRE SAYISI	BAŞARIM ORANI
ConvLSTM Model 1 Adım: 20 Yığın: 64	28x28x2	32	30 , 10, 7	93,393	0.398299
ConvLSTM Model 2 Adım: 50 Yığın: 64	28x28x2	32	30 , 10, 7	94,602	0.289210
ConvLSTM Model 3 Adım:16 Yığın: 32	28x28x2	16	30 , 10, 7	9,471	0.233180

Bu çalışmada bir SDR kullanarak kablosuz ağlar için derin öğrenme tabanlı sinyal sınıflandırması üzerinde çalışılmıştır. Günümüzde oldukça popüler olan derin öğrenme algoritmaları, sınıflandırma uygulamalarında oldukça yüksek başarı göstermektedir. Bu çalışmada kullanılan CNN ve ConvLSTM mimarileri karşılaştırıldığında kablosuz verilerin sınıflandırılması için CNN yapısının daha uygun olduğu görülmüştür. Bu durumun en önemli sebebinin, I/Q veri yapısının iki boyutlu olması olduğu düşünülmektedir. CNN modelinin iki boyutlu sinyaller için tasarlanmış olması bu farkı doğurmuştur. Çalışmada mümkün olduğunca farklı frekans türleri kullanılmış ve bu frekanslarda gerçek hayat vericilerinden yararlanılarak gerçek zamanlı veriler toplanmıştır. Bunu yaparken farklı frekans aralıkları için kullanılan antenlerin niteliklerinin (kazanç, frekans hassasiyeti vs.) önemi anlaşılmıştır. Modelin başarımında büyük rol oynayan giriş verileri, sinyalin özelliklerini olabildiğince açık bir şekilde göstermelidir. Frekans arttıkça özellikle WiFi sinyallerinde yüksek başarımlı antenler kullanılarak veri toplanmasının bu sinyallerin özelliklerini daha belirgin hale getireceği öngörülmektedir. Her ne kadar ConvLSTM modelinin başarımı açık şekilde düşük kalmış olsa da, modelin başarımının artırılabilceği düşünülmektedir. Bunun için ayrıca bir çalışma yapılması önerilmektedir. Başarım oranı yüksek CNN mimarisi için ise farklı frekanslarda veri ekleyerek ve veri miktarını arttırarak başarımın arttırılması planlanmaktadır. Ayrıca dinlenen frekanslara özellikle günümüzde herhangi bir regülasyonu olmayan bazı yayınların da tespit edilebilmesi için frekans eklemeleri yapılabilir. Çalışmanın bu noktada kapsamını artırmak sahadaki ihtiyaçlara göre gerçekleştirilebilir. Son olarak gelecek çalışmalarda alıcı cihazda yüksek kazançlı gelişmiş model antenlerin kullanılarak alıcı anteninin ağı başarımına olan etkisinin ölçülmesi planlanabilir. Ayrıca toplanan verinin faz hassasiyetli yükselteçler veya hata olasılığını algılayıp görüntüleri iyileştirecek yapılar tasarlanarak giriş verisinin netliğinin artırılması ve sınıflandırma doğruluğu

artırılması çalışmaları da hedeflenmektedir. Tüm bu işlemlerin ve tasarlanması tavsiye edilen veya hedeflenen yapıların temel bant düzeyinde demodülasyon süreçlerine işlem gücü harcamadan gerçekleştirilebilmesinin mümkün olduğu gösterilmiştir. Bu sayede yüksek donanımlı sunucu makineleri yerine kenar seviye cihazlarda tüm yapının uygun bir mimari ile çalıştırılabileceği önerilmiştir.



- [1] T. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in International Conference on Engineering Applications of Neural Networks. Springer, 2016, pp. 213–226.
- [2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, pp. 1097–1105, 2012.
- [3] N. Malik, "Artificial Neural Networks and their Applications" National Conference on 'Unearthing Technological Developments & their Transfer for Serving Masses' GLA ITM, Mathura, India 17-18 April 2005
- [4] <https://yapayzeka.ai/yapay-sinir-aglarinin-calisma-prensibi/> , Erişim Tarihi: 10.12.2021
- [5] <https://www.upgrad.com/blog/data-preprocessing-in-machine-learning/> , Erişim Tarihi: 18.11.2021
- [6] M. Abadi, P. Barham, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, and M. Kudlur, "Tensorflow: A system for large-scale machine learning," in Proc. 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 2016
- [7] <https://hackrf.readthedocs.io/en/latest/index.html> , Erişim Tarihi: 18.11.2021
- [8] C.E. Nwankpa, W. Ijomah, A. Gachagan and S. Marshall, "Activation Functions: Comparison of Trends in Practice and Research for Deep Learning ", 2nd International Conference on Computational Sciences and Technologies, 2018
- [9] E. Blossom, "Gnu radio: tools for exploring the radio frequency spectrum," Linux journal, vol. 2004, no. 122, p. 4, 2004.
- [10] K. B. Rasmussen, S. Capkun, "Implications of Radio Fingerprinting on the Security of Sensor Networks," in Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007. Third International Conference on. IEEE, 2007, pp. 331–340.
- [11] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in Proceedings of the 14th ACM international conference on Mobile Computing and Networking (MobiCom). ACM, 2008, pp. 116–127.
- [12] F. Xie, H. Wen, Y. Li, S. Chen, L. Hu, Y. Chen, and H. Song, "Optimized coherent integration-based radio frequency fingerprinting in internet of things," IEEE Internet of Things Journal, vol. 5, no. 5, pp. 3967–3977, Oct 2018.

- [13]L. Peng, A. Hu, J. Zhang, Y. Jiang, J. Yu, and Y. Yan, "Design of a hybrid RF fingerprint extraction and device classification scheme," *IEEE Internet of Things Journal*, pp. 1–1, 2018.
- [14]N. T. Nguyen, G. Zheng, Z. Han, and R. Zheng, "Device fingerprinting to enhance wireless security using nonparametric Bayesian Method," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 1404–1412.
- [15]<https://tvsempre.com.br/2019/12/21/deep-learning/> , Erişim Tarihi: 20.12.2021
- [16]F. Restuccia, T. Melodia, "Big data goes small: Real-time spectrum driven embedded wireless networking through deep learning in the RF loop," in *sTo appear in Proceedings of IEEE INFOCOM 2019*.
- [17]T. O'Shea, J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, 2017.
- [18]T. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, Feb 2018.
- [19]Y. Bengio. & Y. Lecun," Convolutional Networks for Images, Speech, and Time-Series", 1997.
- [20]<http://www.wildml.com/2015/10/recurrent-neural-networks-tutorial-part-3-backpropagation-through-time-and-vanishing-gradients> , Erişim Tarihi: 02.12.2021
- [21]<http://curiously.com/data-science/2017/05/23/tensorflow-for-hackers-part-5.html> , Erişim Tarihi: 02.12.2021
- [22]Transmission standards for FM sound broadcasting at VHF". ITU Rec. BS.450. International Telecommunication Union. pp. 4–5. Archived from the original on 2012-11-06. Retrieved 2011-01-08.dsdsd
- [23]S. Ó Dúill, M. O. Sahni, S. Trebaol, P. Landais, L. Bramerie, S. Murdoch, P. Besnard , and L. Barry, "Numerical investigation of a feed-forward linewidth reduction scheme using a mode-locked laser model of reduced complexity" *Applied Optics*. 57. E89. 10.1364/AO.57.000E89, 2018
- [24]T. Umeki, O. Tadanaga, M. Asobe, Y. Miyamoto, and H. Takenouchi, "First demonstration of high-order QAM signal amplification in PPLN-based phase sensitive amplifier", *Optics express*. 22. 2473-82. 10.1364/OE.22.002473, 2014
- [25]K. Mueller, M. Muller, "Timing Recovery in Digital Synchronous Data Receivers," in *IEEE Transactions on Communications*, vol. 24, no. 5, pp. 516-531, May 1976, doi: 10.1109/TCOM.1976.1093326.
- [26]T. O'Shea, Nathan Wast, "Radio Machine Learning Dataset Generation with GNU Radio", in *GNU Radio Conference*, Sep. 2016.
- [27]S. Hochreiter, J. Schmidhuber, "Long Short-term Memory. Neural computation", 1997, 9. 1735-80. 10.1162/neco.1997.9.8.1735.

- [28]L.D. Loganathan, R. Rengaraj, G. Konganathan, and V. Varatharajan, “Physical Layer Security using an Adaptive Modulation Scheme for Improved Confidentiality”, in proceeding of IET Journals, 2019
- [29]L. Wei, H.M. Furqan, J.M. Hamamreh, and H. Arslan, “Adaptive OFDM-IM for Enhancing Physical Layer Security and Spectral Efficiency of Future Wireless Networks”, *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 3178303, 16 pages, 2018. <https://doi.org/10.1155/2018/3178303>
- [30]Bilgi Teknolojileri ve İletişim Kurumu, “MİLLİ FREKANS PLANI”, Spektrum Yönetimi Dairesi Başkanlığı, 2019
- [31]https://cezannec.github.io/Convolutional_Neural_Networks/, Erişim Tarihi: 22.10.2021
- [32]S. Riyaz, K. Sankhe, S. Ioannidis and K. Chowdhury, “Deep Learning Convolutional Neural Network for Radio Identification”, *IEEE Communications Magazine* 56(9):146-152, September 2018
- [33]E.S. Frank, Y. Zhen, F. Han, T. Shailesh, and D. Matthias, “An Introductory Review of Deep Learning for Prediction Models With Big Data”, *Frontiers in Artificial Intelligence*, vol. 3, 2020
- [34]S. Shi, Q. Wang, P. Xu, and X. Chu, “Benchmarking state-of-the-art deep learning software tools”, In *Cloud Computing and Big Data (CCBD)*, 2016 7th International Conference on, pages 99–104. IEEE, 2016.
- [35]X. Zhang, Y. Wang, “pCAMP: Performance Comparison of Machine Learning Packages on the Edges.
- [36]M. Schmidt, D. Block, and U. Meier, “Wireless interference identification with convolutional neural networks,” *arXiv:1703.00737*, 2017.
- [37]Y. Xing, A. Hu, J. Zhang, L. Peng, and G. Li, “On radio frequency fingerprint identification for DSSS systems in low SNR scenarios,” *IEEE*
- [38]J. Walrand, P. Varaiya, “High-Performance Communication Networks (Second Edition)” 2000, Pages 305-361
- [39]X. Shi, Z. Chen, W. Hao, D.Y. Yeung, W. K. Wong, and W. Woo. “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting”, 2015

TEZDEN ÜRETİLMİŞ YAYINLAR

Konferans Bildirileri

Performance of the Convolutional Neural Network Model in Identification of the Wireless Frequency Bands , 2021 13th International Conference on Electrical and Electronics Engineering (ELECO), IEEE Explore

