

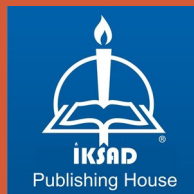


DİSİPLİNLERARASI AKADEMİK ÇALIŞMALAR II

EDİTÖRLER

Doç. Dr. Fatma KUNCAN

Dr. Öğr. Üyesi Yağmur AKIN YILDIRIM



DİSİPLİNLERARASI AKADEMİK ÇALIŞMALAR II

EDİTÖRLER

Doç. Dr. Fatma KUNCAN¹

Dr. Öğr. Üyesi Yağmur AKIN YILDIRIM²

YAZARLAR

Doç. Dr. Ramazan TEKİN

Doç. Dr. Sıtkı ÖZTÜRK

Dr. Öğr. Üyesi Mahmut DURMAZ

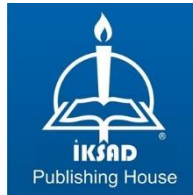
Dr. Öğr. Üyesi Mehmet Ali ALTUNCU

Bahar ÇİFTÇİ

Kübra KAYA

Mohammed Elkhadir CHACHOUA

Sefa KENCO



¹ (ORCID: 0000-0003-0712-6426)

² (ORCID: 0000-0003-1332-1914)

Copyright © 2024 by iksad publishing house
All rights reserved. No part of this publication may be reproduced, distributed
or transmitted in any form or by
any means, including photocopying, recording or other electronic or
mechanical methods, without the prior written permission of the publisher,
except in the case of
brief quotations embodied in critical reviews and certain other noncommercial
uses permitted by copyright law. Institution of Economic Development and
Social

Researches Publications®
(The Licence Number of Publicator: 2014/31220)

TÜRKİYE TR: +90 342 606 06 75

USA: +1 631 685 0 853

E mail: iksadyayinevi@gmail.com

www.iksadyayinevi.com

It is responsibility of the author to abide by the publishing ethics rules. The
first degree responsibility of the works in the book belongs to the authors.

Iksad Publications – 2024©

ISBN: 978-625-367-840-1

Cover Design: İbrahim KAYA

September/ 2024

Ankara / Türkiye

Size = 16x24 cm

İÇİNDEKİLER

ÖNSÖZ.....1

BÖLÜM 1

DİYABETİK RETİNOPATİ SINIFLANDIRILMASI İÇİN DERİN ÖĞRENME UYGULAMALARI

Dr. Öğr. Üyesi Mehmet Ali ALTUNCU.....3

BÖLÜM 2

PYTHON'DA DERİN ÖĞRENME ALANINDA KULLANILAN UYGULAMA GELİŞTİRME ÇATILARI

Bahar ÇİFTÇİ

Doç. Dr. Ramazan TEKİN.....27

BÖLÜM 3

DERİN ÖĞRENME İLE PLC KONTROLLÜ DELTA ROBOT PET ŞİŞE AYRIŞTIRMA SİSTEMİ UYGULAMASI

Mohammed Elkhadir CHACHOUA

Doç. Dr. Sıtkı ÖZTÜRK.....61

BÖLÜM 4

YAPISAL DİRİLİŞ: BETONARME YAPILARDA KOLON ONARIM VE GÜÇLENDİRME YÖNTEMLERİ

Dr. Öğr. Üyesi Mahmut DURMAZ

Kübra KAYA.....95

BÖLÜM 5

BETONARME YAPILARDA DÖŞEMELERDE DEPREM HASARLARINI ONARIM VE GÜÇLENDİRME YÖNTEMLERİ

Dr. Öğr. Üyesi Mahmut DURMAZ

Sefa KENCO.....111

ÖNSÖZ

Son zamanlarda farklı alanlardaki disiplinlerarası çalışmalar ön plana çıkmaktadır. Farklı alanlardaki disiplinlerarası çalışmaların ortaya çıkmasındaki en önemli sebeplerden birisinin gelişen teknolojik uygulamalar ve yeni ihtiyaçlar olduğundan söz edilebilir. Ayrıca farklı alanlarda gerçekleştirilen disiplinlerarası çalışmaların hem akademik hem de uygulama alanında geniş bir öneme ve değere sahip olduğu görülmektedir. Bu nedenle son zamanlarda hem akademisyenler hem araştırmacılar hem de diğer alanlarda çalışan insanların her alanda disiplinlerarası çalışmalara önem verdiği görülmektedir. Bu kitap önceki yıllarda yayınlanan DİSİPLİNLERARASI AKADEMİK ÇALIŞMALAR isimli kitabın devamı niteliğinde olarak yeni ve güncel bölümler ile DİSİPLİNLERARASI AKADEMİK ÇALIŞMALAR II olarak yayınlanmıştır. Hem akademisyenlerin hem öğrencilerin hem araştırmacıların hem de ilgi duyan ve duyacak herkese katkı sağlaması temenni edilmektedir.

Kitabımızın birinci bölümünde son zamanlarda farklı alanda ve farklı uygulamalarda sıklıkla karşılaştığımız Yapay Zeka Uygulamalarından bir örnek olarak söylenebilecek bir çalışmanın TIP alanıyla entegre edilerek DİYABETİK RETİNOPATİ SINIFLANDIRILMASI İÇİN DERİN ÖĞRENME UYGULAMALARI isimli bir çalışma yapılmıştır.

Kitabımızın ikinci bölümünde son zamanlarda farklı alanda ve farklı uygulamalarda sıklıkla karşılaştığımız Yapay Zeka Uygulamalarından bir örnek olarak söylenebilecek bir bölüm olan PYTHON'DA DERİN ÖĞRENME ALANINDA KULLANILAN UYGULAMA GELİŞTİRME ÇATILARI isimli bir çalışma yapılmıştır.

Kitabımızın üçüncü bölümünde son zamanlarda farklı alanda ve farklı uygulamalarda sıklıkla karşılaştığımız Yapay Zeka Uygulamalarından bir örnek olarak söylenebilecek bir çalışmanın Endüstriyel Sanayi Uygulama alanıyla entegre edilerek DERİN ÖĞRENME İLE PLC KONTROLLÜ DELTA ROBOT PET ŞİŞE AYRIŞTIRMA SİSTEMİ UYGULAMASI isimli bir çalışma yapılmıştır.

Kitabımızın dördüncü bölümünde ise YAPISAL DİRİLİŞ: BETONARME YAPILARDA KOLON ONARIM VE GÜÇLENDİRME YÖNTEMLERİ isimli bir çalışma yapılmıştır. Ülkemizin fay hatlarının yoğun olduğu bir bölgede olması sebebiyle bu çalışmanın hem bilimsel hem de

farkındalık anlamında önemli bir çalışma olduğu düşünülmektedir. Türkiye'nin önemli deprem kuşakları arasında yer alması ve geçmişine de dönüp bakıldığında bu kuşakların etkisinde gerçekleşen depremlerde ciddi can ve mal kayıpları yaşandığı da aşikar olduğu bilinmektedir.

Kitabımızın beşinci bölümünde ise BETONARME YAPILARDA DÖŞEMELERDE DEPREM HASARLARINI ONARIM VE GÜÇLENDİRME YÖNTEMLERİ isimli bir çalışma yapılmıştır. Ülkemizin fay hatlarının yoğun olduğu bir bölgede olması sebebiyle bu çalışmanın hem bilimsel hem de deprem sonrasında meydana gelen hasarların onarımı konusunda önemli bir çalışma olduğu düşünülmektedir.

Kitabın yazımında ve hazırlanmasında emek veren ve desteği olan başta bölüm yazarları olmak üzere herkese katkıları için teşekkür ederiz.

EDİTÖRLER

Doç. Dr. Fatma KUNCAN

Dr. Öğr. Üyesi Yağmur AKIN YILDIRIM

BÖLÜM 1

DİYABETİK RETİNOPATİ SINIFLANDIRILMASI İÇİN DERİN ÖĞRENME UYGULAMALARI

Dr. Öğr. Üyesi Mehmet Ali ALTUNCU¹
(0000-0002-2948-3937)

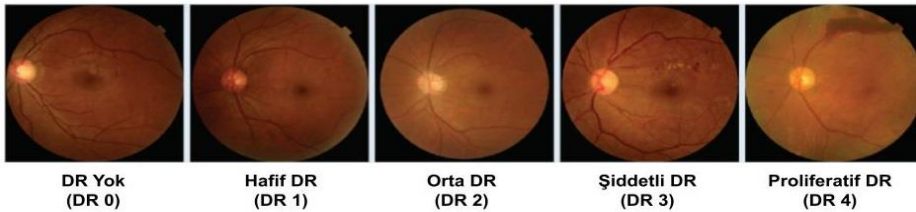
DOI: <https://dx.doi.org/10.5281/zenodo.13825213>

¹ Kocaeli Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Kocaeli, Türkiye, mehmetali.altuncu@kocaeli.edu.tr

1. GİRİŞ

Diyabet, dünya genelinde ölüm oranlarında endişe verici bir artışa yol açan kronik bir hastalık olarak bilinmektedir. Diyabet şu anda dünya genelinde 463 milyon insanı etkilemekte olup, 2045 yılına kadar bu rakamın 700 milyona ulaşacağı tahmin edilmektedir. Bu durum, hastalığı önemli bir küresel sağlık sorunu haline getirmektedir. Tedavi edilmediğinde diyabet, potansiyel olarak ölümcül sonuçlara yol açabilir. Diyabet, böbrekler, gözler, sinirler ve diğer organları etkileyerek ciddi sağlık sorunlarına neden olabilir. Diyabetin zararlı etkilerinden biri de gözlerin retinasında ortaya çıkar ve hastalığın ileri evrelerinde ciddi görme kaybına neden olabilir. Diyabetin gözlerde yarattığı bir duruma Diyabetik Retinopati (DR) denilmektedir. Diyabet hastalarının yaklaşık üçte biri, ciddi bir görme sorunu olan DR ile mücadele etmektedir. DR, kanda uzun süreli normalden yüksek glikoz (şeker) seviyesi nedeniyle retinada yavaş yavaş gelişen damar anormalliklerinden kaynaklanır ve on yılı aşkın süredir diyabetle yaşayan bireylerde yaygın olarak görülmektedir. Bu kronik durum, retinadaki kan damarlarının zarar görmesiyle ortaya çıkar ve zamanla ciddi sinir hasarına ve nihayetinde körlüğe yol açabilir. Erken teşhis ve tedavi ile bu ciddi sonuçların önüne geçilebilir, bu nedenle düzenli göz kontrolleri ve kan şekeri düzeylerinin yönetimi hayati öneme sahiptir (Saini ve Suan, 2022, Naz ve ark., 2023, Shaik ve ark., 2022).

DR derecelendirmesi, fundus görüntülerindeki retinanın yüzeyindeki lezyonların boyutları, sayıları ve çeşitleri kullanılarak sınıflandırılabilir. Uluslararası Klinik DR Hastalık Şiddeti Ölçeği'ne göre (Wilkinson ve ark., 2003), DR şiddetlerine göre derecelendirilmektedir. Bunlar, DR yok (DR 0), hafif DR (DR 1), orta DR (DR 2), şiddetli DR (DR 3) ve proliferatif DR (DR 4) olmak üzere beş sınıfa ayrılmaktadır (Wu ve ark., 2020). DR şiddet derecelerine ait temsili örnekler Şekil 1'de gösterilmektedir.



Şekil 1. Farklı şiddet derecelerindeki DR temsili örnekleri (Wang ve ark., 2022)

DR için otomatik tarama işlemi, gelişmiş derin öğrenme teknikleri kullanılarak gerçekleştirilebilir. Bu teknikler, yüksek verimlilik sunar ve sonuçlar hızla elde edilebilir. Nitelikli personel gerektirmediği için bilgisayar destekli tarama, maliyetleri önemli ölçüde azaltır. Son yıllarda, otomatik sınıflandırma görevleri için çeşitli makine öğrenimi teknikleri yaygın olarak kullanılmaktadır. Görüntü sınıflandırma işlemi, ön işleme aşamasında standart bir prosedürü takip eder. Bu aşamada, konvolüsyonel katmanlar kullanılarak görüntü veri setlerinden önemli özellikler çıkarılır. Konvolüsyonel Sinir Ağları (CNN), bu süreçte merkezi bir rol oynar. CNN'ler, görüntülerin özelliklerini otomatik olarak öğrenme ve sınıflandırma yetenekleri sayesinde, hastalıkları etkili bir şekilde tanımlamak için yeni ve ileri düzey algoritmalar geliştirilmesine olanak tanır. Bu algoritmalar, göz doktorları için büyük bir avantaj sağlar; çünkü hastalıkları hızlı ve doğru bir şekilde sınıflandırarak, gerekli zaman ve çabayı önemli ölçüde azaltır. CNN'lerin sağladığı bu avantaj sayesinde, göz hastalıkları gibi karmaşık ve hassas durumların daha etkili bir şekilde yönetilmesine ve tedavi edilmesine yardımcı olur, böylece klinik karar alma süreçlerini ve hasta bakımını geliştirir (Goel ve ark., 2021, Saranya ve ark., 2023, Bodapati ve ark., 2021).

Son zamanlardaki çalışmalarda, DR şiddetlerinin derecelendirmesi için birçok derin öğrenme tabanlı model geliştirilmiş ve sunulmuştur. Bazı çalışmalarda, geleneksel makine öğrenimi yöntemlerinin aksine, derin öğrenme modelleri fundus görüntülerinden otomatik olarak özellik çıkarımı yapılmıştır. Bazı çalışmalarda, 2. bölümde anlatılan DR tespitinde kullanılan açık kaynak veri setlerindeki sınıflar arasındaki dengesizliği ortadan kaldırmak için dikkat mekanizması yöntemlerine başvurulmuştur. Bazı çalışmalarda ise, önceden eğitilmiş CNN'ler üzerinde transfer öğrenme kullanılarak derin öğrenme özellikleri çıkarılmış ve bu özellikler kullanılarak DR şiddetlerinin otomatik olarak derecelendirilmesi amaçlanmıştır. Bu çalışmada, literatürdeki son dönem popüler DR derecelendirme çalışmalarının performansları, doğruluk ve AUC parametreleri açısından değerlendirilmiştir. Bölüm 2, bu çalışmalarda kullanılan veri setlerine dair detaylı bilgi sunmaktadır. Bölüm 3'te, incelenen yöntemler ve bu yöntemlerin performansları analiz edilmiştir. Son olarak, Bölüm 4, gelecekteki araştırmalar için bazı öneriler ve yönlendirmelerle tamamlanmaktadır.

2. DR VERİ SETLERİ

İncelenen makalelerde 6 farklı veri seti kullanılmıştır. Veri setlerinin tamamı fundus görüntülerinden oluşmaktadır ve sınıflar arasındaki görüntü sayıları dengesizdir. Bu yüzden birçok çalışmada veri setlerini dengeli hale getirmek için veri arttırma teknikleri uygulanmıştır. Veri setleri hakkında detaylı bilgiler alt bölümlerde verilmiştir.

2.1. DDR

DDR veri seti, 2016 ile 2018 yılları arasında Çin'in 23 ilindeki 147 hastaneden, yaşları 1 ile 100 arasında değişen 9598 hastadan toplanmasıyla elde edilmiştir. Veri setinde 1151 derecelendirilemeyen, 6266 normal ve 6256 DR olmak üzere, toplamda 13.673 renkli fundus görüntüsü bulunmaktadır. Görüntüler, DR şiddetine göre yok, hafif, orta, şiddetli ve proliferatif DR olmak üzere beş sınıfa ayrılmıştır. Bu veri setinde düşük kalitedeki görüntüler yer almamaktadır ve tüm görüntüler siyah arka planı silmek için önceden işlenmiştir (Li ve ark., 2019).

2.2. IDRİD

IDRİD veri seti, Hindistan'daki bir göz hastanesinde yapılan klinik testlerden elde edilen toplam 516 renkli fundus görüntüsünden oluşmaktadır. Görüntüler 4288x2848 piksel çözünürlükte çekilmiş ve JPG formatında saklanmıştır. Veri seti, DDR veri setindeki gibi hastalık şiddetine göre beş sınıfa ayrılan DR belirtilerini içermektedir (Nasir ve ark., 2021).

2.3. APTOS 2019

APTOS 2019, yine Hindistan'daki bir göz hastanesinden toplanan veri setidir. Bu veri seti de beş sınıftan oluşmaktadır. 1805 DR içermeyen, 370 hafif, 999 orta, 193 şiddetli ve 295 proliferatif DR olmak üzere toplam 3662 görüntü içermektedir (Bodapati ve ark., 2020).

2.4. Kaggle EyePACS

Kaggle EyePACS, California ve diğer bölgelerdeki çeşitli sağlık hizmetleri merkezlerinden farklı cihazlarla elde edilen yüksek çözünürlüklü fundus görüntüsü içermektedir. Veri seti, incelenen veri setleri arasında en çok görüntüye sahip olanıdır. 25810 DR içermeyen, 2443 hafif, 5292 orta, 873 şiddetli ve 708 proliferatif DR olmak üzere toplam 35126 görüntü

içermektedir. Veri setinin, gerçek dünya senaryosunu daha iyi simüle edilmesini sağlamak için gerçek etiketler gürültü içerecek şekilde düzenlenmiştir (Tsiknakis ve ark., 2021).

2.5. Messidor

Messidor 2005-2006 yılında Fransa'daki 3 göz sağlığı ve hastalıkları departmanı tarafından toplanmış 1200 fundus görüntüsünden oluşmaktadır (Tsiknakis ve ark., 2021). 548 DR içermeyen, 152 hafif, 246 orta ve şiddetli, 254 proliferatif DR olmak üzere toplam 1200 görüntü içermektedir (Nneji ve ark., 2022).

2.6. Messidor-2

Messidor-2 ise, Messidor veri setinden 1.058 görüntü ile 2009 ve 2010 yılları arasında Fransa'daki Brest Üniversitesi Hastanesi Oftalmoloji departmanında toplanmış 690 ek görüntüyü içermektedir. DR içermeyen, hafif, orta, şiddetli ve proliferatif DR olmak üzere toplam 1748 görüntü içermektedir (Fatima ve ark., 2022).

3. İLGİLİ VER SETLERİYLE YAPILAN ÇALIŞMALAR

DR derecelendirilmesi için derin öğrenme modellerinin kullanımını öneren 55 makale incelenmiştir. Bu makalelerden 20'si DDR veri setini, 14'ü IDRID veri setini, 21'i APTOS 2019 veri setini, 16'sı EYEPACS veri setini, 18'i Messidor veri setini ve 7'si Messidor-2 veri setini kullanmışlardır. Bazı çalışmalarda tek bir veri seti kullanılmamış; önerilen yöntemin performansı, bu veri setlerinden birden fazlasının kullanılmasıyla değerlendirilmiştir.

Tablo 1, DDR veri setini kullanılan çalışmalara ait sonuçları göstermektedir. En iyi doğruluk değeri %98.24 ile Obayya ve ark. (2022)'nin yaptığı çalışmada elde edilmiştir. AUC değeri verilen çalışmalarda ise en yüksek skor 0.99 ile Ashwini ve Dash (2023)'in yaptığı çalışmada elde edilmiştir.

Tablo 1. DDR Veri Seti Kullanılan Çalışmalarda Elde Edilen Sonuçlar

Referans	Veri Dengeleme İşlemi	Önerilen Yöntem	Doğruluk (%)	AUC
He ve ark., (2020)	Var	Dikkat mekanizması + DenseNet-121	78.98	-
Wang ve ark. (2021)	-	Gradyan tabanlı çok ölçekli ağ mimarisi	82.5	-
Alyoubi ve ark., (2021)	Var	CNN + YOLOv3	89.0	97.0
Saini ve Susan (2022)	Yok	VGG19	76.46	0.913
Da Rocha ve ark., (2022)	Var	Optimize edilmiş VGG16	85.4	-
Nanda ve Duraipandian (2022)	Var	Gradyan tabanlı optimizasyon algoritması + InceptionV3	86.7	-
Zhao ve ark., (2022)	Var	Xception Ağı ile Bağlamsal Transformatör	83.1	-
Obayya ve ark., (2022)	-	U-Net + SqueezeNet + GRU (Kapılı Tekrarlama Ünitesi) + CNN	98.24	-
Wang ve ark., (2022)	Var	Dikkat mekanizması + ResNet50	75.45	-
Mustafa ve ark., (2022)	Var	Temel Bileşen Analizi (PCA) + ResNet50 + DenseNet-121	76.81	-

Ashwini ve Dash (2023)	Var	Ayrık Dalga Dönüşümü + CNN	96.2	0.99
Tian ve ark. (2023)	Var	Dikkat ve bilgi tabanlı işbirlikçi ağ modeli	84.49	-
Liu ve Chi (2023)	Yok	Dikkat mekanizması + CNN	79.12	-
Alwakid ve ark., (2023)	Var	Görüntü iyileştirme algoritmaları + DenseNet-121	79.67	-
Gu ve ark., (2023)	Var	Dikkat mekanizması + CNN	82.35	90.18
Naz ve ark., (2023)	Var	Çekişmeli Üretken Ağ (GNN) + CNN topluluk modeli	96.1	-
Vijayan ve Venkatakrisnan (2023)	Var	EfficientNet-B0 omurga ağına sahip CNN	84.8	-
Chen ve ark., (2023)	Var	İki aşamalı derin öğrenme modeli	90.75	-
Kasim (2023)	Var	Optimize edilmiş MobileNet + Topluluk Sınıflandırıcı	82.74	-
Zang ve Ma (2024)	Var	CNN + Dikkat Mekanizması	83.1	-

Tablo 2, IDRID veri setini kullanılan çalışmalara ait sonuçları göstermektedir. En iyi doğruluk değeri %99.03 ile Gayathri ve ark. (2021)'nin

yaptığı çalışmada elde edilmiştir. AUC değeri verilen çalışmalarda ise en yüksek skor 0.99 ile Goel ve ark. (2021)'nin yaptığı çalışmada elde edilmiştir.

Tablo 2. IDRID Veri Seti Kullanılan Çalışmalarda Elde Edilen Sonuçlar

Referans	Veri Dengeleme İşlemi	Önerilen Yöntem	Doğruluk (%)	AUC
Elsawah ve ark., (2020)	Var	ResNet-50 + Yapay Sinir Ağı	88.00	-
Elsawah ve ark., (2020)	Var	AlexNet + Yapay Sinir Ağı	95.73	-
Sulaiman ve ark., (2020)	Var	CNN	68.00	-
Wu ve ark., (2020)	Var	Beş aşamalı CNN	56.19	0.77
Gayathri ve ark., (2020a)	Var	CNN + J48 algoritması	92.46	-
Gayathri ve ark., (2020b)	Belirtilmemiş	Hibrit iki öznitelik çıkarma yöntemi + Çok Katmanlı Algılayıcı	92.01	-
Bhardwaj ve ark., (2021a)	Var	Optimize edilmiş InceptionResnet-V2	92.38	-
Bhardwaj ve ark., (2021b)	Belirtilmemiş	InceptionV3 + İstatistiksel Analiz + SVM	90.01	0.91
Goel ve ark., (2021)	Belirtilmemiş	Modifiye edilmiş VGG16 + Lineer Regresyon	95.9	0.99
Gayathri ve ark., (2021)	Belirtilmemiş	Özgün CNN + J48 Algoritması	99.03	-

Saini ve Susan (2022)	Yok	Resnet-101	60.58	0.803
Da Rocha ve ark., (2022)	Var	Optimize edilmiş VGG16	89.8	-
Ashwini ve Dash (2023)	Var	Ayrık Dalga Dönüşümü + CNN	90.07	0.93
Liu ve Chi (2023)	Yok	Dikkat mekanizması + CNN	71.84	-

Tablo 3, APTOS 2019 veri setini kullanılan çalışmalara ait sonuçları göstermektedir. En iyi doğruluk değeri %98.7 ile Alkawid ve ark. (2023)'ün yaptığı çalışmada elde edilmiştir. AUC değeri verilen çalışmalarda ise en yüksek skor 0.988 ile Nahiduzzaman ve ark. (2023)'ün yaptığı çalışmada elde edilmiştir.

Tablo 3. APTOS 2019 Veri Seti Kullanılan Çalışmalarda Elde Edilen Sonuçlar

Referans	Veri Dengeleme İşlemi	Önerilen Yöntem	Doğruluk (%)	AUC
Taufiqurrahman ve ark., (2020)	Var	MobileNetV2 + SVM	85	0.93
Alyoubi ve ark., (2021)	Var	CNN + Seyreltme Katmanı (dropout)	84.1	0.973
Fan ve ark., (2021)	Var	Dikkat Mekanizması + Küresel Ortalama Havuzlama (GAP) + MobileNetV3	85.32	0.97
Shaik ve Cherukuri (2021)	Belirtilmemiş	Dikkat Mekanizması + Sinirsel Destek Vektör	84.31	-

		Makinesi (NSVM)		
Bodapati ve ark., (2021)	Belirtilmemiş	VGG16 + Derin Sinir Ağları (DNN)	84.31	0.97
Shaik ve Cherukuri (2022)	Belirtilmemiş	VGG16 + Dikkat Mekanizması + Uzun kısa süreli bellek (LSTM)	85.54	0.946
Oulhadj ve ark., (2022)	Belirtilmemiş	Dört Transfer Öğrenme Yönteminde Oylama (Voting) Sistemi	85.28	-
Obayya ve ark., (2022)	-	U-Net + SqueezeNet + GRU (Kapılı Tekrarlama Ünitesi) + CNN	96.00	-
Mustafa ve ark., (2022)	Var	Temel Bileşen Analizi (PCA) + ResNet50 + DenseNet-121	89.00	-
Fatima ve ark., (2022)	Var	Ayrık Dalgacık Dönüşümü + Hibrit Sinir Ağları	92.0	0.95
Zhao ve ark., (2022)	Var	Xception Ağı ile Bağlamsal Transformatör	84.18	-
Islam ve ark., (2022)	Var	Adaptif Histogram Eşitleme + Denetimli Kontrastif Öğrenme (SCL)	84.36	0.938
Fousiya ve Muneer (2022)	Var	Dikkat Mekanizması + UNet	85.8	-
Nahiduzzaman ve	Yok	Adaptif Histogram	97.27	0.988

ark., (2023)		Eşitleme + CNN + Aşırı Öğrenme Makinesi (ELM)		
Vijayan ve Venkatakrishnan (2023)	Var	EfficientNet-B0 omurga ağına sahip CNN	86.2	-
Tian ve ark., (2023)	Var	Dikkat ve bilgi tabanlı işbirlikçi ağ modeli	86.46	-
Alwakid ve ark., (2023)	Var	Görüntü iyileştirme algoritmaları + DenseNet-121	98.7	-
Mutawa ve ark., (2023)	Var	DenseNet-121	98.50	-
Kasim (2023)	Var	Optimize edilmiş MobileNet + Topluluk Sınıflandırıcı	89.86	-
Lahmar ve Idri (2023)	Var	SVM + MobileNetV2	88.80	-
Zang ve Ma (2024)	Var	CNN + Dikkat Mekanizması	89.1	-

Tablo 4, EYEPACS veri setini kullanılan çalışmalara ait sonuçları göstermektedir. En iyi doğruluk değeri %99.73 ile Saeed ve ark. (2021)'nin yaptığı çalışmada elde edilmiştir. AUC değeri verilen çalışmalarda ise en yüksek skor 0.992 ile Nneji ve ark. (2022)'nin yaptığı çalışmada elde edilmiştir.

Tablo 4. EYEPACS Veri Seti Kullanılan Çalışmalarda Elde Edilen Sonuçlar

Referans	Veri Dengeleme İşlemi	Önerilen Yöntem	Doğruluk (%)	AUC
He ve ark., (2020)	Var	Dikkat mekanizması + MobileNet1.0	86.68	-
Wu ve ark., (2020)	Var	Beş aşamalı CNN	83.10	0.84
Chetoui ve Akhloufi (2020)	Belirtilmemiş	Uçtan uca CNN	97.9	0.986
Wang ve ark., (2021)	-	Gradyan tabanlı çok ölçekli ağ mimarisi	85.7	-
Saeed ve ark., (2021)	Var	ResNet-152 + Gradyan Arttırma (GB)	99.73	0.98
Jabbar ve ark., (2022)	Var	VGG Tabanlı transfer öğrenme yöntemi	96.6	0.971
Mehboob ve ark., (2022)	Var	LSTM + CNN	83.78	-
Da Rocha ve ark., (2022)	Var	Optimize edilmiş VGG16	77.9	-
Zhao ve ark., (2022)	Var	Xception Ağı ile Bağlamsal Transformatör	84.1	-
Berber (2022)	Var	İki aşamalı CNN Mimarisi	95.06	-
Nneji ve ark., (2022)	Var	Özellik Çıkarımı (VGG16 ve Inception V3) + Derin Öğrenme	98.0	0.992
Naz ve ark.,	Var	Çekişmeli Üretken Ağ	97.4	-

(2023)		(GNN) + CNN topluluk modeli		
Ashwini ve Dash (2023)	Var	Ayrık Dalga Dönüşümü + CNN	93.53	0.91
Tian ve ark., (2023)	Var	Dikkat ve bilgi tabanlı işbirlikçi ağ modeli	86.78	-
Mutawa ve ark., (2023)	Var	DenseNet-121	89.10	-
Ishtiaq ve ark., (2023)	Var	Optimize edilmiş CNN + SVM	98.85	-

Tablo 5, Messidor veri setini kullanılan çalışmalara ait sonuçları göstermektedir. En iyi doğruluk değeri %99.75 ile Gayathri ve ark. (2021)'nin yaptığı çalışmada elde edilmiştir. AUC değeri verilen çalışmalarda ise en yüksek skor 0.995 ile Nneji ve ark. (2022)'nin yaptığı çalışmada elde edilmiştir.

Tablo 5. Messidor Veri Seti Kullanılan Çalışmalarda Elde Edilen Sonuçlar

Referans	Veri Dengeleme İşlemi	Önerilen Yöntem	Doğruluk (%)	AUC
Luo ve Feng (2020)	Var	Dikkat Mekanizması + CNN	92.9	0.965
He ve ark., (2020)	Var	Dikkat mekanizması + DenseNet-121	84.08	96.9
Shankar ve ark., (2020)	Var	Histogram tabanlı segmentasyon + Sinerjik Derin Öğrenme (SDL)	99.28	-
Pour ve ark., (2020)	-	Histogram Eşitleme + EfficientNet-B5	Belirtilmemiş	0.945

Gayathri ve ark., (2020b)	Belirtilmemiş	Hibrit iki öz nitelik çıkarma yöntemi + Çok Katmanlı Algılayıcı	98.25	-
Bhardwaj ve ark., (2021a)	Var	Optimize edilmiş InceptionResnet-V2	93.3	-
Bhardwaj ve ark., (2021b)	Belirtilmemiş	InceptionV3 + İstatistiksel Analiz + SVM	90.51	0.92
Baget-Bernaldiz ve ark., (2021)	Belirtilmemiş	Özgün Derin Öğrenme Yöntemi	94.79	0.959
Gayathri ve ark., (2021)	Belirtilmemiş	Özgün CNN + J48 Algoritması	99.75	-
Saaed ve ark., (2021)	Var	ResNet-152 + Gradyan Arttırma (GB)	98.88	0.98
Mustafa ve ark., (2022)	Var	Temel Bileşen Analizi (PCA) + ResNet50 + DenseNet-121	86.78	-
Nanda ve Duraipandian (2022)	Var	Gradyan tabanlı optimizasyon algoritması + ResNet-18	89.9	-
Gayathri ve ark., (2022a)	Var	CNN + J48 algoritması	91.22	-
Nneji ve ark., (2022)	Var	Özellik Çıkarımı (VGG16 ve Inception V3) + Derin Öğrenme	98.5	0.995
Berbar (2022)	Var	İki aşamalı CNN Mimarisi	90.83	-
Saranya ve ark., (2023)	Var	UNet	95.94	-

Kasim (2023)	Var	Optimize edilmiş MobileNet + Topluluk Sınıflandırıcı	96.06	-
Liu ve Chi (2023)	Yok	Dikkat mekanizması + CNN	78.17	-

Tablo 6 ise, Messidor-2 veri setini kullanılan çalışmalara ait sonuçları göstermektedir. En iyi doğruluk değeri %99.08 ile Bhuiyan ve ark. (2021)'nin yaptığı çalışmada elde edilmiştir. AUC değeri verilen çalışmalarda ise en yüksek skor 0.999 ile yine Bhuiyan ve ark. (2021)'nin yaptığı çalışmada elde edilmiştir.

Tablo 6. Messidor-2 Veri Seti Kullanılan Çalışmalarda Elde Edilen Sonuçlar

Referans	Veri Dengeleme İşlemi	Önerilen Yöntem	Doğruluk (%)	AUC
Chetoui ve Akhloufi (2020)	Belirtilmemiş	Uçtan uca CNN	97.9	0.962
Bhuiyan ve ark., (2021)	Belirtilmemiş	Transfer Öğrenme Yöntemleri + Lojistik Model Ağacı	99.08	0.999
Berbar (2022)	Var	İki aşamalı CNN Mimarisi	94.11	-
Mustafa ve ark., (2022)	Var	Temel Bileşen Analizi (PCA) + ResNet50 + DenseNet-121	86.78	-
Islam ve ark., (2022)	Var	Adaptif Histogram Eşitleme + Denetimli Kontrastif Öğrenme (SCL)	74.21	87.26
Fatima ve ark., (2022)	Var	Ayrık Dalgacık Dönüşümü + Hibrit Sinir Ağları	80.0	0.82

Lahmar ve ark., (2023)	Var	SVM + MobileNetV2	84.05	-
------------------------	-----	-------------------	-------	---

4. SONUÇLAR

DR derecelerinin tespit edilmesi, diyabet hastalarında görme kaybı ve körlüğe karşı önlem alınması için kritik bir öneme sahiptir. Bu çalışmada, otomatik DR derecelendirmesi için derin öğrenme tabanlı yöntemler incelenmiş ve genel olarak CNN mimarilerine dayalı hibrit modellerin kullanıldığı görülmüştür. Ancak, bazı çalışmalar etiketli örneklerin sınırlı olması ve veri dengesizliği nedeniyle hafif DR'nin tanımlanması ve sınıflandırılmasında zorluk yaşandığını belirtmiştir (Saini ve Suan, 2022, Ashwini ve Dash, 2023, Gu ve ark., 2023, Bhardwaj ve ark., 2021a, Sulaiman ve ark., 2020, Islam ve ark., 2022, Jabbar ve ark., 2022, Fatima ve ark., 2022). Daha iyi sonuçlar elde etmek için, gelecekteki çalışmalarda DR derecelendirilmesinde pekiştirmeli öğrenme veya kendi kendine gözetimli öğrenme (self-supervised learning) modelleri kullanılabilir. Bu yöntemler, modelin genel performansını artırmanın yanı sıra daha detaylı ve anlamlı tıbbi analizler yapma imkânı sunabilir. Ayrıca, DR derecelendirilmesinde kullanılan fundus görüntülerinde mevcut ön işleme yöntemleri etkili olmakla birlikte, görüntü kalitesini daha da artırmak için daha etkili tekniklerin de araştırılması gereklidir.

KAYNAKÇA

- Alwakid, G., Gouda, W., Humayun, M., & Jhanjhi, N. Z. (2023). Deep learning-enhanced diabetic retinopathy image classification. *Digital Health*, 9, 20552076231194942.
- Alyoubi, W. L., Abulkhair, M. F., & Shalash, W. M. (2021). Diabetic retinopathy fundus image classification and lesions localization system using deep learning. *Sensors*, 21(11), 3704.
- Ashwini, K., & Dash, R. (2023). Grading diabetic retinopathy using multiresolution based CNN. *Biomedical Signal Processing and Control*, 86, 105210.
- Baget-Bernaldiz, M., Pedro, R. A., Santos-Blanco, E., Navarro-Gil, R., Valls, A., Moreno, A., ... & Puig, D. (2021). Testing a deep learning algorithm for detection of diabetic retinopathy in a Spanish diabetic population and with messidor database. *Diagnostics*, 11(8), 1385.
- Berbar, M. A. (2022). Diabetic retinopathy detection and grading using deep learning. *Menoufia Journal of Electronic Engineering Research*, 31(2), 11-20.
- Bhardwaj, C., Jain, S., & Sood, M. (2021a). Deep learning-based diabetic retinopathy severity grading system employing quadrant ensemble model. *Journal of Digital Imaging*, 34, 440-457.
- Bhardwaj, C., Jain, S., & Sood, M. (2021b). Transfer learning based robust automatic detection system for diabetic retinopathy grading. *Neural Computing and Applications*, 33(20), 13999-14019.
- Bhuiyan, A., Govindaiah, A., Deobhakta, A., Hossain, M., Rosen, R., & Smith, T. (2021). Automated diabetic retinopathy screening for primary care settings using deep learning. *Intelligence-based medicine*, 5, 100045.
- Bodapati, J. D., Naralasetti, V., Shareef, S. N., Hakak, S., Bilal, M., Maddikunta, P. K. R., & Jo, O. (2020). Blended multi-modal deep convnet features for diabetic retinopathy severity prediction. *Electronics*,

9(6), 914.

- Bodapati, J. D., Shaik, N. S., & Naralasetti, V. (2021). Deep convolution feature aggregation: an application to diabetic retinopathy severity level prediction. *Signal, Image and Video Processing*, 15, 923-930.
- Chen, Y., Xu, S., Long, J., & Xie, Y. (2023). DR-Net: Diabetic retinopathy detection with fusion multi-lesion segmentation and classification. *Multimedia Tools and Applications*, 82(17), 26919-26935.
- Chetoui, M., & Akhloufi, M. A. (2020). Explainable end-to-end deep learning for diabetic retinopathy detection across multiple datasets. *Journal of Medical Imaging*, 7(4), 044503-044503.
- Da Rocha, D. A., Ferreira, F. M. F., & Peixoto, Z. M. A. (2022). Diabetic retinopathy classification using VGG16 neural network. *Research on Biomedical Engineering*, 38(2), 761-772.
- Elsawah, D., Elnakib, A., & Moustafa, H. E. D. S. (2020). Deep Learning Grading System for Diabetic Retinopathy using Fundus Images. *MEJ-Mansoura Engineering Journal*, 45(4), 1-8.
- Elsawah, D. K., Elnakib, A. A., & Moustafa, H. E. D. (2020, September). Automated diabetic retinopathy grading using resnet. In *2020 37th National radio science conference (NRSC)* (pp. 248-254). IEEE.
- Fan, R., Liu, Y., & Zhang, R. (2021). Multi-scale feature fusion with adaptive weighting for diabetic retinopathy severity classification. *Electronics*, 10(12), 1369.
- Fatima, Imran, M., Ullah, A., Arif, M., & Noor, R. (2022). A unified technique for entropy enhancement based diabetic retinopathy detection using hybrid neural network. *Computers in Biology and Medicine*, 145, 105424.
- Fousiya, T. T., & KV, A. M. (2022, November). Diabetic retinopathy classification based on segmented retinal vasculature of fundus images using attention u-net. In *2022 IEEE 19th India Council International Conference (INDICON)* (pp. 1-6). IEEE.

- Gayathri, S., Gopi, V. P., & Palanisamy, P. (2020a). A lightweight CNN for Diabetic Retinopathy classification from fundus images. *Biomedical Signal Processing and Control*, 62, 102115.
- Gayathri, S., Gopi, V. P., & Palanisamy, P. (2020b). Automated classification of diabetic retinopathy through reliable feature selection. *Physical and Engineering Sciences in Medicine*, 43(3), 927-945.
- Gayathri, S., Gopi, V. P., & Palanisamy, P. (2021). Diabetic retinopathy classification based on multipath CNN and machine learning classifiers. *Physical and engineering sciences in medicine*, 44(3), 639-653.
- Goel, S., Gupta, S., Panwar, A., Kumar, S., Verma, M., Bourouis, S., & Ullah, M. A. (2021). Deep learning approach for stages of severity classification in diabetic retinopathy using color fundus retinal images. *Mathematical Problems in Engineering*, 2021(1), 7627566.
- Gu, Z., Li, Y., Wang, Z., Kan, J., Shu, J., & Wang, Q. (2023). Classification of diabetic retinopathy severity in fundus images using the vision transformer and residual attention. *Computational Intelligence and Neuroscience*, 2023(1), 1305583.
- He, A., Li, T., Li, N., Wang, K., & Fu, H. (2020). CABNet: Category attention block for imbalanced diabetic retinopathy grading. *IEEE Transactions on Medical Imaging*, 40(1), 143-153.
- Ishtiaq, U., Abdullah, E. R. M. F., & Ishtiaque, Z. (2023). A hybrid technique for Diabetic Retinopathy Detection based on ensemble-optimized CNN and texture features. *Diagnostics*, 13(10), 1816.
- Islam, M. R., Abdulrazak, L. F., Nahiduzzaman, M., Goni, M. O. F., Anower, M. S., Ahsan, M., ... & Kowalski, M. (2022). Applying supervised contrastive learning for the detection of diabetic retinopathy and its severity levels from fundus images. *Computers in biology and medicine*, 146, 105602.
- Jabbar, M. K., Yan, J., Xu, H., Ur Rehman, Z., & Jabbar, A. (2022). Transfer learning-based model for diabetic retinopathy diagnosis using retinal images. *Brain Sciences*, 12(5), 535.

- Kasim, Ö. (2023). Ensemble classification based optimized transfer learning feature method for early stage diagnosis of diabetic retinopathy. *Journal of Ambient Intelligence and Humanized Computing*, 14(8), 11337-11348.
- Lahmar, C., & Idri, A. (2023). Deep hybrid architectures for diabetic retinopathy classification. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 11(2), 166-184.
- Li, T., Gao, Y., Wang, K., Guo, S., Liu, H., & Kang, H. (2019). Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening. *Information Sciences*, 501, 511-522.
- Liu, X., & Chi, W. (2023). A cross-lesion attention network for accurate diabetic retinopathy grading with fundus images. *IEEE Transactions on Instrumentation and Measurement*.
- Luo, L., Xue, D., & Feng, X. (2020). Automatic diabetic retinopathy grading via self-knowledge distillation. *Electronics*, 9(9), 1337.
- Mehboob, A., Akram, M. U., Alghamdi, N. S., & Abdul Salam, A. (2022). A deep learning based approach for grading of diabetic retinopathy using large fundus image dataset. *Diagnostics*, 12(12), 3084.
- Mustafa, H., Ali, S. F., Bilal, M., & Hanif, M. S. (2022). Multi-stream deep neural network for diabetic retinopathy severity classification under a boosting framework. *IEEE Access*, 10, 113172-113183.
- Mutawa, A. M., Alnajdi, S., & Sruthi, S. (2023). Transfer learning for diabetic retinopathy detection: a study of dataset combination and model performance. *Applied Sciences*, 13(9), 5685.
- Nahiduzzaman, M., Islam, M. R., Goni, M. O. F., Anower, M. S., Ahsan, M., Haider, J., & Kowalski, M. (2023). Diabetic retinopathy identification using parallel convolutional neural network based feature extractor and ELM classifier. *Expert Systems with Applications*, 217, 119557.
- Nanda, P., & Duraipandian, N. (2022). A Novel Optimizer in Deep Neural Network for Diabetic Retinopathy Classification. *Computer Systems*

Science & Engineering, 43(3).

- Nasir, N., Afreen, N., Patel, R., Kaur, S., & Sameer, M. (2021). A Transfer Learning Approach for Diabetic Retinopathy and Diabetic Macular Edema Severity Grading. *Rev. d'Intelligence Artif.*, 35(6), 497-502.
- Naz, H., Nijhawan, R., Ahuja, N. J., Al-Otaibi, S., Saba, T., Bahaj, S. A., & Rehman, A. (2023). Ensembled Deep Convolutional Generative Adversarial Network for Grading Imbalanced Diabetic Retinopathy Recognition. *IEEE Access*, 11, 120554-120568.
- Nneji, G. U., Cai, J., Deng, J., Monday, H. N., Hossin, M. A., & Nahar, S. (2022). Identification of diabetic retinopathy using weighted fusion deep learning based on dual-channel fundus scans. *Diagnostics*, 12(2), 540.
- Obayya, M., Nemri, N., Nour, M. K., Al Duhayyim, M., Mohsen, H., Rizwanullah, M., ... & Motwakel, A. (2022). Explainable artificial intelligence enabled TeleOphthalmology for diabetic retinopathy grading and classification. *Applied Sciences*, 12(17), 8749.
- Oulhadj, M., Riffi, J., Chaimae, K., Mahraz, A. M., Ahmed, B., Yahyaouy, A., ... & Tairi, H. (2022). Diabetic retinopathy prediction based on deep learning and deformable registration. *Multimedia Tools and Applications*, 81(20), 28709-28727.
- Pour, A. M., Seyedarabi, H., Jahromi, S. H. A., & Javadzadeh, A. (2020). Automatic detection and monitoring of diabetic retinopathy using efficient convolutional neural networks and contrast limited adaptive histogram equalization. *IEEE Access*, 8, 136668-136673.
- Saeed, F., Hussain, M., & Aboalsamh, H. A. (2021). Automatic diabetic retinopathy diagnosis using adaptive fine-tuned convolutional neural network. *IEEE Access*, 9, 41344-41359.
- Saini, M., & Susan, S. (2022). Diabetic retinopathy screening using deep learning for multi-class imbalanced datasets. *Computers in Biology and Medicine*, 149, 105989.
- Saranya, P., Pranati, R., & Patro, S. S. (2023). Detection and classification of

- red lesions from retinal images for diabetic retinopathy detection using deep learning models. *Multimedia Tools and Applications*, 82(25), 39327-39347.
- Shaik, N. S., & Cherukuri, T. K. (2021). Lesion-aware attention with neural support vector machine for retinopathy diagnosis. *Machine Vision and Applications*, 32(6), 126.
- Shaik, N. S., & Cherukuri, T. K. (2022). Hinge attention network: A joint model for diabetic retinopathy severity grading. *Applied Intelligence*, 52(13), 15105-15121.
- Shankar, K., Sait, A. R. W., Gupta, D., Lakshmanprabu, S. K., Khanna, A., & Pandey, H. M. (2020). Automated detection and classification of fundus diabetic retinopathy images using synergic deep learning model. *Pattern Recognition Letters*, 133, 210-216.
- Sulaiman, T., Angel Arul Jothi, J., & Bengani, S. (2020, January). Automated grading of diabetic macular edema using deep learning techniques. In *International Conference on Modelling, Simulation and Intelligent Computing* (pp. 264-272). Singapore: Springer Singapore.
- Taufiqurrahman, S., Handayani, A., Hermanto, B. R., & Mengko, T. L. E. R. (2020, November). Diabetic retinopathy classification using a hybrid and efficient MobileNetV2-SVM model. In *2020 IEEE Region 10 Conference (Tencon)* (pp. 235-240). IEEE.
- Tian, M., Wang, H., Sun, Y., Wu, S., Tang, Q., & Zhang, M. (2023). Fine-grained attention & knowledge-based collaborative network for diabetic retinopathy grading. *Heliyon*, 9(7).
- Tsiknakis, N., Theodoropoulos, D., Manikis, G., Ktistakis, E., Boutsora, O., Berto, A., ... & Marias, K. (2021). Deep learning for diabetic retinopathy detection and classification based on fundus images: A review. *Computers in biology and medicine*, 135, 104599.
- Vijayan, M. (2023). A regression-based approach to diabetic retinopathy diagnosis using efficientnet. *Diagnostics*, 13(4), 774.

- Wang, H., Sun, Y., Cao, Y., Ouyang, G., Wang, X., Wu, S., & Tian, M. (2022, March). Classification for diabetic retinopathy by using staged convolutional neural network. In *2022 Asia Conference on Algorithms, Computing and Machine Learning (CACML)* (pp. 228-233). IEEE.
- Wang, X., Xu, M., Zhang, J., Jiang, L., & Li, L. (2021, May). Deep multi-task learning for diabetic retinopathy grading in fundus images. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 4, pp. 2826-2834).
- Wilkinson, C. P., Ferris III, F. L., Klein, R. E., Lee, P. P., Agardh, C. D., Davis, M., ... & Global Diabetic Retinopathy Project Group. (2003). Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales. *Ophthalmology*, 110(9), 1677-1682.
- Wu, Z., Shi, G., Chen, Y., Shi, F., Chen, X., Coatrieux, G., ... & Li, S. (2020). Coarse-to-fine classification for diabetic retinopathy grading using convolutional neural network. *Artificial Intelligence in Medicine*, 108, 101936.
- Zang, F., & Ma, H. (2024). CRA-Net: Transformer guided category-relation attention network for diabetic retinopathy grading. *Computers in Biology and Medicine*, 170, 107993.
- Zhao, S., Wu, Y., Tong, M., Yao, Y., Qian, W., & Qi, S. (2022). Cot-xnet: contextual transformer with xception network for diabetic retinopathy grading. *Physics in Medicine & Biology*, 67(24), 245003.

BÖLÜM 2

PYTHON'DA DERİN ÖĞRENME ALANINDA KULLANILAN UYGULAMA GELİŞTİRME ÇATILARI

Bahar ÇİFTÇİ¹
(0000-0001-5976-6236)

Doç. Dr. Ramazan TEKİN²
(0000-0003-4325-6922)

DOI: <https://dx.doi.org/10.5281/zenodo.13825217>

¹ Siirt Üniversitesi, Rektörlük, Uzaktan Eğitim Uygulama ve Araştırma Merkezi, Siirt, Türkiye, bahar.ciftci@siirt.edu.tr

² Batman Üniversitesi, Mühendislik Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü, Batman, Türkiye, ramazan.tekin@batman.edu.tr

1. GİRİŞ

Son yıllarda Derin Öğrenme (DÖ), artan veri seti boyutları ve gelişen teknolojilerle birlikte büyük ilgi görmeye başlamıştır. Geçmişte, veri toplamanın zorlukları ve maliyetleri nedeniyle daha küçük veri setleri kullanılıyordu. Ancak 20. yüzyılın ortalarından itibaren Makine Öğrenmesi (MÖ) yöntemlerinin gelişmesiyle birlikte daha büyük ve karmaşık veri setleri toplanabilir hale gelmiş, bu alandaki çalışmalar hız kazanmıştır.

Teknolojik ilerlemeler, veri toplama, işleme ve saklama kapasitelerini artırmış; bu da üretilen veri miktarı ve çeşitliliğinde patlamaya yol açmıştır. DÖ, büyük veri setlerini işleyebilme kapasitesiyle öne çıkan ve birçok katmana sahip bir yapay zekâ türüdür.

Aynı zamanda, GPU (Grafik İşlem Birimi) gibi donanım geliştirmeleri, DÖ modellerinin daha verimli bir şekilde geliştirilmesine imkân tanıyan DÖ Uygulama Geliştirme Çatılarının (Framework) yaygınlaşmasına olanak sağlamıştır. Python programlama dili, DÖ alanında kullanılan birçok Uygulama Geliştirme Çatısı sunmaktadır; bu çatılar, farklı ihtiyaçlara yönelik olarak geliştirilmiş ve farklı işleme yöntemlerine sahiptir. En yaygın kullanılan Uygulama Geliştirme Çatıları arasında TensorFlow, Keras, Theano, Caffe, PyTorch ve Microsoft Cognitive Toolkit (CNTK) bulunmaktadır.

Bu çalışmanın temel amacı, DÖ alanında çalışan araştırmacılara, yaygın olarak kullanılan bu Uygulama Geliştirme Çatılarının performansları hakkında detaylı bilgiler sunmaktır. Özellikle GPU üzerinde elde edilen performans sonuçlarının, araştırmacılara uygun Uygulama Geliştirme Çatısını seçme konusunda bir kılavuz olacağı düşünülmektedir. Bu çerçevede, çalışmada PyTorch, Theano, Caffe, Keras, TensorFlow ve CNTK Uygulama Geliştirme Çatıları ayrıntılı bir şekilde ele alınmış, literatürde en popüler yöntemlerden olan TensorFlow, Keras (TensorFlow Backend) ve PyTorch için deneysel bir değerlendirme yapılmış ve performans karşılaştırmaları üzerinde durulmuştur.

2. TENSORFLOW

TensorFlow, Google Brain Ekibi tarafından 2015 yılında DÖ modelleri oluşturmak için oluşturulmuştur (Abadi et al., 2015). Bu Uygulama Geliştirme Çatısı, geliştiricilere uygulama oluşturma ve dağıtma kolaylığı sağlamasıyla popülerlik kazanmıştır. TensorFlow, işlem gücü sınırlamaları dikkate alınarak tasarlanmıştır ve her türlü bilgisayarda, hatta akıllı telefonlarda bile

çalışabilmektedir. Verileri ‘tensör’ adı verilen çok boyutlu diziler şeklinde kabul eden TensorFlow, büyük miktarda veriyi işlemek için son derece etkilidir. TensorFlow'dan önce, MÖ ve DÖ modelleri için kodlama çok daha karmaşıktı. Ancak TensorFlow, üst düzey bir API sunarak bu süreci basitleştirmiştir, böylece bir sinir ağı tasarlamak için karmaşık kodlamaya gerek kalmamaktadır. Başlangıçta Python kullanılarak geliştirilen TensorFlow, şu anda C++, Java, C#, JavaScript ve R gibi diğer programlama dilleriyle de entegre edilebilmektedir. TensorFlow'un en büyük avantajlarından biri, sadece CPU'ları değil, aynı zamanda GPU'ları da desteklemesidir. Ayrıca, Keras ve Torch gibi diğer DÖ kütüphanelerine kıyasla daha hızlı derleme sürelerine sahiptir. TensorFlow, MÖ geliştirmesi için önemli bir soyutlama sağlamaktadır. Geliştiriciler, algoritmaların temel ayrıntılarıyla uğraşmak yerine, uygulamanın genel mantığına odaklanabilmektedirler. TensorFlow, perde arkasındaki teknik detayları yönetip ve geliştiricilerin işini kolaylaştırmaktadır.

2.1. TensorFlow Çalışma Prensipleri

Tensorflow; veri akışı grafları ile sayısal hesaplamalar gerçekleştiren bir Uygulama Geliştirme Çatısıdır. Veri akışı grafları, verilerin bir dizi işlem düğümünde nasıl ilerlediğini tanımlayan yapılar oluşturmaya olanak sağlamaktadır. Her düğüm, grafta bir matematiksel işlemi temsil etmektedir. Düğümler arasındaki bağlantılar, çok boyutlu veri dizileri veya tensörler olarak ifade edilmektedir. Tensorflow mimarisi üç aşamada çalışmaktadır:

- Verilerin ön işleme
- Modelin oluşturulması
- Modelin eğitilmesi ve tahmin edilmesi

2.2. TensorFlow Bileşenleri

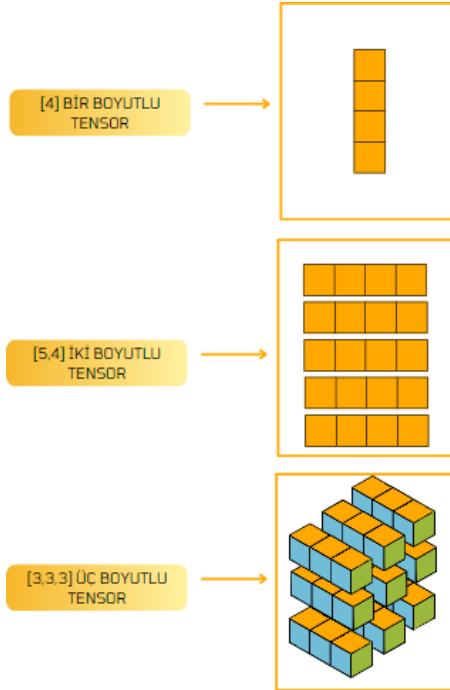
Girdiyi tensör olarak bilinen çok boyutlu bir dizi şeklinde aldığı için Tensorflow olarak adlandırılmaktadır. Girdi bir uçtan girer ve daha sonra çoklu işlemlerden geçip diğer uçtan çıktı olarak çıkmaktadır.

2.2.1. Tensörler

Bir tensör, vektör veya matrisin bir genellemesidir ve çok boyutlu bir dizi olarak gösterilebilmektedir. Ayrıca bir vektör, tek boyutlu veya birinci

dereceden bir tensördür ve bir matris, iki boyutlu veya ikinci dereceden bir tensördür. Şekil 1'de farklı tensör boyutlarına yer verilmiştir.

Tensörler; DÖ modellerinin eğitiminde ve işleyişinde MÖ' de bilinen bir terimdir. Phyton' da dizi yapısı kullanılarak temsil edilebilmektedirler.



Şekil 1. Farklı Tensör boyutu örnekleri

2.2.2. Grafikler

TensorFlow'da tüm işlemler bir graf içinde gerçekleştirilmektedir. Bu graf yapısı, ardışık hesaplama modüllerinden oluşmaktadır. Her işlem 'op node' olarak adlandırılır ve bu düğümler birbirine bağlıdır. Graf, düğümler arasındaki işlemleri ve bağlantıları özetler, ancak değerleri göstermez. Bir tensör, bir düğüm ve bir kenardan oluşmaktadır. Düğüm, matematiksel işlemi temsil etmektedir. Graf, eğitim sırasında yapılan tüm hesaplamaları gerçekleştirmek için kullanılır ve birçok avantajı vardır:

- Birden fazla CPU veya GPU'da ve hatta mobil işletim sistemlerinde çalışacak şekilde tasarlanmıştır.

- Grafın taşınabilirliği, hesaplamaların hemen veya daha sonra kullanılmak üzere korunmasına izin vermektedir. Ayrıca gelecekte kullanılmak üzere kaydedilebilmektedir.
- Graftaki tüm hesaplamalar, tensörleri birbirine bağlayarak yapılmaktadır.

2.3. TensorFlow'a Veri Yükleme Seçenekleri

TensorFlow' da verileri yüklemenin iki yolu vardır; bu yöntemler, veri setinin büyüklüğü ve sistemin kaynakları doğrultusunda farklı avantajlar sağlamaktadır.

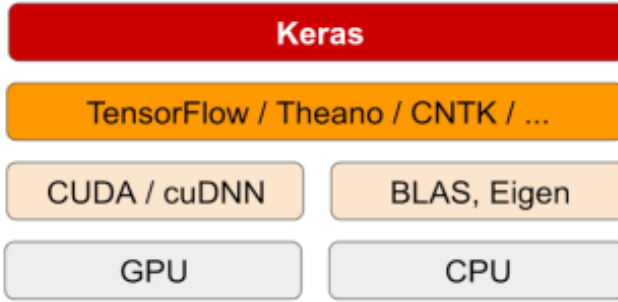
İlk yöntem, tüm veri setini belleğe yükleyerek işlemektir. Bu yöntem basit olsa da büyük veri setleri için bellek yetersizliği sorununa yol açabilir. Bu nedenle, küçük ve orta ölçekli veri setleri için daha uygundur. Yani veriler 10 Gigabayttan küçükse belleğe sığabilmekte ve bir problem oluşturmamaktadır.

İkinci yöntem ise TensorFlow'un veri hattı (Dataset API) özelliğidir. Bu yöntem, büyük veri setlerini daha küçük parçalara ayırarak işlemektedir. Böylece, bellek kullanımını optimize etmekte ve daha hızlı sonuçlar elde edilmesini sağlamaktadır. Ayrıca, veri ön işleme, artırma ve dağıtık eğitim gibi işlemler için de esneklik sunmaktadır. Büyük veri setleri, hız ve ölçeklenebilirlik gerektiren durumlarda TensorFlow veri hattı tercih edilmelidir. Veri hattı, büyük veri setlerini daha küçük parçalara (Batch) ayırarak ve bu parçaları sırayla modele besleyerek çalışmaktadır. Bu sayede, tüm veri setini aynı anda belleğe yüklemeye gerek kalmadan verimli bir eğitim süreci gerçekleştirilmektedir. Örneğin, 16 GB belleğe sahip bir bilgisayarda 50 GB'lık bir görüntü veri setini işlemek istediğinizde, veri hattı sayesinde bu veriyi küçük parçalara ayırarak belleği etkin kullanır ve model eğitimini sorunsuz bir şekilde tamamlamaktadır.

TensorFlow'un sunduğu iki veri besleme yöntemi arasında seçim yaparken, veri setinin büyüklüğü, sistem kaynakları ve uygulama ihtiyaçları göz önünde bulundurulmalıdır. Küçük veri setleri için tüm veriyi belleğe yükleme yöntemi yeterli olabilirken, büyük veri setleri ve daha karmaşık işlemler için TensorFlow veri hattı daha uygun bir seçenektir.

3. KERAS

Keras, DÖ modellerini geliştirmek ve değerlendirmek için güçlü ve kullanımı kolay ücretsiz bir açık kaynak kodlu bir Python Uygulama Geliştirme Çatısıdır (Chollet, 2018). TensorFlow, Theano ve CNTK gibi altyapılarla uyumlu olarak çalışan bir yüksek seviyeli sinir ağı API' dır. Oldukça faydalı fonksiyonlara sahip olan Keras; sinir ağı modellerini, yalnızca birkaç kod satırında tanımlamaya ve eğitmeye olanak tanımaktadır.



Şekil 2. Derin öğrenmede yazılım ve donanımın beraber çalışması (Chollet, 2018)

Keras, özellikle başlangıç seviyesindeki kodlayıcılar için kullanım kolaylığı sunarak, hızlı ve etkili bir şekilde model oluşturmayı mümkün kılmaktadır. Kullanıcılar, modeller üzerinde yapılan değişikliklerin etkilerini hızla gözlemleyebilmektedir. Ayrıca, yaygın kullanılan DÖ algoritmaları olan Tekrarlayan Sinir Ağları (RNN) ve Evrişimli Sinir Ağları (CNN) gibi yapıları desteklemektedir. Güçlü TensorFlow altyapısı üzerinde çalışan Keras, model eğitim süreçlerini hızlandırarak daha az zamanda daha etkili sonuçlar elde etme imkânı sunmaktadır. Bu özellik, özellikle büyük veri kümeleri ile çalışan araştırmacılar ve geliştiriciler için büyük avantaj sağlamaktadır. Keras ile ilgili geniş çapta dokümantasyon mevcuttur ve bu da kullanıcıların sorunlarına çözüm bulmalarını veya yeni teknikler öğrenmelerini kolaylaştırmaktadır. Keras'ın geniş kullanıcı topluluğu, çeşitli sorunlar ve projeler üzerinde fikir alışverişinde bulunulabilen bir ortam sunmaktadır. Keras, model karmaşıklıklarına takılmak yerine, doğrudan sonuçlara odaklanmaktadır. Bu yaklaşım sayesinde, görüntü sınıflandırması veya zaman serileri analizi gibi

uygulamalarda, kullanıcılar hızlı ve etkili modelleme yapabilmektedirler. Keras'ta kullanılacak üç temel model türü vardır:

- Sıralı Model (Sequential Model): Model katmanlarının sıralı bir şekilde düzenlendiği basit ve etkili bir yapıdır.
- İşlevsel Model (Functional Model): Daha karmaşık model yapılarını, örneğin çoklu giriş ve çıkış katmanlarını destekleyen esnek bir yapıdır.
- Model Alt Sınıflandırma (Model Subclassing): Karmaşık ve alışılmadık dışındaki uygulamalarda en fazla esneklik sunan yapıdır.

3.1. Keras Sıralı Model (Keras Sequential Model)

Keras Sıralı Model, basit ve kullanımı kolay bir modeldir. Adından anlaşılacağı gibi, asıl görevi Keras'ın katmanlarını sıralı olarak düzenlemektir. Özellikle doğrusal bir yığın oluşturan katmanlara sahip basit modeller tasarlamak için çok uygundur. Bu modelde veri bir katmandan diğerine akar. Veri akışı son katmana ulaşıncaya kadar devam eder. 'Sequential ()' metodu çağırılarak bir Sıralı Model oluşturulabilmektedir.

3.1.1. Keras Sıralı Model'e Katman Ekleme

Sıralı Modelde katman eklemek 'add ()' yöntemiyle herhangi bir katman kolayca gerçekleştirilebilmektedir. 'add ()' yöntemini kullanmak için önce API kullanarak bir katman oluşturulması ve ardından katmanın 'add ()' metodundan geçirilmesi gerekmektedir. Dense yapısı ise 'add ()' ile beraber kullanılarak ağ katman oluşturmaktadır.

3.1.2. Keras Sıralı Model' de Girdi Verileri

Keras'ta bir model oluştururken, ilk katmana girdi şeklinin nasıl tanımlandığını belirtmek önemlidir. İlk katman, modelin geri kalanı için girdi boyutlarını belirleyen temel yapı taşıdır. İlk katmanın çıktısı, ardışık katmanların girdisi olarak otomatik olarak kullanılmaktadır. Bu nedenle, yalnızca ilk katmanda girdi boyutunu belirtmek yeterlidir. Keras, ilk katman için girdi şeklini belirlemek amacıyla kullanılacak birkaç yöntem sunmaktadır:

- 'input_shape' bağımsız değişkeni ilk katmana girdi tensörünün şeklini belirtmek için kullanılmaktadır. Örneğin, bir CNN'in ilk

katman için girdi şekli; yükseklik, genişlik, kanal sayısı şeklinde belirtilebilmektedir.

- ‘Input_dim’ bağımsız değişkeni özellikle Yoğun Katmanlar (Dense layers) gibi tek boyutlu girdiler için kullanılmaktadır. input_dim, girdi verisinin özellik sayısını ifade etmektedir.

Her iki parametre, input_shape ve input_dim, yalnızca modelin ilk katmanında tanımlanır. İlk katmanın tanımlanmasının ardından, Keras otomatik olarak sonraki katmanlara gerekli girdi boyutlarını aktarır, böylece diğer katmanlarda bu parametrelerin ayrıca tanımlanmasına gerek kalmaz.

3.1.3. Keras Sıralı Model’ de Batch Yapısı

Sınırlı hafıza kapasitesine sahip sistemlerde, büyük veri setlerinin tamamı bir defada işlenememektedir. Bu nedenle, veriler belirli miktarda küçük gruplara, yani Batch'lere bölünmektedir. Her bir Batch, model tarafından ayrı ayrı işlenmekte ve bir dizi Batch'in tamamı işlendiğinde bir "Epoch" tamamlanmış olmaktadır. Epoch, eğitim veri setinin tamamının model tarafından bir kez işlenmesi sürecini ifade etmektedir. Bu nedenle, modelde 'Batch_size' parametresinin tanımlanması gerekmektedir. Bu parametre, modelin ne kadar hızlı öğrendiği ve genelleme yeteneği üzerinde önemli etkilere sahip olabilmektedir. Büyük Batch boyutları genellikle daha hızlı hesaplama süresi ve daha kararlı bir hata gradyanı güncellemesi sağlamaktadır; ancak, çok büyük Batch boyutları modelin genelleme yeteneğini olumsuz etkileyebilmektedir. Öte yandan, çok küçük Batch boyutları modelin eğitim süresini uzatmakta ve eğitim sürecini daha gürültülü hale getirebilmektedir.

3.1.4. Keras Sıralı Model’ de Aktivasyon Fonksiyonu

Model tanımlaması yapılırken en önemli konulardan birisi de seçilen Aktivasyon Fonksiyonudur. Bu fonksiyonlar çıkışı (0-1) veya (-1-1) arasında değerler üreten fonksiyonlardır. Dense yapısı içerisinde modelde kullanılmak istenen aktivasyon fonksiyonu oluşturulabilmektedir. Ya da ayrıca ‘model.add ()’ yapısının içerisinde tanımlama yapılabilmektedir. Aktivasyon fonksiyonu her katmanda ayrı ayrı tanımlanabilmektedir.

3.1.5. Keras Sıralı Model’ de Dropout Yapısı

Dropout Katmanı, DÖ modellerinde aşırı öğrenmeyi (overfitting) azaltmak için kullanılan bir yöntemdir. Modeldeki fazla uyum, eğitim

verilerinde daha fazla doğruluk gösterdiğinde ancak test verilerinde daha az doğruluk gösterdiğinde anlaşılmaktadır. Her bir Epoch döngüsü boyunca, katmanlarda yer alan bazı nöronlar silinmektedir, ayrıca bu nöronların giriş ve çıkış bağlantıları da tamamen silinmektedir. Ağ bu şekilde eğitildikten sonra yeni Epoch' ta silinen nöronların hepsi geri getirilerek tekrardan bir silme işlemi gerçekleştirilmektedir. Bu işlem tüm Epoch' lar boyunca devam etmektedir.

Dropout değeri ise kullanıcı tarafından belli bir değer olarak belirlenmektedir. Örneğin; 0,2 değerinde bir Dropout işleminde modeldeki nöronların yüzde 20'si silinecektir demektir. Bu silme işlemi de her seferinde tamamen random olarak gerçekleştirilmektedir.

3.1.6. Keras Sıralı Model' de Derleme

Keras'ta model oluşturmanın ve eğitimin bir parçası olarak derleme işlemi, modelin nasıl öğreneceğini ve kendisini nasıl iyileştireceğini belirleyen kritik bir adımdır. Bu işlem, compile () metodu ile gerçekleştirilir ve modelin eğitimi için gerekli yapılandırmaları tanımlamaktadır. Derleme işlemi, modeli eğitime hazırlar ve belirli parametrelerin ayarlanmasını gerektirir. Özellikle; 'Kayıp Fonksiyonu' (loss function) ve 'Optimizer' temel parametreleri oldukça önemlidir.

3.1.7. Keras Sıralı Model'de Kayıp Fonksiyonları

Kayıp fonksiyonları, DÖ modellerinde hayati bir rol oynamaktadır. Bu fonksiyonlar, modelin eğitimi sırasında hata miktarını ölçmek ve minimize etmek amacıyla kullanılmaktadır. Modelin performansını doğrudan etkileyen bu fonksiyonlar, beklenen çıktılara model çıktılarının mümkün olduğunca yaklaşmasını hedeflemektedir. Kayıp fonksiyonu, modelin tahminlerinin gerçek değerlerden ne kadar farklı olduğunu nicel olarak ifade etmektedir. Bu fark, modelin ağırlıklarını ayarlamak için kullanılan bir gradyan hesaplamasıyla elde edilmektedir. En çok bilinen kayıp fonksiyonları aşağıdaki gibidir:

1. Ortalama Kare Hatası (Mean Squared Error (MSE))
2. Ortalama Mutlak Hata (Mean Absolute Error (MAE))
3. Ortalama Mutlak Yüzde Hata (Mean Absolute Percentage Error (MAPE))
4. Squared_Hinge

5. Hinge
6. İkili Crossentropy (Binary Crossentropy)
7. Kategorik Crossentropy (Categorical Crossentropy)
8. Sparse Categorical Crossentropy
9. Kullback Leibler Divergence (KLD)
10. Poisson
11. Kosinüs Benzerliği (Cosine Proximity)

3.1.8. Keras Sıralı Model'de Optimize Ediciler

Modelin performansını iyileştirmek için model ağırlıkları iteratif olarak güncellenmektedir. Bu süreç, modelin eğitim veri setindeki hataları minimize edecek şekilde ağırlıklarını ayarlamasını sağlamaktadır. Optimizasyon işlemi, belirlenen bir 'kayıp fonksiyonu' (loss function) kullanılarak gerçekleştirilmektedir. Kayıp fonksiyonu, modelin mevcut durumunun ne kadar iyi veya kötü olduğunu ölçmekte ve Optimize ediciye modelin hangi yönde geliştirilmesi gerektiği konusunda bilgi vermektedir. Optimize ediciler, modelin ağırlıkları üzerinde değişiklik yaparak, yani bu ağırlıkları sürekli olarak güncelleyerek, modeli en iyi hale getirmeye çalışmaktadır. Bu işlem, kayıp fonksiyonunun değerini azaltarak modelin daha doğru tahminler yapmasını sağlamaktadır. Optimize edici, her bir iterasyonda model ağırlıklarını kayıp fonksiyonuna göre ayarlamakta ve böylece modelin performansını adım adım iyileştirmektedir. Yaygın olarak bilinen Optimizasyon algoritmaları; SGD (Stochastic Gradient Descent), Adagrad (Adaptive Gradient Algorithm), Adadelta, RMSprop (Root Mean Square Propagation), Adam (Adaptive Moment Estimation), Adamax, Nadam'dır.

SGD (Stochastic Gradient Descent): SGD veya Stokastik Gradyan İnişi; klasik bir Optimizasyon algoritmasıdır (Robbins & Monro, 1951). SGD'de, ağırlık her bir ağırlığı için kayıp fonksiyonunun gradyanı hesaplanmaktadır. Ağırlık her bir ileri yayılımı, belirli parametrelerle tanımlanan bir kayıp fonksiyonu ile sonuçlanmaktadır. Ağırlıkları, eğiminin gösterdiği yöne hareket ettirmek için, ağırlıkların her biri için belirli bir öğrenme oranıyla çarpılarak oluşturulan degradelerin her biri kullanılmaktadır. SGD, hem kavramsal yapısı hem de işleyişi açısından en sade algoritmalardan biridir. Her iterasyonda güncellenen yeni ağırlıklar, önceki iterasyondaki ağırlıklara göre kesin bir iyileşme sağlamaktadır. 3 adımda gerçekleştirilmektedir.

1. Kayıp fonksiyonunu etkileyen her bir ağırlık parametresindeki tüm küçük değişiklikler hesaplanmaktadır.
2. Her bir ağırlık gradyanına göre ayarlanmaktadır.
3. Kayıp fonksiyonu minimuma ulaşana kadar 1 ve 2. adım tekrarlanmaktadır.

Ancak, gradyan; kısmi bir türev ve değişimin ölçüsü olduğundan, bu algoritmada bazı zorluklar vardır. Sorun, büyük çok boyutlu veri kümeleriyle uğraşıldığında, yerel minimumda takılı kaldığında ortaya çıkmaktadır.

Adagrad (Adaptive Gradient Algorithm): SGD'nin bir varyasyonu olup, her bir parametre için öğrenme oranını adaptif olarak ayarlayarak çalışan gelişmiş bir MÖ optimizasyon algoritmasıdır (Duchi JDUCHI & Singer, 2011). Adagrad, büyük ölçekli veri setlerinde ve özellikle yüksek boyutlu veri uzaylarında etkilidir. Her bir parametrenin ne kadar sık güncellendiğine bağlı olarak öğrenme oranını otomatik olarak ayarlayarak, sık güncellenen özellikler için daha düşük, nadiren güncellenen özellikler için daha yüksek öğrenme oranları uygulamaktadır.

Adadelta: Adadelta, öğrenme hızının monoton olarak azalması sorununu çözmek amacıyla, momentum tekniklerini kullanan Adagrad'ın bir varyasyonudur (Zeiler, 2012). Bu yöntem, her ağırlığın gradyan güncellemesini mevcut gradyanların ağırlıklı toplamıyla ve geçmişte yapılan sınırlı sayıda gradyan güncellemesinin üstel olarak azalan ortalamasıyla bütünleştirmektedir. Bu sayede gradyan paydası monoton bir şekilde artmadığı için öğrenme oranı daha stabil hale gelmekte ve algoritma genel olarak daha sağlam olmaktadır.

RMSprop (Root Mean Square Propagation): RMSprop, özellikle Adagrad'ın uzun süreli eğitimlerde karşılaştığı monoton öğrenme oranı azalması probleminde bir çözüm olarak geliştirilmiştir. Bu algoritma, spesifik olarak bir makalede yayınlanmamış olmakla birlikte, araştırmacının çevrimiçi bir dersi üzerinde tanıtılmıştır. RMSprop; öğrenme sürecini hızlandırmak ve daha stabil hale getirmek amacıyla, her parametrenin öğrenme oranını adaptif olarak ayarlamaktadır.

Adam (Adaptive Moment Estimation): RMSprop ve Adadelta'nın sağladığı özellikleri temel alarak, hem geçmiş gradyanların üstel olarak azalan

ortalamasını hem de bu gradyanların karelerinin ortalamasını kullanmaktadır (Kingma & Lei Ba, 2015). Bu yaklaşım hem momentumun hem de adaptif öğrenme oranlarının avantajlarını birleştirerek, onu birçok durumda diğer optimizasyon algoritmalarına göre daha hızlı ve etkili kılabilmektedir.

Adamax : Adından da anlaşılacağı gibi AdaMax, Adam algoritmasını yazan aynı araştırmacılar tarafından Adam optimizer'ın bir uyarlamasıdır (Kingma & Lei Ba, 2015). Adam, gradyanların karelerinin üstel olarak azalan ortalamasını kullanırken, Adamax, gradyanların maksimumlarını kullanarak ölçeklendirme yapmaktadır. Bu özellik, Adamax'ı, algoritmanın uzun vadeli bağımlılıkları daha iyi idare etmesini sağlayarak öğrenme oranının stabilitesini artırmaktadır.

Nadam : Nesterov Hızlandırılmış Gradyan (NAG) ve Adam optimizasyon algoritmalarının birleşiminden oluşmaktadır (Dozat, 2016). Ancak sıradan momentum yerine Nesterov momentumuna sahiptir. Nesterov momentumu, normal momentumdan farklı olarak, mevcut gradyan yerine bir sonraki adımda beklenen konumu hesaplamak için kullanılmaktadır. Bu, genellikle öğrenme sürecinde daha az dalgalanmalara yol açmaktadır. Nadam, özellikle hızlı ve etkili bir şekilde minimum kayıp değerine ulaşmak isteyen uygulamalarda tercih edilmektedir.

3.1.9. Keras Sıralı Model' in Eğitimi ve Tahmini

Modelin eğitimi, fit() metodu ile gerçekleştirilmektedir. Bu metod, girdi (input) ve hedef (target) verileri, Epoch sayısı, Batch Size gibi parametreleri alarak çalışmaktadır. Eğitim sürecinde, model veri üzerinde iteratif olarak çalışmakta, ağırlıklarını ayarlamakta ve kayıp fonksiyonunu minimize etmeye çalışmaktadır.

Eğitimden sonra, predict() metodu, modelin yeni veriler üzerinde tahminler yapmasını sağlamaktadır. Bu metod, modelin eğitildiği veri türüne benzer şekilde hazırlanmış girdileri almakta ve çıktı olarak tahminleri döndürmektedir.

3.1.10. Keras Sıralı Model'de Metrikler

Modelin derlenmesi aşamasında kullanılan en son parametre 'metrics' parametresidir. Eğitim esnasında performans değerlendirmesine yardımcı olan yapılardır. Keras'ta farklı metrik türleri vardır:

- Accuracy; modelin tüm tahminleri arasındaki doğru tahminlerin oranını hesaplamaktadır. Çok sınıflı sınıflandırma problemlerinde genel başarıyı ölçmek için kullanılmaktadır.
- Binary_Accuracy; ikili sınıflandırma problemleri için (çıkış değerinin iki değer döndürdüğü yapılarda) tüm tahminlerde ortalama doğruluk oranını hesaplamaktadır.
- Categorical_Accuracy; çıkış değeri ikiden fazla olan modellerde doğruluğu tespit etmek için kullanılmaktadır.
- Sparse_Categorical_Accuracy; tıpkı Categorical_Accuracy gibi çıkış değeri ikiden fazla olan modellerde doğruluğu tespit etmek için kullanılmaktadır. Ancak biraz daha az çıkışları olan modellerde kullanılmaktadır.

3.2. Keras Fonksiyonel Model

Keras Fonksiyonel Model; daha karmaşık ve esnek model yapılarının oluşturulabilmesine olanak tanıyan bir yapıdır. Sıralı modelin aksine, birden fazla giriş ve çıktıya sahip modellerin yanı sıra, katmanlar arasında karmaşık bağlantıların kurulmasını mümkün kılmaktadır. Bu modeli kullanarak görevleri gerçekleştirmek için öncelikle modelin bir örneğinin oluşturulması gerekmektedir. Ayrıca, modelin girişine ve çıkışına erişmek için katmanların bu modelle bağlanması gerekmektedir. Keras Fonksiyonel Modelde tanımlanan katmanlar, başka bir model içerisinde tekrar kullanılabilir. Bu, model parçalarının modüler bir şekilde yeniden kullanılmasını sağlamak ve geliştirme sürecini hızlandırmaktadır. Model tensör üzerinde çağırılarak, katman olarak kullanılabilir. Ayrıca dallanmalar, birleşmeler ve tekrar eden katman yapıları gibi karmaşık ağ topolojilerinin oluşturulmasına izin vermektedir. Bu, modelin daha özelleşmiş ve belirli görevlere daha uygun hale getirilmesine yardımcı olmaktadır. Özellikleri aşağıda sıralandığı gibidir.

- Modele başlamak için, 'Input' fonksiyonu ile bir veya daha fazla giriş katmanı oluşturulmaktadır. Giriş katmanı, veri setinin özelliklerine göre şekillendirilmektedir.

- Sıralı modelden farklı olarak, önce giriş verilerinin şeklini belirten bağımsız bir giriş katmanının oluşturulması ve tanımlaması gerekmektedir.
- Yeni bir model oluşturduktan sonra bir girdi katmanının oluşturulması gerekmektedir.
- ‘Dense’ modeli tanımlanmaktadır. Modeldeki katmanlar, her yeni katman tanımlanırken girdinin nereden geldiği belirtilerek ikili olarak bağlanmaktadır. Giriş katmanını belirten bir parantez gösterimi kullanılmaktadır.
- Dense modeli tanımlandıktan sonra, giriş için bir ‘Dense Katmanı’ oluşturulması gerekmektedir.
- Fonksiyonel model, katmanlardan bir model oluşturmak için ‘model ()’ metodunu kullanmaktadır. Giriş ve çıkış katmanlarının belirtilmesi gerekmektedir.

3.3. Keras Model Alt Sınıflandırma

Keras Model Alt Sınıflandırması, yüksek düzeyde kişiselleştirilmiş modeller oluşturmak için esnek bir çözüm sunmaktadır. Özellikle mimariyi hassas bir şekilde kontrol etmek mecburiyetinde olan karmaşık modeller için veya standart dışı fonksiyonlar eklemek istenildiğinde tercih edilmektedir. Bu yöntem, katmanlara ve bu katmanlar aracılığıyla akışını sağladığınız verilere doğrudan erişim imkânı tanımakta, bu da özellikle hata ayıklama veya daha karmaşık mekanizmalar geliştirme sürecinde büyük bir avantaj sağlamaktadır. Ancak, bu modelin dezavantajları arasında artan karmaşıklık ve modellerin çalışma prensiplerine ilişkin daha ayrıntılı bir bilgi gereksinimi bulunmaktadır. Ayrıca, özel yürütme yolu nedeniyle olası hataları saptamak zorlaşmaktadır. Keras'ta bir modeli alt sınıfa ayırmak için izlenmesi gereken adımlar aşağıdaki gibidir:

- Keras'ta alt sınıf oluşturma, ‘tf.keras.Model’ den miras alan yeni bir sınıf tanımlamakla başlamaktadır. Bu sınıf içinde, ‘__init__ ()’ metodunda katmanlar tanımlanabilmekte ve mimari oluşturulabilmektedir.
- ‘call’ metodu ile ‘__init__ ()’ metodunda tanımlanan katmanlar kullanılabilen veya özel bir kullanım gerçekleştirilebilmektedir.

- Sınıf tanımlandıktan sonra, modelin örneklerini oluşturulabilmektedir. Tıpkı diğer Keras modellerinde olduğu gibi, model, optimize edici, kayıp fonksiyonu ve metriklerle derlenebilmektedir.

4. THEANO

Theano, çok boyutlu dizileri içeren matematiksel ifadeleri tanımlamak, optimize etmek ve değerlendirmek için tasarlanmış bir Python Uygulama Geliştirme Çatısıdır (The Theano Development Team et al., 2016) . Özellikle ML topluluğunda en yaygın kullanılan CPU ve GPU matematik derleyicilerinden biri haline gelmiş ve sürekli olarak performansını artırmıştır. Theano, 2008'den beri aktif olarak geliştirilmekte olup, üzerine birden fazla Uygulama Geliştirme Çatısı inşa edilmiş ve birçok ML modeli geliştirmek için kullanılmıştır. Ayrıca akıllı kod optimizasyonları ile DÖ için büyük sinir ağlarındaki hesaplamalarda kullanılmaktadır. Theano'nun özellikleri genel olarak aşağıdaki gibidir:

- Matematiksel sembolik ifade derleyicisi- Tanınması kolay ifadelerden oluşmuştur. İfadeleri Numpy kütüphanesinin semantik ve sözdizimine benzemektedir.
- Otomatik farklılaşma- Theano otomatik olarak çeşitli noktalarda gradyanların nasıl hesaplanacağını anlayarak model eğitimi için gradyan inişi gerçekleştirme olanağı sağlamaktadır.
- GPU' nun şeffaf kullanımı- Aynı kodu hem CPU hem de GPU'da çalıştırabilme yeteneğine sahiptir. Özellikle, hesaplamaların hangi bölümlerinin GPU'ya taşınması gerektiğini otomatik olarak anlayarak performans iyileştirmeleri sağlamaktadır.
- Hız ve kararlılık optimizasyonları- Theano, hesaplamaların daha hızlı ve sayısal olarak daha kararlı çalışabilmesi için, dahili olarak yeniden düzenlemeler yapıp optimize edebilmektedir. Kararsız ifadeleri optimize etmek ve değerlendirmek için kararlı araçlar sağlamaktadır.
- Keras ile Tam Uyumluluk- Keras ile tamamen uyumlu şekilde çalışmaktadır. Bu da kullanıcıların daha yüksek seviyeli model oluşturma araçlarını Theano'nun performans avantajları ile birleştirmesine olanak tanımaktadır.
- Arka uçta tanımlı C kodları olan bir lineer cebir derleyicisidir.

Tensorflow ile karşılaştırılacak olursa; görevleri Tensorflow' dan daha hızlı çalışmaktadır. Özellikle tekli GPU görevlerinde Theano2' de daha hızlı çalışmaktadır. Ancak çoklu GPU görevlerinde Tensorflow daha hızlı çalışmaktadır.

5. CAFFE

Caffe, ifade, hız ve modülerlik göz önünde bulundurularak yapılmış bir DÖ Uygulama Geliştirme Çatısıdır (Jia et al., 2014). Görüntü sınıflandırma ve görüntü segmentasyonu için çeşitli DÖ mimarilerini desteklemektedir. Caffe' nin özellikleri aşağıdaki gibi sıralanabilmektedir:

- Etkileyici mimari- ifadeyi kolay ve anlaşılır hale getiren metin tabanlı bir yapı sunmaktadır. Bu yapı, kullanıcıların kod yazmadan modelleri tanımlamasına olanak tanımaktadır.
- Optimizasyon- Kodlama yapılmadan Uygulama Geliştirme Çatısı üzerinden gerçekleştirilebilmektedir. Modelleri tanımlaması kolaydır ve kodlama çabalarına gerek kalmadan optimize edilebilmektedir.
- GPU' nun şeffaf kullanımı- Kullanıcı, tek bir etiket kullanarak GPU ve CPU arasında geçiş yapabilmekte ve DÖ modelini bir GPU makinesinde eğitebilmektedir.
- Hız- Caffe'nin tercih edilmesinin önemli nedenlerinden biri hızıdır. Son sürümler ve geliştirmelerle beraber çalışma hızı daha da iyileştirilmeye devam etmektedir. Bu, büyük veri setleri ve karmaşık modellerle çalışırken önemli bir avantaj sağlamaktadır.

Caffe' nin, DÖ mimarilerindeki hesaplama karmaşıklığı arttıkça, çeşitli kısıtlılıkları gözlemlenmiştir. Bu yüzden Caffe2, sunulmuştur. Caffe2; Facebook AI Research (FAIR) tarafından desteklenmektedir. Daha kolay uygulanması ve bulut yapısı üzerindeki GPU' lara olan erişimiyle çeşitli yapılarıdaki kitaplıklara erişebilmektedir. Yani Caffe2'nin çıkış iddiası, daha ölçeklenebilir ve hafif olduğudur. Ayrıca Caffe2 mobil uygulama geliştirmesini desteklemektedir.

6. PYTORCH

PyTorch, Facebook tarafından geliştirilen ve sürdürülen önde gelen açık kaynaklı bir yapıdır (Paszke et al., 2019). Tensör hesaplamaları, otomatik farklılaşma ve GPU hızlandırma konularında uzmanlaşmış bu yapısı, popülerliği nedeniyle Keras ve TensorFlow ile rekabet eden en çok kullanılan

Uygulama Geliştirme Çatılarından biridir. PyTorch'un özellikleri aşağıda sıralanmıştır.

- PyTorch, tensör hesaplamalarını güçlü GPU hızlandırmasıyla gerçekleştirmektedir. Tensörler, NumPy'nin sunduğuna benzer bir şekilde işlem yapılmasına olanak tanımakta, ancak ek olarak GPU'lar üzerinde paralel hesaplama imkânı sağlamaktadır. Bu özellik, büyük veri kümeleri ve karmaşık hesaplamalar için önemli performans avantajları sunmaktadır.
- PyTorch, DÖ modelleri tasarlayıp eğitmek amacıyla otomatik farklılaştırma özelliğini kullanmaktadır. Bu, geri yayılım sürecini otomatik olarak gerçekleştirerek ağ parametrelerinin yeniden güncellenmesini sağlamaktadır. Kullanıcıların manuel olarak türev hesaplamasına gerek kalmadan, PyTorch bu süreci etkin bir şekilde yönetmektedir.
- PyTorch, yüksek esneklik ve hız sağlayan bir DÖ araştırma platformu olarak öne çıkmaktadır. Model oluşturma ve eğitim süreçlerinde kullanıcı dostu bir deneyim sunmaktadır.
- PyTorch, geniş bir geliştirici ve araştırmacı topluluğuna sahiptir. Sürekli güncellenen belgeler, eğitim materyalleri ve aktif topluluk desteği ile kullanıcılar, karşılaştıkları sorunları hızlıca çözebilmekte ve en iyi uygulamaları öğrenebilmektedirler.

7. CNTK (MICROSOFT BİLİŞSEL ARAÇ TAKIMI)

CNTK (Microsoft Bilişsel Araç Takımı), Microsoft tarafından geliştirilmiş bir DÖ Uygulama Geliştirme Çatısı'dır (The Microsoft Cognitive Toolkit - Cognitive Toolkit - CNTK | Microsoft Learn, 2017). İleri beslemeli Derin Sinir Ağlarının, CNN'ler ve Tekrarlayan Sinir Ağları (RNN'ler/LSTM'ler) gibi yaygın model türlerini kolaylıkla oluşturup entegre etmesine olanak tanımaktadır.

- CNTK, gradyan hesaplama ve paralelleştirme özellikleri ile öne çıkmaktadır. Bu özellikler, modelin eğitim sürecinde SGD ve hata geri yayılımı gibi öğrenme algoritmalarının verimli bir şekilde uygulanmasını sağlamaktadır. CNTK, birden çok GPU ve sunucu arasında paralel hesaplama yaparak eğitim sürecini hızlandırmaktadır.

- CNTK, çok çeşitli uygulama alanlarında kullanılabilir (görüntü işleme, ses tanıma, doğal dil işleme ve zaman serisi analizi). Özellikle büyük veri setleri ve karmaşık model yapılarına sahip projelerde CNTK'nin sağladığı performans ve esneklik önemli avantajlar sunmaktadır.

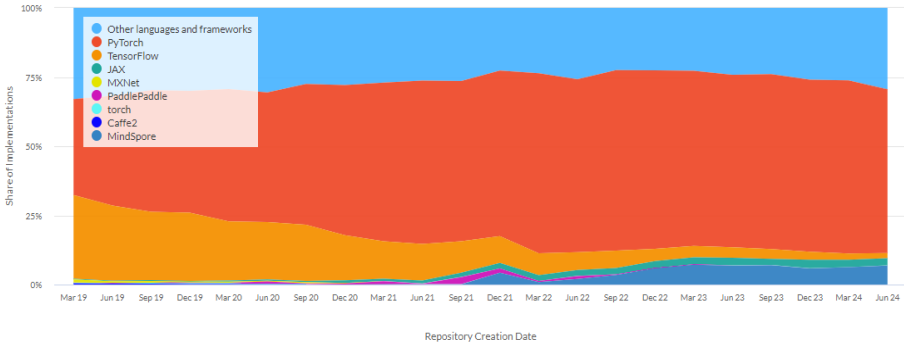
8. DERİN ÖĞRENME ALANINDA KULLANILAN UYGULAMA GELİŞTİRME ÇATILARININ KARŞILAŞTIRILMASI

DÖ alanında kullanılan Uygulama Geliştirme Çatılarının her birinin ayrı ayrı avantajları ve sınırlamaları vardır. Eğitilecek olan ağ için doğru olanı seçmek için bu Uygulama Geliştirme Çatıları hakkında bilgi birikimine sahip olmak önemlidir. Bu yapıların ara yüzleri, farklı modellerle entegre kullanılabilirlikleri, çalışma hızları, GPU destekleyip desteklemedikleri, paralel programlama sağlayıp sağlamamaları önemli parametrelerdir. Tablo 1’de DÖ Uygulama Geliştirme Çatılarının karşılaştırılması sunulmuştur.

Tablo 1. Derin Öğrenme Kütüphanelerinin karşılaştırılması

Kütüphane	Yıl	Yazıldığı dil	Geliştiricisi	Popülerite-Büyüme	Öne çıkan özellikleri
Caffe	2014	Python	Berkeley Vision and Learning Center	- Orta-Düşük popülerite - Hızlı büyüme	- Demo uygulamaları oluşturmak için önceden eğitilmiş modeller sunmaktadır. - PyTorch gibi diğer kütüphanelerle iyi çalışmaktadır.
TensorFlow	2014	Python	Google Brain Team	- Çok yüksek popülerite - Çok hızlı büyüme	- Tek bir API ile bir masaüstü, sunucu veya mobil cihazdaki bir veya daha fazla CPU’ya veya GPU’ya dağıtma olanağı sunmaktadır. (Şeker et al., 2017)

Theano	2016	Python	Theano Development Team	-Orta-Düşük popülarite -Düşük büyüme	- Keras, Blocks gibi API sayesinde matematiksel hesapları kolaylaştırmaktadır (Şeker et al., 2017). -GPU desteği
Microsoft Cognitive Toolkit (CNTK)	2017	C++	Microsoft	-Orta popülarite -Hızlı büyüme	-İyi performans ve ölçeklenebilirlik sağlamaktadır.
Keras	2018	Python	Google (François Chollet)	-Yüksek popülarite -Çok hızlı büyüme	- Basit ve sezgisel bir ara yüze sahiptir, yeni başlayanlar için uygun bir seçenektir. -Sinir ağlarında hızlı denemeler yapmak için tasarlanmıştır. -GPU desteği vardır.
Pytorch	2019	Python ve C++	Facebook (FAIR)	-Çok yüksek popülarite -Çok hızlı büyüme	-Mimari tarzı sayesinde modelleme süreci basit ve şeffaftır. -Önceden eğitilmiş birçok modele ve birleştirilmesi hazır ve kolay modüler parçalara sahiptir.



Şekil 3. DÖ Uygulama Geliştirme Çatılarının popülerliğinin incelenmesi³

Şekil 3’ te DÖ alanında kullanılan Uygulama Geliştirme Çatılarının; GitHub Repository’ nin içeriği incelenerek elde edilmiş bir grafik çalışması yer almaktadır. Son beş yıldan günümüze kadar gerçekleştirilmiş olan yayınlar analiz edilerek elde edilmiş olan bir grafikdir. Grafığe göre özellikle PyTorch’un oldukça yaygın olarak kullanıldığı görülmektedir.

9. DERİN ÖĞRENME ALANINDA KULLANILAN UYGULAMA GELİŞTİRME ÇATILARININ DESTEKLEDİKLERİ DONANIM BİRİMLERİ

DÖ uygulamaları oldukça karmaşık olup, eğitim süreçleri büyük miktarda hesaplama gerektirmektedir. Bu süreçler, büyük veri setlerinin işlenmesi, yinelemeli işlemler, matematiksel hesaplamalar ve matris çarpımları gibi yoğun hesaplama gerektiren işlemleri içermektedir. Bu işlemler, normal bir CPU’da gerçekleştirildiğinde çok uzun sürmektedir. Verilerin verimli bir şekilde işlenmesi için GPU en ideal tercihtir. Hesaplama boyutu arttıkça, GPU’nun CPU’ya kıyasla sağladığı avantajlar da o ölçüde artmaktadır.

GPU’nun en büyük avantajı, paralel bilgi işlem yeteneğidir. GPU, SIMD (Single Instruction Stream Multiple Data Stream) mimarisine sahip bir donanım aygıtıdır. SIMD mimarisinde, aynı anda birden fazla veri parçası üzerinde tek işlem gerçekleştirilmektedir. Bu özellik, sinir ağlarını eğitmek için yapılan işlemlerin CPU’dan daha hızlı gerçekleştirilmesini sağlamaktadır. GPU’lar, aritmetik mantık birimlerine (ALU) daha fazla transistör ayırarak,

³ <https://paperswithcode.com/trends>

önbellek ve akış denetimine daha az kaynak ayırmaktadır. Bu durum, büyük hesaplamaların verimli bir şekilde yapılmasını sağlamaktadır. Bu mimari, büyük veri kümeleri üzerinde yapılan hesaplamaların daha hızlı gerçekleştirilmesine olanak tanımaktadır.

Ayrıca GPU'lar, geniş bellek kapasitesi ve yüksek veri aktarım hızı sağlamaktadır. Bu, geniş veri kümelerinin hızlıca işlenmesini mümkün kılmaktadır. Veri kümesi büyüdükçe, GPU'ların kullanımı o kadar avantajlı hale gelmektedir. GPU kullanımının DÖ modellerinde sağladığı faydalar aşağıdaki gibidir:

- **Hızlı Eğitim Süreleri:** GPU'lar, sinir ağlarının eğitim sürecini hızlandırarak CPU'lara kıyasla daha kısa sürede model eğitimi sağlamaktadır.
- **Verimli Büyük Veri İşleme:** Büyük veri kümeleri, GPU'lar ile daha verimli işlenmekte, bu da daha büyük ve daha karmaşık modellerin eğitilmesine olanak tanımaktadır.
- **Daha Yüksek Performans:** GPU'lar, DÖ modellerinde yüksek performans sağlamak ve bu da daha doğru ve hızlı sonuçlar elde edilmesine yardımcı olmaktadır.

Son zamanlarda, araştırma çalışmaları eğitim sürecini hızlandırmaya odaklanmıştır. Tensor İşleme Birimi (TPU) gibi özel grafik işlemcileri, DÖ uygulamaları için optimize edilmiş olup, GPU'lardan bile daha hızlı performans sunabilmektedir. TPU'lar, Google tarafından geliştirilmiş olup, özellikle tensör işlemleri için tasarlanmıştır ve DÖ modellerinin eğitim süresini büyük ölçüde azaltmaktadır.

Tablo 2. DÖ Uygulama Geliştirme Çatılarının destekledikleri donanım birimleri

	Tensorflow	Keras	PyTorch	CNTK	Caffe2
CPU	Evet	Evet	Evet	Evet	Evet
GPU	Evet	Evet (Arka planda TensorFlow veya Theano ile)	Evet	Evet	Evet
TPU	Evet	Evet (TensorFlow üzerinden)	Hayır	Hayır	Hayır
Paralel İşleme	Evet	Evet	Evet	Evet	Evet

9.1. Derin Öğrenme için Paralel İşleme

GPU kullanılarak DÖ modellerinin eğitimi önemli ölçüde hızlandırılabilir. GPU'ların paralel hesaplama yetenekleri, büyük veri setleriyle ve karmaşık modellerle çalışırken ciddi performans avantajları sağlamaktadır. Paralel işleme, DÖ eğitim süreçleri Model paralelleştirme (model parallelism) ve veri paralelleştirme (data parallelism) olmak üzere iki temel şekilde gerçekleştirilebilir.

9.1.1. Modeller Arası Paralellik

Modeller arası paralellik (Model Parallelism), DÖ eğitim sürecinde sinir ağının farklı bölümleriyle ilişkili hesaplamaların farklı işlemciler (örneğin, GPU'lar) dağıtılmasını ifade etmektedir. Bu yaklaşım, modelin büyük ve karmaşık olduğu, yani tek bir işlemcinin belleğine sığmadığı durumlarda kullanılmaktadır. Model paralelligi, özellikle büyük modellerin eğitimi sırasında bellek ve hesaplama kaynaklarının verimli kullanımını sağlamaktadır.

Model paralelliginde, sinir ağının farklı katmanları veya bölümleri farklı işlemciler atanmakta ve eğitimin ileri geçiş aşamasında her işlemci, kendi payına düşen hesaplamalardan sorumlu olmaktadır. Geri yayılım aşamasında, ağırlıkların gradyanları hesaplanmakta ve her işlemci, kendi bölgesindeki ağırlıkları güncellemektedir. Bu süreçte, gradyan bilgileri işlemciler arasında iletilmekte, böylece tüm modelin ağırlıkları senkronize edilmektedir.

Model paralelligi, literatürde nadiren incelenmekte olup, modelin çoklu işlemcilerde dağılımı zorlu bir problem teşkil etmektedir.

9.1.2. Veri Paralelligi

Veri paralelligi, eğitim verilerinin alt bölümlere ayrılmasını ve bu bölümlerin farklı makinelerde paralel olarak işlenmesini ifade etmektedir. Bu yöntem, büyük veri setleri üzerinde DÖ modellerinin eğitim süresini önemli ölçüde azaltmaktadır. Her makine, verilerin farklı bir bölümünü işlemekte ve bu süreçlerin çıktılarını birleştirilmektedir. Veri paralelligi, özellikle geniş veri setleriyle çalışıldığında verimlilik sağlamaktadır. Eğitim veri seti, her bir makinenin veya GPU'nun işleyebileceği alt kümelere bölünmektedir. Bu alt kümeler, veri paralelligi yönteminin temelini oluşturmaktadır. Her makine veya GPU, kendisine atanmış olan veri alt kümesi üzerinde ileri ve geri yayılım hesaplamalarını gerçekleştirmektedir. Her birim, kendi verisi üzerinde modeli eğitmekte ve gradyanları hesaplamaktadır. Her makineden elde edilen

gradyanlar, merkezi bir parametre sunucusuna (parameter server) gönderilmekte; bu sunucu, gradyanları birleştirerek model ağırlıklarını güncellemektedir. Güncellenmiş model ağırlıkları, tüm makinelere geri gönderilmekte ve bir sonraki iterasyon için kullanılmaktadır.

10. DERİN ÖĞRENME ÇATILARININ PERFORMANS KARŞILAŞTIRMASI: DENEYSEL BİR UYGULAMA

10.1. Materyal

Bu çalışmada Kaggle platformunda bulunan Jellyfish veri seti kullanılmıştır (*Jellyfish Image Dataset*, n.d.). Veri seti; altı farklı türe ait 900 denizanası görüntüsü içermektedir. Bu türler; Moon jellyfish (*Aurelia aurita*), Barrel jellyfish (*Rhizostoma pulmo*), Blue jellyfish (*Cyanea lamarckii*), Compass jellyfish (*Chrysaora hysoscella*), Lion's mane jellyfish (*Cyanea capillata*), Mauve stinger (*Pelagia noctiluca*) şeklindedir. Veri setinde her bir türe ait 150 adet 224x224 boyutlu renkli görüntü yer almaktadır.



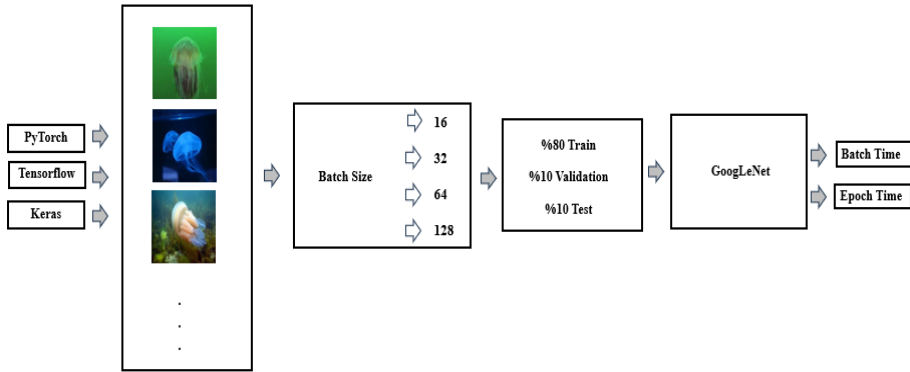
Şekil 4. Jellyfish veri setinde altı farklı denizanası türüne ait örnek görüntüler

10.2. Metot

Bu çalışmada; Jellyfish veri seti üzerinde Pytorch, Tensorflow ve Keras (Tensorflow Backend) Uygulama Geliştirme Çatılarının performans değerlendirilmesi yapılmıştır. Sınıflandırma; bir CNN modeli olan GoogLeNet ile gerçekleştirilmiştir. Veri seti; farklı Batch Size değerleri ile %80 train, %10 validation ve %10 test seti olarak ayrılmıştır. Sınıflandırma sonucunda farklı Batch Size değerlerinde elde edilen Batch Time ve Epoch Time değerleri

kaydedilerek analizler gerçekleştirilmiştir. Önerilen yöntemin akış şeması Şekil 5'te gösterilmektedir.

GoogLeNet, InceptionV1 olarak da bilinen; Google tarafından geliştirilen derin bir konvolüsyonel sinir ağıdır (Szegedy et al., 2014). GoogLeNet'in ayırt edici özelliği Inception modülüdür. Bu modül, tümü paralel olarak gerçekleştirilen farklı boyutlarda (1x1, 3x3 ve 5x5) birden fazla konvolüsyon ve 3x3 maksimum havuzlama işlemi içermektedir. Bu işlemlerden elde edilen çıktılar modülün nihai çıktısını oluşturmak üzere birleştirilmektedir. Bu yaklaşım, ağır farklı ölçeklerde ve ayrıntı düzeylerinde bilgi yakalamasına olanak tanıyarak karmaşık temsilleri öğrenme becerisini arttırmaktadır. GoogLeNet; bu çalışmada ortalama 5,2M gibi az bir parametre sayısı ile etkili bir sınıflandırma gerçekleştirmiştir. Inception modülünün birden fazla ölçekte ve ayrıntı düzeyinde bilgi yakalama yeteneği, çeşitli görevlerde sağlamlığına ve etkinliğine katkıda bulunmaktadır.



Şekil 5. Önerilen yöntemin akış şeması

10.3. Deneysel Uygulama ve Sonuçlar

Bu çalışmada, literatürde en sık kullanılan DÖ Uygulama Geliştirme Çatılarının performansları karşılaştırılmıştır. Araştırmacılara rehberlik etmek amacıyla, önce literatürdeki çeşitli DÖ Uygulama Geliştirme Çatıları karşılaştırılarak analiz edilmiştir. Daha sonra literatür analizi sonuçlarına göre en iyi DÖ Uygulama Geliştirme Çatılarının performans karşılaştırmaları yapılarak deneysel sonuçlar paylaşılmaktadır. Literatür sonuçlarına göre performans karşılaştırması için TensorFlow, Keras ve PyTorch Uygulama Geliştirme Çatıları seçilmiştir. Seçilen Uygulama Geliştirme Çatıları arasındaki

performans karşılaştırması, farklı Batch Size oranlarında; Batch Time, Epoch Time oranlarına ve Doğruluk (Accuracy) sonuçlarına göre gerçekleştirilmiştir. Performans karşılaştırmaları GPU ile gerçekleştirilmiştir. Tüm deneysel sonuçlar 32 GB belleğe sahip NVIDIA GeForce RTX 4060 32GB grafik kartı üzerinde elde edilmiştir. Çalışma CUDA 12.4 versiyonu ve Python 3.12 kullanılarak gerçekleştirilmiştir. Literatürde elde edilen bazı farklı sonuçların CUDA ve Python sürümlerindeki farklılıklardan kaynaklandığı düşünülmektedir. Bu nedenle yazılımın versiyonlarının belirtilmesi uygun görülmüştür.

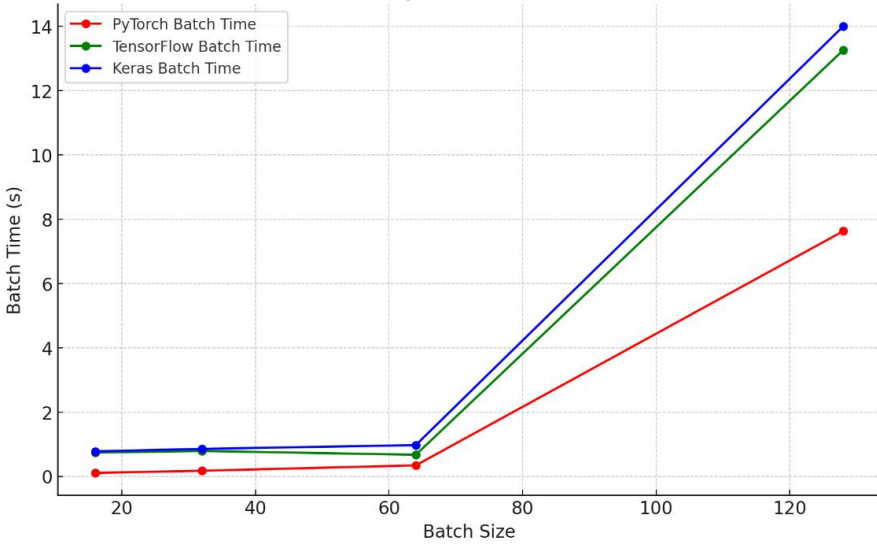
10.4. Sonuçların Değerlendirilmesi

Karşılaştırma sonuçları, Jellyfish veri seti üzerinde gerçekleştirilmiş olup, GPU üzerinde çalışan modellerin Batch Time ve Epoch Time performansları açısından en verimli Uygulama Geliştirme Çatısının Pytorch olduğu belirlenmiştir. Pytorch; Batch Size değeri 128, 64, 32 ve 16 olmak üzere tüm Batch size büyüklüklerinde en hızlı Uygulama Geliştirme Çatısı olmuştur. Uygulama sonuçları Tablo 3'de sunulmuştur. Optimizasyon işlemi için AdamW algoritması tercih edilmiştir. Learning Rate başlangıçta 0.001 olarak belirlenmiştir. Ancak, modelin daha stabil ve verimli bir öğrenme süreci geçirmesi için öğrenme oranının kademeli olarak azaltılması düşünülmüştür. Bu amaçla, step_size 3 ve gamma 0,8 olarak belirlenmiştir. Bu parametreler, her 3 Epoch'ta bir Learning Rate'in, belirli bir katsayı olan Gamma ile çarpılarak azaltılmasını sağlamaktadır. Modelin toplam parametre sayısı 23.520.326 iken eğitim parametre sayısı; 23.270.295'dir. Tablo 3'te Batch size = 16, 32, 64 ve 128 için, Batch Time (s) ve Epoch Time(s) değerleri ile performans karşılaştırması yapılmıştır.

Tablo 3. Jellyfish veri setinde kullanılan Pytorch, Tensorflow ve Keras (TensorFlow Backend) uygulama geliştirme çatılarının, farklı Batch Size (16, 32, 64 ve 128) için performans karşılaştırması

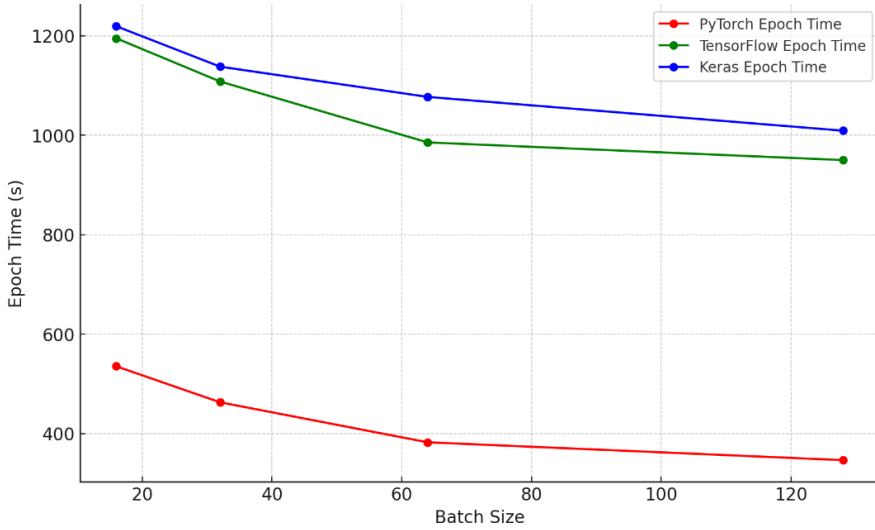
DÖ Uygulama Geliştirme Çatıları	Batch Size	Batch Time(s)	Epoch Time(s)
Pytorch	16	0.1121	535.1610
	32	0.1784	462.7370
	64	0.3452	382.2724
	128	7.6322	346.4381
Tensorflow	16	0.7414	1194.8128
	32	0.7930	1107.8940
	64	0.6756	985.3870
	128	13.2631	949.8553
Keras (TensorFlow Backend)	16	0.7825	1219.0645
	32	0.8589	1137.7416
	64	0.9759	1076.9452
	128	14.005	1009.006

Şekil 6’da Jellyfish verisetinde Pytorch, Tensorflow ve Keras Uygulama Geliştirme Çatılarının farklı Batch Size değerlerinde Batch Time analizi gerçekleştirilmiştir. Şekil 6’ya göre en verimli Uygulama Geliştirme Çatısının Pytorch olduğu görülmektedir. Pytorch’un küçük Batch size (16, 32, 64) değerlerinde Batch Time değeri oldukça düşükken, Batch Size 128’e çıkarıldığında yüksek bir artış gözlemlenmiştir. Bu, PyTorch’un küçük Batch Size ile oldukça verimli çalıştığını, ancak büyük Batch Size ile GPU’ nun sınırlarına yaklaşabileceğini veya bellek yönetiminde sorunlar yaşayabileceğini göstermektedir. Tensorflow’un ise Batch size büyüdükçe, Batch Time’ın nispeten daha stabil bir şekilde arttığını, ancak 64’ten 128’e geçişte belirgin bir yükseliş yaşandığı görülmektedir. Bu, TensorFlow’un daha büyük Batch Size’larda daha fazla zaman harcadığını, ancak yine de PyTorch’a göre daha tutarlı olduğunu göstermektedir. Keras (TensorFlow Backend), TensorFlow’a oldukça yakın bir performans sergilemekle birlikte, genel olarak biraz daha yüksek Batch Time ile çalışmaktadır. Bu, Keras’ın TensorFlow’a göre biraz daha fazla overhead (ek yük) getirdiğini göstermektedir.



Şekil 6. Jellyfish verisetinde Pytorch, Tensorflow ve Keras Uygulama Geliştirme Çatılarının farklı Batch Size değerlerinde Batch Time analizi

Şekil 7’de Jellyfish verisetinde Pytorch, Tensorflow ve Keras Uygulama Geliştirme Çatılarının farklı Batch Size değerlerinde Epoch Time analizi gerçekleştirilmiştir. Şekil 7’ye göre Pytorch, her bir Batch Size için en düşük Epoch Time değerine sahiptir, bu da bu Uygulama Geliştirme Çatısının verimli bir şekilde çalıştığını ve daha az işlem süresi gerektirdiğini ortaya koymaktadır. Batch size arttıkça PyTorch’un Epoch Time değeri düzenli bir şekilde azalmıştır, bu da daha büyük Batch Size ile eğitim süresinin verimliliğinin arttığını göstermektedir. Özellikle Batch size 128’e ulaştığında en düşük Epoch Time değerine sahiptir. TensorFlow, küçük Batch Size (16 ve 32) için PyTorch’a kıyasla daha yüksek Epoch Time değerine sahiptir. Ancak Batch Size arttıkça, Epoch Time daha stabil bir şekilde düşmektedir. Keras, her Batch Size için en yüksek Epoch Time değerine sahiptir. Bu, Keras’ın TensorFlow üzerinde bir üst katman olarak ek bir yük oluşturduğunu ve bunun performansı olumsuz etkilediğini ortaya koymaktadır. Büyük Batch size’larda Keras’ın Epoch time’ı azalmaktadır, ancak TensorFlow ve PyTorch’a göre daha yüksektir.



Şekil 7. Jellyfish verisetinde Pytorch, Tensorflow ve Keras Uygulama Geliştirme Çatılarının farklı Batch Size değerlerinde Epoch Time analizi

Literatürde Uygulama Geliştirme Çatılarının performans karşılaştırmasının gerçekleştirildiği birçok çalışma mevcuttur. Yapıcı ve Topaloğlu çalışmalarında iki farklı verisetini; TensorFlow, Keras (TensorFlow Backend), Theano, Keras (Theano Backend), Torch ile 200, 100 ve 50 Batch size parametrelerine göre CNN ile analiz etmişlerdir (Yapıcı & Topaloğlu, 2021). Analizler sonucu en iyi Uygulama Geliştirme çatısının Tensorflow olduğunu belirtmişlerdir. Shi ve arkadaşları; gerçekleştirdikleri çalışmada, Tensorflow'un Pytorch'tan daha iyi performans gösterdiğini belirtmişlerdir (Shi et al., 2017). Alrfou ve arkadaşları; Mutlu ve arkadaşları ve Shi ve arkadaşları gibi, Tensorflow'un RNN mimarisinde Theano ve Pytorch'tan daha üstün performans sergilediğini belirtmişlerdir. Fakat CNN mimarisinde ise en iyi Uygulama Geliştirme Çatısının Pytorch olduğunu gözlemlemişlerdir (Team et al., 2016). Shatnawi ve arkadaşları, çalışmalarında iki farklı veriseti ile farklı Batch Size değerlerinde TensorFlow, Theano ve CNTK için performans karşılaştırması gerçekleştirmişlerdir (Shatnawi et al., 2018). Gerçekleştirdikleri deneylerinin çoğunda, CNTK'nın uygulamalarının diğerlerinden daha üstün performans sergilediğini ifade etmişlerdir. Wu ve arkadaşları çalışmalarında; farklı verisetleri ile Caffe, Pytorch ve Theano Uygulama Geliştirme çatılarını hem CPU hem de GPU üzerinde, hiperparametreler yönünden ve farklı Batch

Time yönünden geniş bir şekilde değerlendirmişlerdir (Wu et al., 2022). Sonuçlar her bir uygulamada farklılık göstermiştir. Literatür incelendiğinde, farklı Uygulama Geliştirme Çatılarının, farklı Python sürümlerini desteklemesi nedeniyle; çoğu çalışmanın aynı GPU ve CUDA versiyonları ile değil, farklı versiyonları ile gerçekleştirildiği görülmüştür. Bu durum, sonuç ve değerlendirmelerde farklılıklara yol açmıştır. Ancak, bu çalışmanın öne çıkan yönlerinden biri, aynı GPU üzerinde aynı özelliklere sahip cihazlarla değerlendirmelerin yapılmış olmasıdır, bu da karşılaştırmada daha tutarlı ve güvenilir sonuçlar elde edilmesini sağlamıştır. Ayrıca, farklı boyutlardaki veri setlerinin, aynı GPU ve parametreler kullanılsa bile Uygulama Geliştirme Çatılarının performansını etkileyebileceği görülmüştür (Wu et al., 2022).

KAYNAKÇA

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., ... Research, G. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. www.tensorflow.org.
- Chollet, F. (2018). Keras: The Python Deep Learning library. *Ascl*, ascl:1806.022.
<https://ui.adsabs.harvard.edu/abs/2018ascl.soft06022C/abstract>
- Dozat, T. (2016). Incorporating Nesterov Momentum into Adam. *ICLR Workshop, 1*, 2013–2016.
- Duchi JDUCHI, J., & Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization * Elad Hazan. *Journal of Machine Learning Research*, 12, 2121–2159.
- Jellyfish Image Dataset*. (n.d.). Retrieved August 11, 2024, from <https://www.kaggle.com/datasets/anshtanwar/jellyfish-types>
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., & Darrell, T. (2014). Caffe: Convolutional Architecture for Fast Feature Embedding. *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, 675–678.
<https://doi.org/10.1145/2647868.2654889>
- Kingma, D. P., & Lei Ba, J. (2015). *ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION*.
- Papers With Code : Trends | Papers With Code*. (n.d.). Retrieved August 5, 2024, from <https://paperswithcode.com/trends>
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., ... Chintala, S. (2019). PyTorch: An Imperative Style, High-

- Performance Deep Learning Library. *Advances in Neural Information Processing Systems*, 32. <https://arxiv.org/abs/1912.01703v1>
- Robbins, H., & Monro, S. (1951). A Stochastic Approximation Method. *https://doi.org/10.1214/Aoms/1177729586*, 22(3), 400–407. <https://doi.org/10.1214/AOMS/1177729586>
- Şeker, A., Diri, B., & Balık, H. (2017). Derin Öğrenme Yöntemleri ve Uygulamaları Hakkında Bir İnceleme. *Gazi Journal of Engineering Sciences*, 3(3), 47–64. <http://dergipark.gov.tr/gmbd/issue/31064/372661>
- Shatnawi, A., Al-Bdour, G., Al-Qurran, R., & Al-Ayyoub, M. (2018). A comparative study of open source deep learning frameworks. *2018 9th International Conference on Information and Communication Systems, ICICS 2018, 2018-January*, 72–77. <https://doi.org/10.1109/IACS.2018.8355444>
- Shi, S., Wang, Q., Xu, P., & Chu, X. (2017). Benchmarking state-of-the-art deep learning software tools. *Proceedings - 2016 7th International Conference on Cloud Computing and Big Data, CCBD 2016*, 99–104. <https://doi.org/10.1109/CCBD.2016.029>
- Team, T. T. D., Al-Rfou, R., Alain, G., Almahairi, A., Angermueller, C., Bahdanau, D., Ballas, N., Bastien, F., Bayer, J., Belikov, A., Belopolsky, A., Bengio, Y., Bergeron, A., Bergstra, J., Bisson, V., Snyder, J. B., Bouchard, N., Boulanger-Lewandowski, N., Bouthillier, X., ... Zhang, Y. (2016). *Theano: A Python framework for fast computation of mathematical expressions*. <https://arxiv.org/abs/1605.02688v1>
- The Microsoft Cognitive Toolkit - Cognitive Toolkit - CNTK | Microsoft Learn.* (2017). <https://learn.microsoft.com/en-us/previous-versions/cognitive-toolkit/>
- The Theano Development Team, Al-Rfou, R., Alain, G., Almahairi, A., Angermueller, C., Bahdanau, D., Ballas, N., Bastien, F., Bayer, J., Belikov, A., Belopolsky, A., Bengio, Y., Bergeron, A., Bergstra, J., Bisson, V., Snyder, J. B., Bouchard, N., Boulanger-Lewandowski, N., Bouthillier, X., ... Zhang, Y. (2016). *Theano: A Python framework for*

fast computation of mathematical expressions. 1–19.
<http://arxiv.org/abs/1605.02688>

Wu, Y., Liu, L., Pu, C., Cao, W., Sahin, S., Wei, W., & Zhang, Q. (2022). A Comparative Measurement Study of Deep Learning as a Service Framework. *IEEE Transactions on Services Computing*, 15(1), 551–566.
<https://doi.org/10.1109/TSC.2019.2928551>

Yapıcı, M. M., & Topaloğlu, N. (2021). Performance comparison of deep learning frameworks. *Computers and Informatics*, 1(1), 1–11.
<https://dergipark.org.tr/tr/pub/ci>

Zeiler, M. D. (2012). *ADADELTA: An Adaptive Learning Rate Method*.
<https://arxiv.org/abs/1212.5701v1>

BÖLÜM 3

DERİN ÖĞRENME İLE PLC KONTROLLÜ DELTA ROBOT PET ŞİŞE AYRIŞTIRMA SİSTEMİ UYGULAMASI

Mohammed Elkhadir CHACHOUA¹
(0009-0004-6069-673X)

Doç. Dr. Sıtkı ÖZTÜRK²
(0000-0003-3804-5581)

DOI: <https://dx.doi.org/10.5281/zenodo.13825221>

¹ Kocaeli Üniversitesi, Mühendislik Fakültesi, Elektronik ve Haberleşme Mühendisliği Bölümü, Kocaeli, Türkiye, 225103018@kocaeli.edu.tr

² Kocaeli Üniversitesi, Mühendislik Fakültesi, Elektronik ve Haberleşme Mühendisliği Bölümü, Kocaeli, Türkiye, sozturk@kocaeli.edu.tr

1. GİRİŞ

Paralel delta robot 1990'lı yıllarda profesör Reymond Clavel tarafından tasarlanmış, 1999 yılında altın robot ödülünü almış ve aynı yıl Asea Brown Boveri (ABB) şirketi tarafından üretilmeye başlanmıştır [Clavel, 1991]. Delta robotların birçok türü yapısal ve işlevsel olarak birbirinden farklılık göstermektedir [Gasparetto, 2019]. Delta robotlar, hızlı ve hassas hareketleri nedeniyle endüstri 4.0 alanında giderek daha önemli hale gelmektedir. Programlanabilir mantıksal denetleyicileri (PLC) gibi endüstriyel kontrol sistemleri kullanılarak çalıştırılabilirler. Entegre edilebilen çeşitli kamera uygulamaları sayesinde paralel delta robot, ürün ayırma, sıralama, seçme, aktarma, 3D baskı ve paketleme sistemleri gibi endüstride yaygın olarak kullanılmaktadır [Bonev, 2001].

Delta robot, paralel (kapalı döngü) robotların bir türüdür [Guglielmetti, 1994]. Bu robotların seri (açık döngü) robotlara kıyasla birçok avantajı ve dezavantajı vardır, en büyük dezavantajlarından biri küçük bir kullanılabilir çalışma alanında çalışabilmeleridir. Bununla birlikte, paralel manipülatörler, eklemlerin konumunun devamlılığı veya daha yüksek sertliği aynı zamanda yüksek doğruluğu gibi başka avantajlar da sunmaktadır [Nzue et al., 2013].

Geleneksel Fabrikalar ayırıştırmak için manuel toplama, yerleştirme ve paketleme sürecini kullanır, bu da daha fazla zaman ve iş gücü gerektireceğinden pahalı ve uzun süreçlerdir. Ayrıca, insan kaynaklarını da oldukça tüketmektedir. Bu çalışmada ayırıştırma görevlerinde doğruluğu ve istikrarı artırmak için, daha az maliyetli bir sistem için, otomatik bilgisayarlı görü destekli bir delta robot sisteminin gerçekleştirilmesi üzerinde çalışılmıştır.

Önceki yapılan çalışmalardan, F. Erzincanlı ve J.M. Sharp (Ağustos 1997) robotik bir gıda ürünü sınıflandırma sistemi geliştirmiştir. Bu ürün sınıflandırması, her bir ürünün fiziksel özelliklerine dayanmaktadır. Her bir ürünün fiziksel özelliklerini göz önünde bulundurarak robotik uç işlevlerinin nesnelere zarar vermeden tutabileceğini ve hareket ettirebileceği gösterilmiştir [Erzincanlı vd., 1997]. 2016 yılında Chih-Jer Lin ve arkadaşları, bir konveyör bant üzerindeki hareketli iş parçalarını şekillerine göre ayırıştırmak ve yerleştirmek için delta-paralel robot kullanan image-servo sistemi önermişlerdir. Bu sistemin görüntü işlemlerini sobel ve canny kenar algılama teknikleri kullanarak gerçekleştirmişlerdir [Lin et al., 2016]. Daha sonra Vo Duy Cong ve arkadaşları, görüntü işleme tekniklerini kullanarak nesnelere

renklerine göre sınıflandırmak için paralel bir delta robot kullanan bir sistem tasarlanmış ve geliştirmiştir. Raspberry Pi 4 bilgisayarı kullanılarak kamera ile konveyör bant üzerindeki hareketli nesnelerin görüntülerini alınıp işlenerek nesnelere renklerine göre ayırtmışlardır [Cong vd., 2023].

Plastik atık ayrıştırma görevlerinde robotik sistemlerin kullanılması için bilgisayarlı görü destekli robotik kontrol sistemi araştırılmıştır, ancak yaygın olarak kullanılmadığı tespit edilmiştir. Bu çalışmada eski robotik ayrıştırma çalışmalarının kullanılabilirliğini artırmak için derin öğrenme tabanlı bilgisayarlı görü destekli delta robot PET şişe ayrıştırma sistemi üzerinde çalışılmıştır. Bu çalışma aşağıdaki gibi alt bölümlerden oluşmaktadır. İlk olarak, robotun koordinat sistemlerinin atamasının yapıldığı ve ilgili sembollerin verildiği kinematik model açıklanmıştır. Ardından, paralel delta robotun ileri kinematik ve ters kinematik analizi gerçekleştirilmiştir. Ters kinematik analizinin sonuç denklemleri, gerekli konum koordinatlarına dayalı olarak aktif eklem açılarını hesaplamak için kullanılır ve bunun tersi de geçerlidir. Daha sonra Servo motorların PLC ile pozisyon kontrolü ele alınmıştır. Ondan sonra delta robotun tespit edilen PET şişeleri almasını ve bunları ayırma alanına yerleştirmesini sağlamak için robotik sistem parçasını yönlendirmek üzere bilgisayarlı görü sistemi tasarlanmıştır. Bu çalışmada görsel işleme, konumlandırma ve hareket planlama sistem kodları MATLAB programlama dili kullanılarak oluşturulmuştur. Görüntü sınıflandırma tekniği olarak You Only Look Once version 2 (YOLOv2) algoritması kullanılmıştır.

2. ROBOTİK KONTROL SİSTEMİ

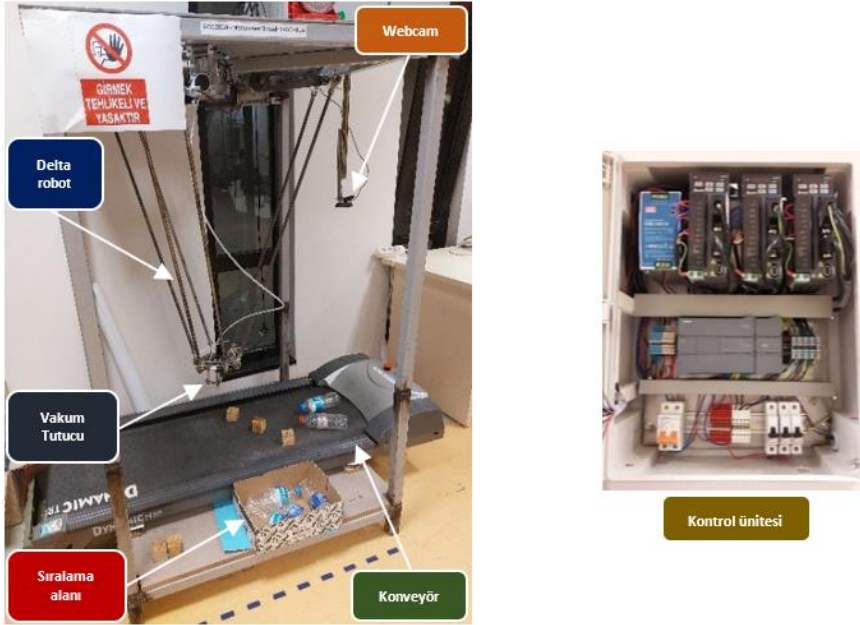
2.1 Delta Robot

Bu çalışmada, paralel bir robotun bir konveyör bant üzerindeki hareketli nesnelere (Plastik şişeler) alma ve yerleştirme görevlerini yerine getirmesini sağlamak için derin öğrenme tabanlı bir bilgisayarlı görü destekli robotik sistem üzerinde çalışılmıştır. Bu sistemin uygulanmasında paralel bir delta robot kullanılmıştır. Bu delta robotun tasarımı ve mekanik modeli Schneider Elektrik [Schneider Electric, 2023] tarafından geliştirilen modelden uyarlanmıştır. Bu model Şekil 2.1'de gösterilmektedir.



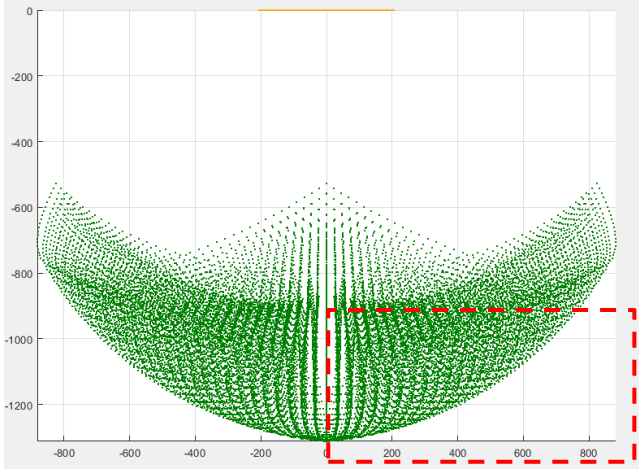
Şekil 2.1. Paralel delta robotun 3D modeli [Schneider Electric, 2023].

Uyarlanan robotun mekanik yapısında, üç AC servo motor ve bu motorları süren servo sürücü ve sistemin emniyeti ve referans bilgilerini sağlamak için ilave sensörlerden oluşmaktadır. Kullanılan motorların temel amacı manipülâtörün aktif eklemlerindeki hareketliliği sağlamaktır. Bu paralel robot üç uzuvdan oluşmaktadır, her uzuvda bir döner eklem ve iki küresel pasif eklem bulunmaktadır. Bu çalışmada kullanılan robotik sistemin nihai kurulumu Şekil 2.2'de verilmiştir.

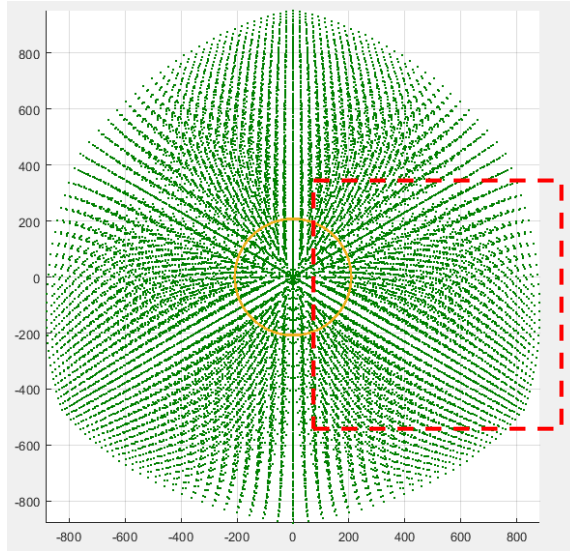


Şekil 2.2. Tasarlanan robotik sistem ve kontrol panosu

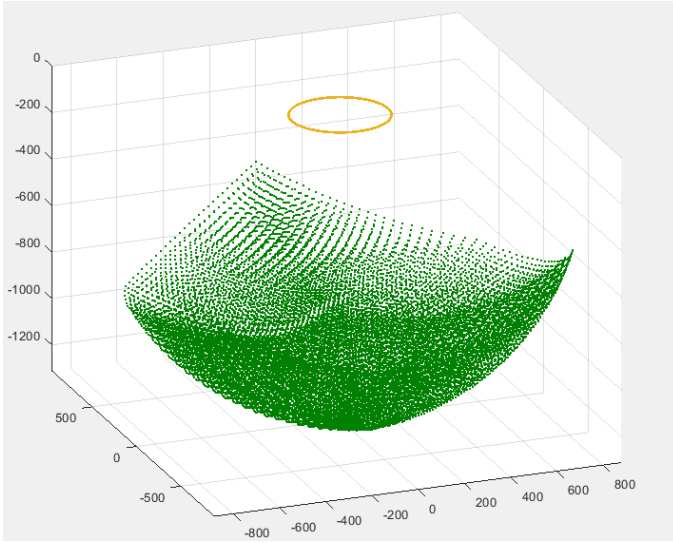
Elde edilen ters ve ileri kinematik hesaplama denklemleri yardımıyla (Bölüm 2.2), gerçek sistem uygulamasının mekanik sınırlamaları dikkate alınarak robotun çalışma alanı belirlenmektedir [Smirnova, 2023]. Delta robotun gerçek döngü süresinin çalışma koşullardan etkilendiğini belirtmek de önemlidir. Ayrıca, manipülatörün hareket aralığı (Şekil 2.3 ve 2.4'te gösterildiği gibi) Al-Bırak (Pick-and-Place) işlemlerinde önemli bir rol oynamaktadır.



Şekil 2.3. XZ düzlemindeki delta robot çalışma alanı.



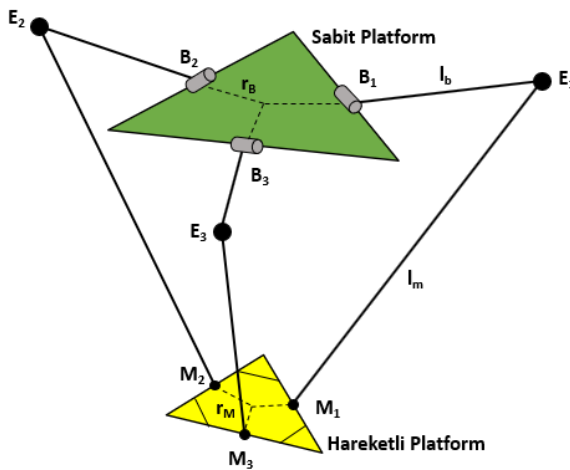
Şekil 2.4. XY düzlemindeki delta robot çalışma alanı.



Şekil 2.5. Üç boyutlu delta robot çalışma alanı.

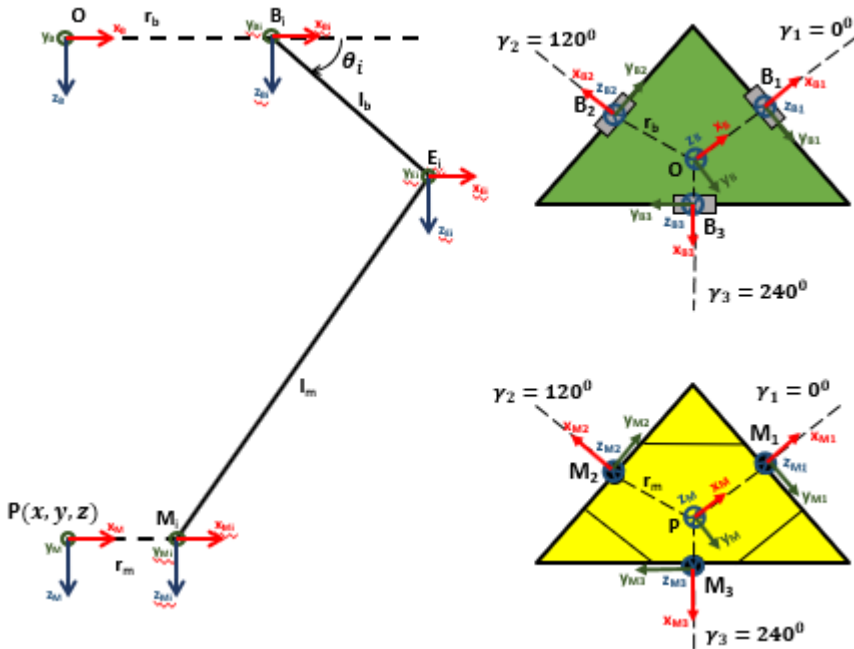
2.2 Kinematik Analizi

Kinematik modelin temel amacı, uç işlevcinin konumu verildiğinde eklem açılarını hesaplamak ve bunun tersini yapmaktır. Eklem açılardan konumun koordinatları hesaplandığı işleme ileri veya manipülatör kinematiği denir. Ters kinematik analiz ise istenen uç işlevcinin konum koordinatları verildiğinde eklem açılarının hesaplanmasıdır [Doğan, 2010], [Robert, 2016]. Paralel delta robotun mekanik çizimi Şekil 2.5'te verilmiştir.



Şekil 2.6. Delta robotun mekanizma gösterimi.

Şekil 2.6'da gösterildiği gibi paralel delta robot, eşkenar üçgen şeklinde bir platform ile tavana sabitlenmiştir. Bu platforma (üst platform) atanan koordinat sistemi O merkez noktasına yerleştirilir ve "Temel çerçeve" veya "Sıfır çerçeve" olarak kabul edilmektedir. Robot aynı zamanda hareketli bir platformdan (alt platform) oluşmaktadır. Bu platformun ağırlık merkezi P (x, y, z) işlem noktası veya bu paralel robotun uç işlevçisi olarak kabul edilmektedir. Bu robotu oluşturan platformlar üç kol yardımıyla birbirine bağlanmıştır. Bu kollar ortada bulunan küresel mafsallar (E1, E2, E3) sayesinde içe ve dışa doğru bükülebilmektedir. Sabit platformun ve hareketli platformun sırasıyla her bir kenarının ortasında bulunan döner mafsallar (B1, B2, B3) ve küresel mafsallar (M1, M2, M3) platformun kolların ucuyla bağlantısını sağlamaktadır. Şekil 2.7'da koordinat sistemlerinin ataması yapılmış ve ilgili semboller verilmiştir. Gerekli çerçeveler hesaplamayı mümkün olduğunca azaltacak şekilde atanmıştır, koordinat sistemlerinin doğru şekilde atanması ve sembollerin seçilmesi modelin daha iyi çözülmesi ve anlaşılması için çok önemlidir.



Şekil 2.7. Koordinat sistemlerinin ataması.

Koordinat sistemleri atamasının özeti Tablo 1, 2 ve 3'te verilmiştir.

Koordinat sistemi	Orjin	Koordinat eksenleri
{O}	O	x_B, y_B, z_B
{B ₁ }	B ₁	x_{B1}, y_{B1}, z_{B1}
{B ₂ }	B ₂	x_{B2}, y_{B2}, z_{B2}
{B ₃ }	B ₃	x_{B3}, y_{B3}, z_{B3}

Tablo 2.1. Sabit platform koordinat sistemleri

Koordinat sistemi	Orjin	Koordinat eksenleri
{E ₁ }	E ₁	x_{E1}, y_{E1}, z_{E1}
{E ₂ }	E ₂	x_{E2}, y_{E2}, z_{E2}
{E ₃ }	E ₃	x_{E3}, y_{E3}, z_{E3}

Tablo 2.2. Kolların orta eklemler koordinat sistemleri

Koordinat sistemi	Orjin	Koordinat eksenleri
{P}	P	x_M, y_M, z_M
{M ₁ }	M ₁	x_{M1}, y_{M1}, z_{M1}
{M ₂ }	M ₂	x_{M2}, y_{M2}, z_{M2}
{M ₃ }	M ₃	x_{M3}, y_{M3}, z_{M3}

Tablo 2.3. Hareketli platform koordinat sistemleri

Şekil 2.6'da yapılan tanımlamalara dayanarak, koordinat sistemleri arasında aşağıdaki dönüşüm matrisleri bulunur.

{O} ve {E_i} koordinat sistemleri arasındaki ${}^{KS_{OT}}_{KS_{Ei}}$ dönüşüm matrisi:

$${}^{KS_{OT}}_{KS_{Ei}} = \begin{bmatrix} \cos \gamma_i & -\sin \gamma_i & 0 & \cos \gamma_i l_b \cos \theta_i + r_b \cos \gamma_i \\ \sin \gamma_i & \cos \gamma_i & 0 & \sin \gamma_i l_b \cos \theta_i + r_b \sin \gamma_i \\ 0 & 0 & 1 & l_b \sin \theta_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

{O} ve {M} koordinat sistemleri arasındaki ${}^{KS_O}T_{KS_M}$ dönüşüm matrisi:

$${}^{KS_O}T_{KS_M} = \begin{bmatrix} \cos \gamma_i & -\sin \gamma_i & 0 & r_m \cos \gamma_i + x \\ \sin \gamma_i & \cos \gamma_i & 0 & r_m \sin \gamma_i + y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

{O} ve {P} koordinat sistemleri arasındaki ${}^{KS_O}T_{KS_P}$ dönüşüm matrisi:

$${}^{KS_O}T_{KS_P} = \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

Burada; $\gamma_i = 0^0, 120^0, 240^0$ olarak alınmıştır.

Şekil 2.5'te gösterildiği gibi E ve M noktaları arasındaki mesafe l_m 'ye eşit olduğundan vektörün büyüklüğü \overline{EM} :

$$\|\overline{EM}\|^2 = l_m^2 \quad (4)$$

eşittir.

(1), (2) ve (3) numaralı denklemlerdeki dönüşüm matrislerinin konum vektörlerini kullanarak ve $\overline{EM} = \overline{OM} - \overline{OE}$ vektör denklemine dayanarak, \overline{EM} vektörün koordinatları:

$$\overline{EM} \begin{pmatrix} x - \cos \gamma_i l_b \cos \theta_i - r_b \cos \gamma_i \\ y - \sin \gamma_i l_b \cos \theta_i - r_b \sin \gamma_i \\ z - l_b \sin \theta_i \end{pmatrix}$$

eşitliğinden bulunmaktadır. Vektörler arası mesafe formülünden yararlanarak, vektörün koordinat değerleri \overline{EM} denklem (4)'te yerine konulduğunda, ortaya çıkan denklem (5) deki gibi elde edilecektir.

$$(x - \cos \gamma_i l_b \cos \theta_i - r_b \cos \gamma_i)^2 + (y - \sin \gamma_i l_b \cos \theta_i - r_b \sin \gamma_i)^2 + (z - l_b \sin \theta_i)^2 = l_m^2 \quad (5)$$

Denklem (5) daha sonra sadeleştirilerek:

$$(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 = l_m^2 \quad (6)$$

Elde edilmektedir. Burada;

$$\begin{aligned} x_i &= (r + l_b \cos \theta_i) \cos \gamma_i, & y_i &= (r + l_b \cos \theta_i) \sin \gamma_i, \\ z_i &= l_b \sin \theta_i \end{aligned}$$

olur.

Bu sonuçlardan vektörün koordinat değerleri $\overrightarrow{EM}_i \begin{pmatrix} x - x_i \\ y - y_i \\ z - z_i \end{pmatrix}$ elde edilir.

2.2.1 İleri Kinematik

İleri kinematik hesaplamalarda, uç işlevcinin konum koordinatları manipülatörün eklem açılarından yola çıkarak çözülür, paralel robotlarda, uç işlevci hareketli platformun ağırlık merkezinde yer almaktadır.

Denklem (6)'daki i değeri ile yer değiştirildiğinde aşağıdaki üç denklem elde edilir:

$$(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 = l_m^2 \quad (7)$$

$$(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = l_m^2 \quad (8)$$

$$(x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 = l_m^2 \quad (9)$$

(7) ve (8) denklemlerini kullanarak, x 'in z değeri cinsinden çözümü ise:

$$x = \frac{a_1}{d_1} z + \frac{b_1}{d_1} \quad (10)$$

elde edilir. Burada;

$$a_1 = (z_3 - z_2)(y_2 - y_1) - (z_2 - z_1)(y_3 - y_2)$$

$$b_1 = \left(\frac{w_2 - w_1}{2}\right)(y_3 - y_2) - \left(\frac{w_3 - w_2}{2}\right)(y_2 - y_1)$$

$$d_1 = (x_2 - x_1)(y_3 - y_2) - (x_3 - x_2)(y_2 - y_1)$$

olarak alınmıştır.

Aynı adımlar izlenerek, (8) ve (9) denklemleri z değeri cinsinden y 'nin çözümünü elde etmek için kullanılır, y 'nin değeri ise:

$$y = \frac{a_2}{d_2}z + \frac{b_2}{d_2} \quad (11)$$

elde edilir. Burada ise;

$$a_2 = (z_3 - z_2)(x_2 - x_1) - (z_2 - z_1)(x_3 - x_2)$$

$$b_2 = \left(\frac{w_2 - w_1}{2}\right)(x_3 - x_2) - \left(\frac{w_3 - w_2}{2}\right)(x_2 - x_1)$$

$$d_2 = (x_3 - x_2)(y_2 - y_1) - (x_2 - x_1)(y_3 - y_2)$$

olarak alınmıştır.

Son olarak, denklem (10) ve (11)'da bulunan x ve y değerlerini denklem (6)'da yerine koyarak, bir bilinmeyenli z ile ikinci dereceden bir denklem elde edilmektedir.

$$\left[\frac{a_1^2}{d_1^2} + \frac{a_2^2}{d_2^2} + 1\right]z^2 + \left[\frac{2a_1b_1}{d_1^2} + \frac{2a_2b_2}{d_2^2} + \frac{2a_1x_1}{d_1} - \frac{2a_2y_1}{d_2} - 2\right]z + \left[\frac{b_1^2}{d_1^2} + \frac{b_2^2}{d_2^2} + \frac{2b_1x_1}{d_1} - \frac{2b_1y_1}{d_2} - l_b^2 - w_1\right] = 0$$

Bu ikinci derece denklem çözülerek z değeri bulunmaktadır. Bulunan z değeri (10) ve (11) numaralı denklemlerde yerine konularak x ve y değerleri hesaplanmaktadır. Böylece P noktasının konumu (x, y, z) temel koordinat sistemine göre bulunmuş olmaktadır.

2.2.2 Ters kinematik

Manipülâtörün istenen konuma hareket etmesi için gereken eklem açılarını bulmak amacıyla, bu paralel manipülâtörün ters kinematiği çözülmektedir. Bunu yapmak için, denklem (5) genişletilerek $\sin \theta_i$ ifadesi bulunmaktadır. Daha sonra $\sin \theta_i$ değeri $\cos^2 \theta_i + \sin^2 \theta_i = 1$ formülünde yerine yerleştirilir, bu şekilde tek bilinmeyenli ikinci dereceden bir denklem elde edilmektedir.

$$a \cos^2 \theta_i + b \cos \theta_i + c = 0 \quad (12)$$

Ardından, θ_i değerleri denklem (12) çözülmesi ve $\theta_i = \text{atan2}(\sin \theta_i, \cos \theta_i)$ formülünde $\cos \theta_i$ değerleri ile yer değiştirilerek bulunmaktadır.

3. PLC İLE SERVO MOTOR POZİSYON KONTROLÜ

PLC S7-1200'de Servo motorları ve Servo sürücülerini kontrol etmek için önceden hazırlanmış komutları kullanma seçeneği bulunmaktadır. Eksenin yapılandırılması ve daha sonra kontrol edilmesindeki kolaylığı gerçekten bu teknolojiyle ilgilenmeye değerdir. Çok hızlı bir şekilde eksen konumlandırma yapılabilen ve bu da birçok uygulamada iyi bir şekilde uygulanabilmektedir.

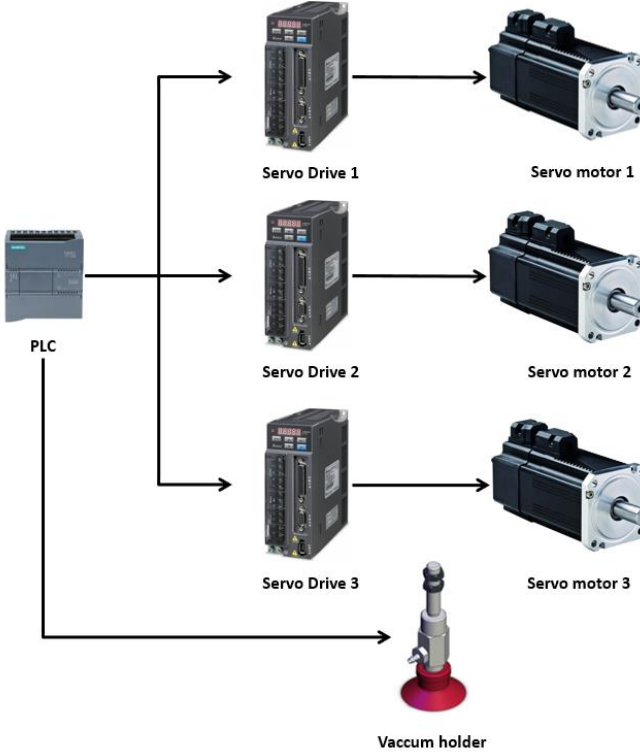
Hareket kontrolü üç farklı yöntemle gerçekleştirilebilir, bu yöntemler **darbe dizisi çıkışı (PTO)**, **analog çıkışı** veya **PROFINET bağlantısı** üzerinden direk sürücünün yazılımına etki ederek. Bu çalışmada PTO çıkışı kullanılarak hareket kontrolü yapılmıştır.

3.1 PLC ile Servo Motor Kontrolünün Kurulması

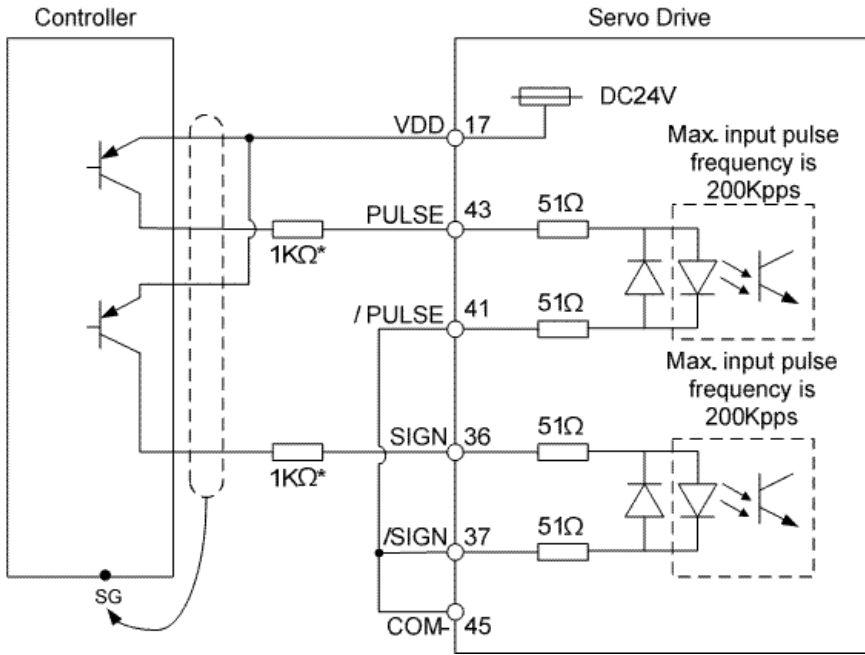
Servo motorlar, konum, hız ve tork kontrolünü kolay bir şekilde kontrol etmek için geliştirilmiş motorlardır. Otomasyon sistemlerinde yaygın olarak kullanılan asenkron motorlara göre hassas motorlardır. Bunlar; bir motor, geri besleme modülü (Enkoder gibi) ve kontrol devresinden oluşur.

PLC ile Servo motor kontrolünün devreye alınması, endüstriyel otomasyonun çok önemli bir yönüdür ve ayrıntılara dikkat edilmesini ve sistematik bir şekilde yürütülmesini gerektirmektedir. Başlangıç olarak, Servo motor kontrolü ile uyumlu bir PLC ve buna karşılık gelen Servo sürücü dahil

olmak üzere gerekli donanım bileşenlerinin bir araya getirilmesi gerekmektedir. Bu bileşenler kontrol sisteminin temel yapısını oluşturarak hassas hareket kontrolünü kolaylaştırmaktadır. Şekil 3.1'de verilen modüller tasarlanan robotik sistemdeki hareketli uç işlevcinin, vakum vantuzun, hareket kontrolünü sağlayan modüller ve sözde bağlantısı verilmiştir. Şekil 3.2'de de PLC ile servo motor sürücü elektriksel bağlantı şeması verilmiştir. Sürücü hem PNP hem de NPN olarak bağlanabildiği için burada PLC'nin çıkış bağlantı şekli önem kazanmaktadır. S7-1200 PLC PNP çıkışlı olduğu için Servo sürücüde PNP bağlantısı tercih edilmiştir. Bu nedenle de sürücünün ortak ucu (COM) eksi (-) olarak bağlanmıştır.



Şekil 3.1. Robotik sistemde kullanılan ekipmanlar



Şekil 3.2. PLC-kontrolör ve Servo sürücü PNP kablo bağlantısı [Delta, 2024].

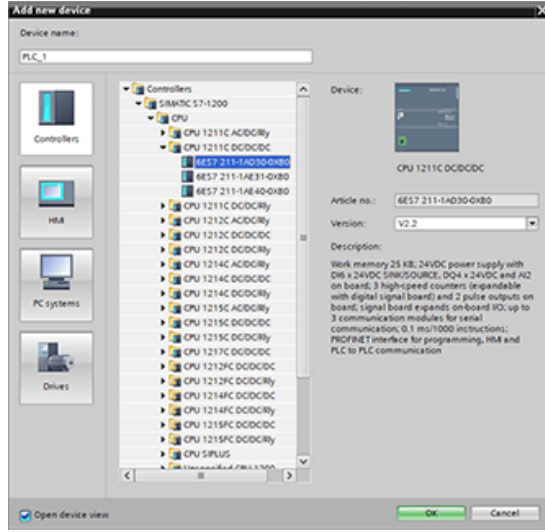
Şekil 3.2'de de gösterildiği gibi elektriksel bağlantılar yapıldıktan sonra, bir sonraki adım PLC ve Servo sürücü arasında haberleşme gerçekleştirmektir. Bu da iki cihaz arasında sorunsuz veri alışverişi sağlayan EtherCAT veya Modbus gibi haberleşme protokollerinin kullanılmasıyla gerçekleştirilmektedir (Lütfen bkz. Şekil 3.2).

Haberleşme başarıyla kurulduktan sonra, son adımda hem PLC hem de Servo sürücü parametreleri istenen hareket şekline ve kontrol gereksinimlerine göre yapılandırılmaktadır. Buna hız, ivme, ivme kaybı ve konumlandırma parametreleri gibi ayarlar dahildir. Bu parametrelerin hassas bir şekilde ayarlanması ile Servo motor kontrolünde optimum performans ve doğruluk sağlayarak uygulamanın özel gereksinimlerini karşılamasına olanak sağlanmaktadır.

3.2 TIA portalda PLC ve servo motor yapılandırması

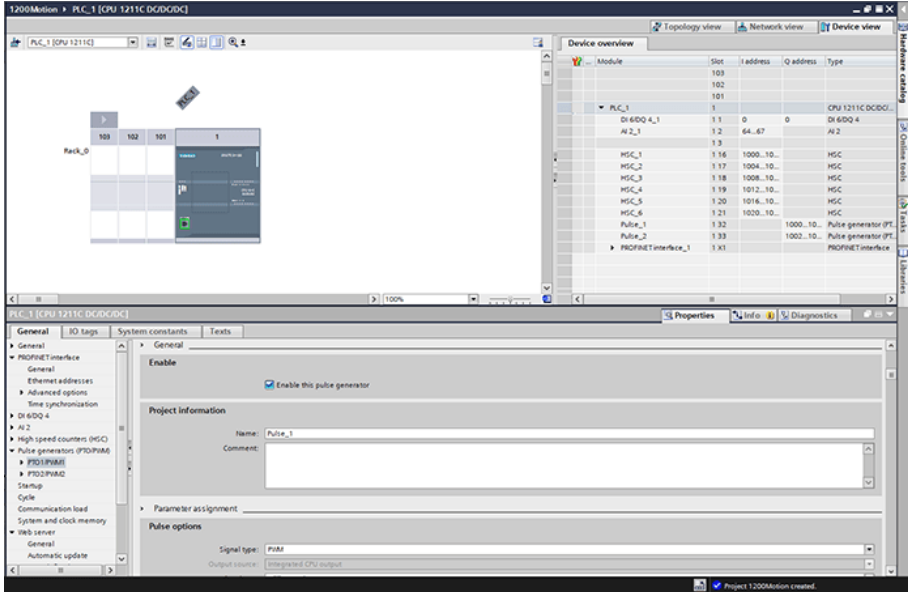
TIA portal Programında Servo motor ile PLC'yi yapılandırmak için aşağıdaki adımlar takip edilmiştir:

İlk olarak, projeye PLC eklenmektedir.



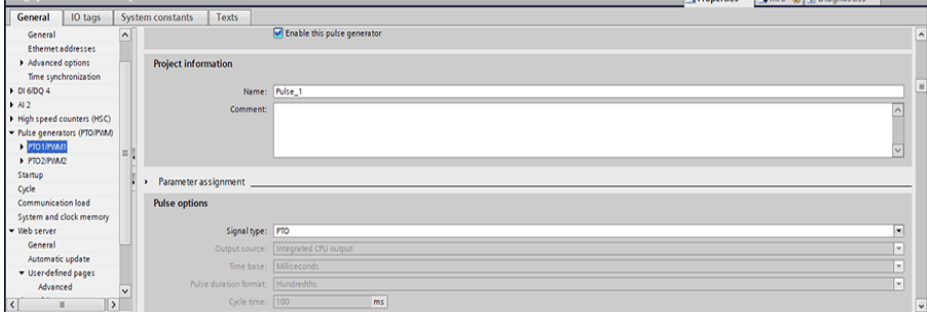
Şekil 3.3. Projeye PLC cihazı ekleme.

Eklendikten sonra, **Cihaz yapılandırması (Device Configuration)** sekmesine gidilerek ilk pals üretici olan **PTO1/PWM1**, **PTO2/PWM2** ve **PTO3/PWM3** etkin hale getirilmesi sağlanmaktadır.



Şekil 3.4. PLC Cihaz yapılandırması.

Servo sürücülerini pals kontrolü ile sürüleceği için projeye pals üreticinin **PTO sinyal tipi** aktif edilmiştir.



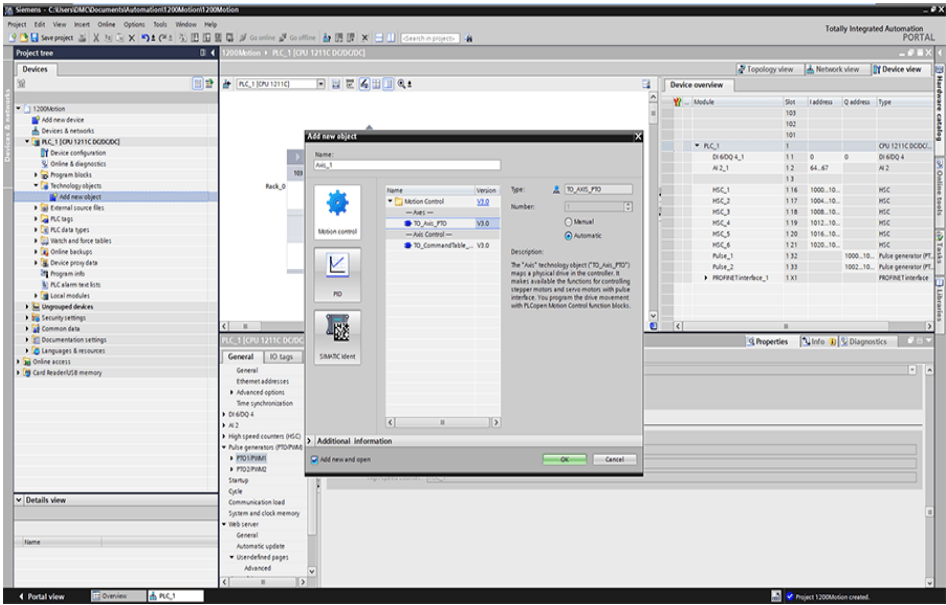
Şekil 3.5. PTO işaretinin seçilmesi.

Pals çıkışının (Pulse Output) Q0.0', **yön çıkışının** (Direction output) Q0.3 ve **Etkinleştirme Çıkışı** (Enable output) Q0.6 olarak seçilmiştir. Bu çalışmada kullanılan 3 Servo için çıkış atamalarının özeti Tablo 3.1'de verilmiştir.

Axis	Pals çıkışı	Yön çıkışı	Etkinleştirme çıkışı
Servo 1	Q0.0	Q0.3	Q0.6
Servo 2	Q0.1	Q0.4	Q0.7
Servo 3	Q0.2	Q0.5	Q1.0

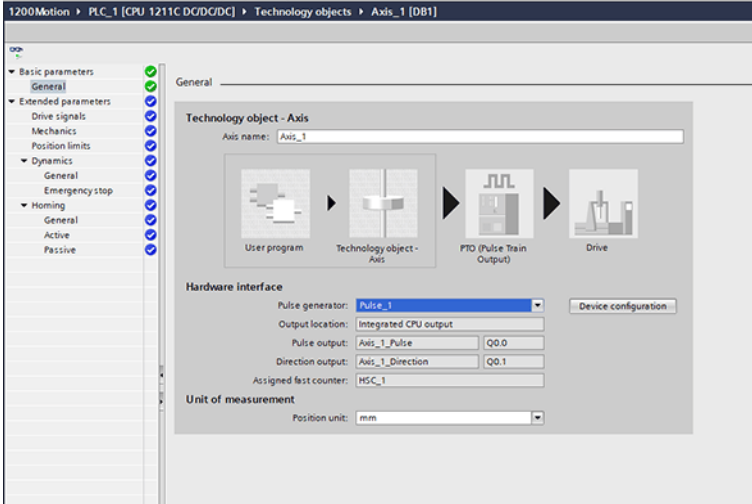
Tablo 3.1. Darbe çıkışı, Yön çıkışı ve Etkinleştirme çıkışı pin ataması.

Bir sonraki aşamada, teknoloji objesinden eksen oluşturulmaktadır. Teknoloji objeleri klasörünün altındaki “Yeni obje ekle” tıklanarak “TO_AXIS_PTO.” seçilmektedir.



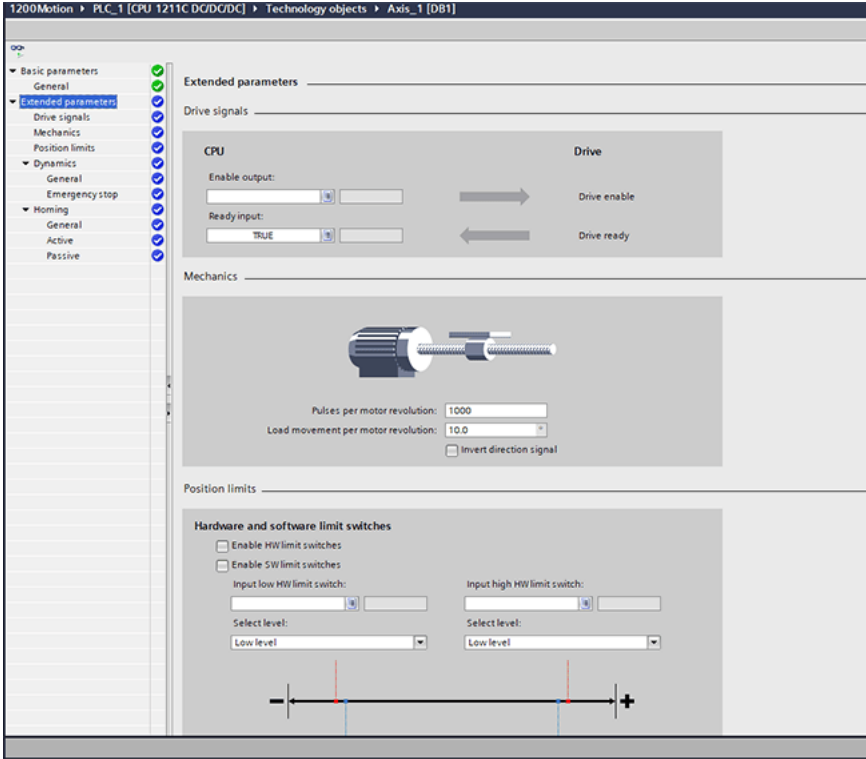
Şekil 3.6. Axis PTO teknoloji objesi ekleme.

Ardından, teknoloji objesi oluşturulduğunda “Pulse_1” seçilmeli ve birimi ise **derece** olarak seçilmektedir.



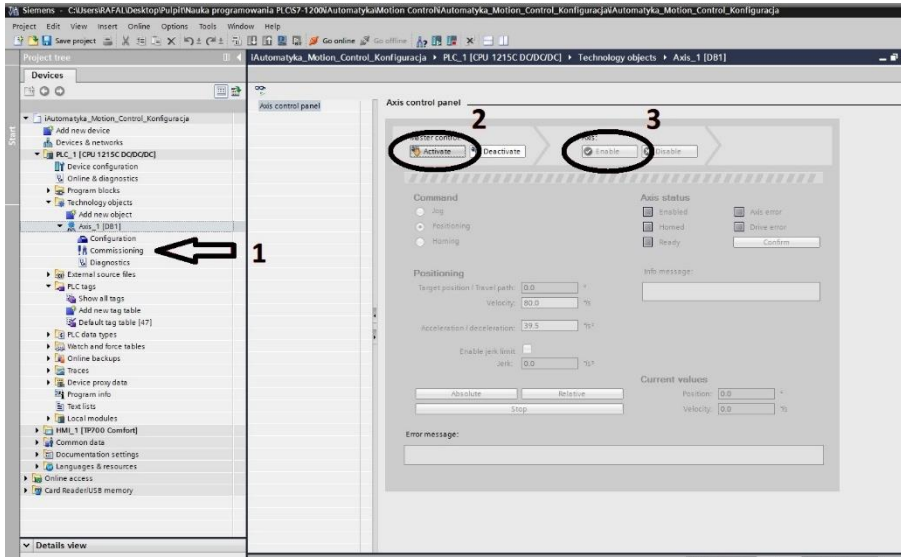
Şekil 3.7. Axis PTO teknoloji objesi yapılandırma.

Son olarak, genişletilmiş parametrelere motor devri başına darbeler ve motor devri başına yük hareketi seçilmektedir. Böylece, programda yazılan komut ile verilen hızların ve konumların doğru olması sağlamaktadır.



Şekil 3.8. Mekanik ve dinamik parametrelerin ayarlanması.

Teknoloji objesi yapılandırıldıktan sonra servo motor programlanmaya hazırdır. Ancak bundan önce, Axis'in çalışıp çalışmadığını görmek için test edilmesi gerekir. Bu amaçla teknoloji objesi içinde comissioning aracı kullanılmaktadır. İlk olarak, axis'in yapılandırması PLC'ye yüklenmiştir. Bundan sonra, proje ağacında, oluşturulan Axis'i seçip ve ardından sürücünün manuel kontrolü için bir panelin verildiği Commissioning (1) sekmesini açılmıştır. PLC ile bağlantı kurulduktan sonra sürücü etkinleştirilmiştir (Activate) (2). Son olarak axis (3) etkinleştirilmiştir (Enable) ve motorun ara yüzden girilen değerlere gittiği gözlemlenmiştir. Bu şekilde sürücü üzerinde kontrol sağlanmış olması, projedeki teknoloji objesi ile sürücünün doğru yapılandırıldığı görülmüş oldu.



Şekil 3.9. Axis Comissioning süreci

Bu çalışmada 3 hareket komut bloğu kullanılmaktadır. Bu komutların detaylarına geçmeden önce, bu komutların durum çıkışlarına (Status Output) dikkat çekmek gerekmektedir. Bu durum çıkışları, hareket komutunun iş yürütmesine ilişkin değerli bilgi sağlamaktadır. Bu bilgiler aşağıdaki gibi verilmiştir:

Done- görev başarıyla tamamlandı,

Busy- görev şu anda işleniyor,

CommandAborted- Görevin yürütme sırasında başka bir iş tarafından sonlandırıldı,

Error- görev işlenirken bir hata oluştu,

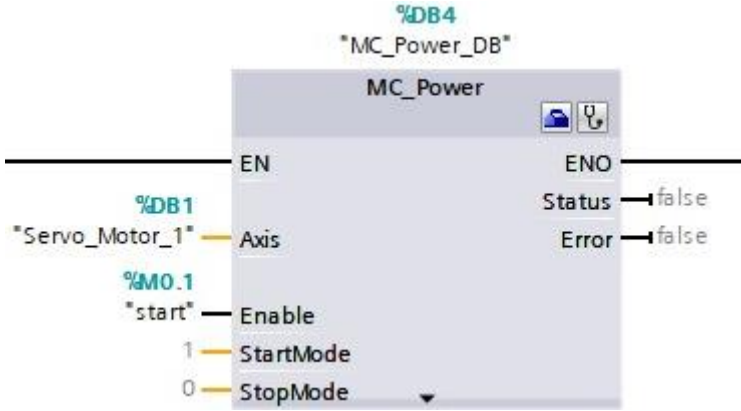
ErrorId- hatanın nedenini hexadecimal formatta görüntüleme,

anlamlarına gelmektedir.

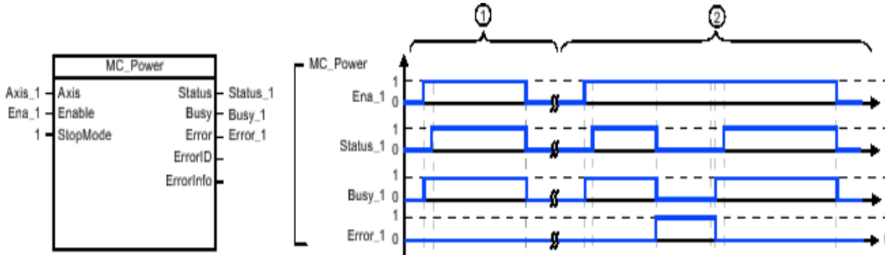
Axis'i programlamada, etkinleştirmek için **MC_POWER**, referanslamak (Homing) için **MC_HOME** ve konumlandırmak için **MC_MOVEABSOLUTE** komutları kullanılabilir.

Bu projede kullanılan PLC kodunda kullanılan hareket kontrol (motion control) komutları aşağıda açıklanmıştır:

1. MC_POWER: Servolar bu komut bloğu olmadan çalışmamaktadır. Hareket koordinatlarını belirleyen veri blokları “Axis” girişlerine bağlanmaktadır. Şekil 3.10 da görülen MC_POWER bloğunun Axis girişine bağlı sürücünün aktif olabilmesi için bloğun “Enable” girişinin aktif olması gerekmektedir.



Şekil 3.10. MC_POWER bloğu.



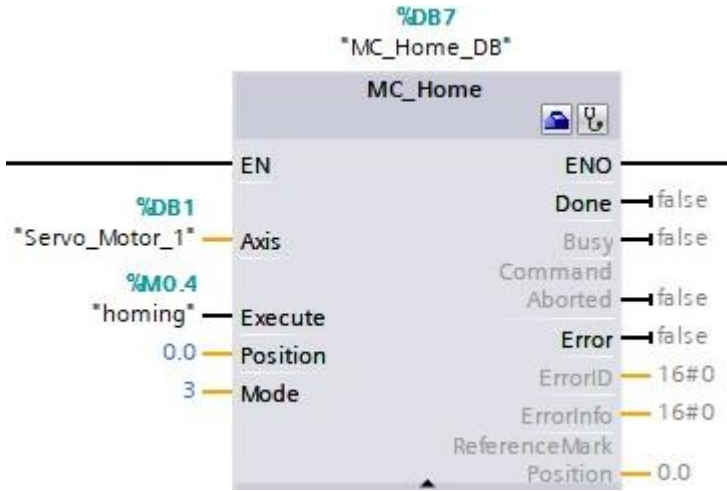
Şekil 3.11. MC_POWER komutu çalışması [Siemens, 2024].

Şekil 3.11'de:

- ①: Bir eksen etkinleştirilir ve sonra tekrar devre dışı bırakılır (Ena_1'de gösterildiği gibi), “Busy_1” yüksektir. Başarılı etkinleştirme “Status_1” üzerinden okunabilir.
- ②: Bir eksen etkinleştirildikten sonra, eksenin devre dışı kalmasına neden olan bir hata meydana gelmiştir (Error_1'de gösterildiği gibi). Hata giderildiğinde eksen kaldığı yerden devam eder.

2. MC_HOME:

Bir Servo ekseninin referanslanması, sisteminizin mekanik kurulumuna referansla bir servo motorun eksenini için bilinen bir konum veya konum belirleme işlemidir. Bu işlem genellikle bir hareket kontrol sisteminin çalışmasının başlangıcında veya bir sistem kapatıldıktan ya da yeniden başlatıldıktan sonra yapılmaktadır. Referanslama, sistemin doğru çalışmasını ve servo motor ekseninin doğru ve güvenilir bir şekilde hareket etmesini sağladığı için hareket kontrol sisteminizi kurmanın ve yapılandırmanın önemli bir parçasıdır. TIA Portal'da, MC_HOME komutu kullanılarak bir servo eksenine referans verilebilmektedir



Şekil 3.12. MC_HOME komut bloğu.

Bir servo eksenine referans vermek için MC_HOME Hareket Kontrolü komutunu kullanırken, Mode girişi aşağıdaki gibi ayarlanabilir:

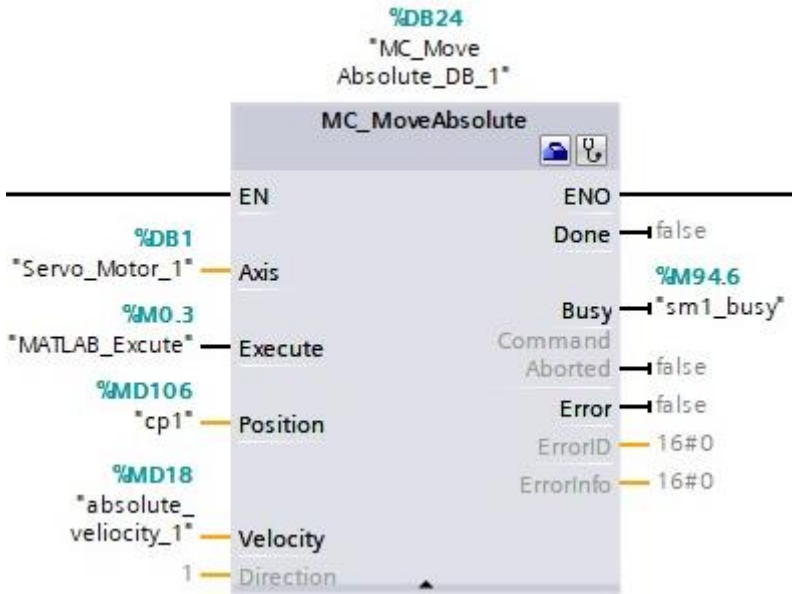
- 0:** Pasif homing.
- 1:** Aktif homing.
- 2:** Gerçek pozisyonu ayarlama.
- 3:** Gerçek pozisyona göreceli ofsetleme.
- 4:** Mutlak enkoder konum ofseti.

Servo eksenini için hangi homing modunun kullanılabileceği enkoder tipine göre değişmektedir. Bu enkoderler ya artımlı enkoder (tek turlu) ya da mutlak enkoderlerdir (çok turlu). Artımlı enkodere sahip bir eksen, güç çevriminden sonra konumunu korumaz. Eksen her güç çevriminden sonra home

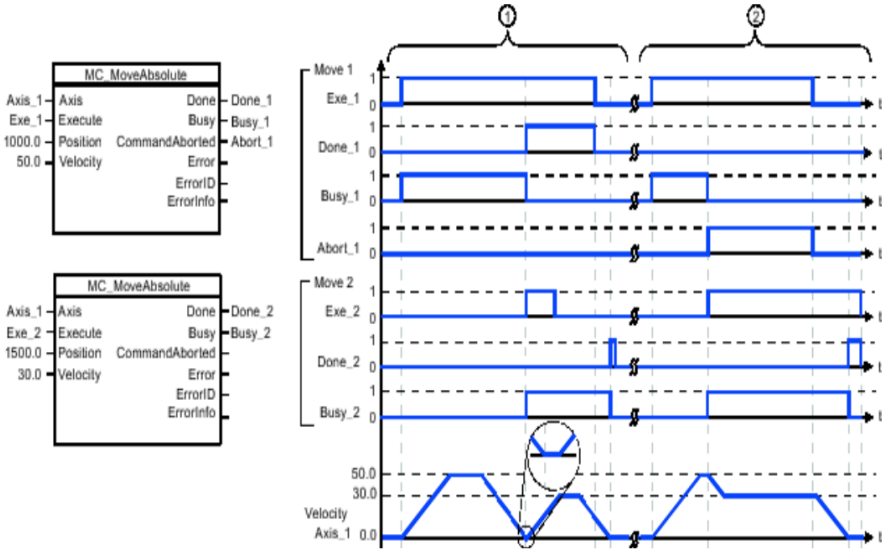
edilmelidir. Mutlak enkoderli bir servo eksen, güç çevriminden sonra konumunu korur, bu nedenle her güç çevriminden sonra ana konuma getirmeye gerek yoktur.

3. MC_MOVEABSOLUTE:

Bu komut bloğu eksenı mutlak bir konuma taşımaktadır. Örneğin, servo eksenı 25 mm konumundayrsa ve komut Pozisyon girişi 50 mm'ye ayarlanmış olarak çıkarılırsa, eksen 50 mm konumuna hareket edip orada duracaktır.



Şekil 3.13. MC_MOVEABSOLUTE komut bloğu [Siemens, 2024].



Şekil 3.14. MC_MOVEABSOLUTE komutu çalışması [Siemens,2024].

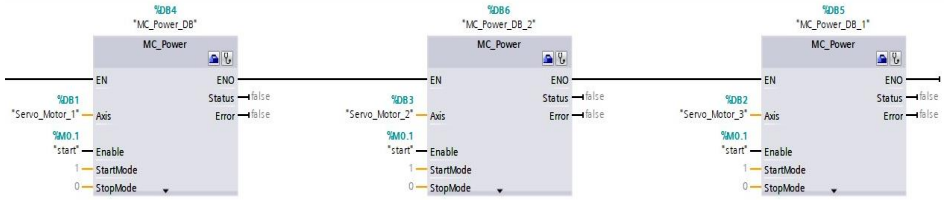
Şekil 3.14'de:

- ①: Axis_1, bir MC_MoveAbsolute görevi ile 50.0 hızı ile 1000.0 mutlak konumuna taşınır. Eksen hedef konuma ulaştığında, “Done_1” biti aracılığıyla bildirilir. “Done_1” = TRUE olduğunda, hedef konumu 1500.0 olan başka bir MC_MoveAbsolute görevi başlatılır. Eksen yeni hedef konuma ulaştığında, bu durum “Done_2” aracılığıyla bildirilir.
- ②: Bir aktif MC_MoveAbsolute görevi, başka bir MC_MoveAbsolute görevi tarafından iptal edilir. İptal, “Abort_1” aracılığıyla bildirilir. Axis daha sonra yeni hızda yeni hedef konum 1500.0'a hareket ettirilir. Yeni hedef konuma ulaşıldığında, bu durum “Done_2” aracılığıyla bildirilir.

3.3 PLC'de oluşturulan kontrol programı

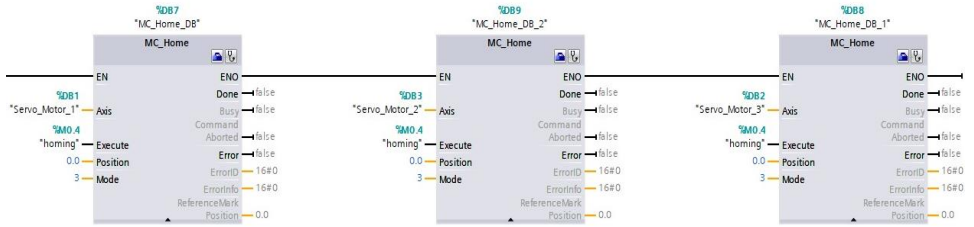
Delta robotun kontrolü S7-1200 PLC ile gerçekleştirilmiştir. Bu PLC TIA Portal'da Ladder Logic kullanılarak programlanmıştır. Program kodunda yer alan networkler aşağıdaki gibi açıklanmıştır:

Şekil 3.15'de gösterilen Networkta MC_POWER blokları eklenmiş ve StartMode “1” olarak ayarlanmıştır. Enable girişi TRUE olduğunda, üç servo eksen devreye girmektedir.



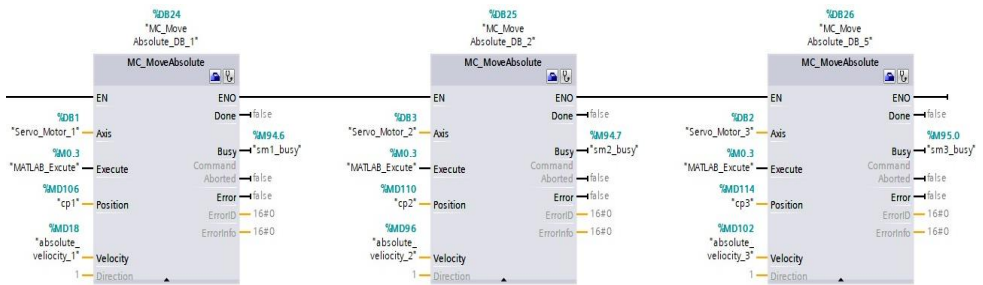
Şekil 3.15. Servo Axis'lerine Güç Verme.

Sıradaki network, şekil 3.16’te gösterildiği gibi, MC_home blokları homing yapmak için eklenmiştir. Bu çalışmada, Mode girişi “3” olarak ayarlanmıştır (MC_Home komut bölümünde açıklandığı gibi).



Şekil 3.16. Servo Axis'leri Home etme.

Şekil 3.17’de açıklanan networke MC_MOVEABSOLUTE blokları eklenmiştir. burada Execute girişi MATLAB’daki bilgisayar görüş sisteminden gelen geri bildirim verileriyle bağlantılıdır.

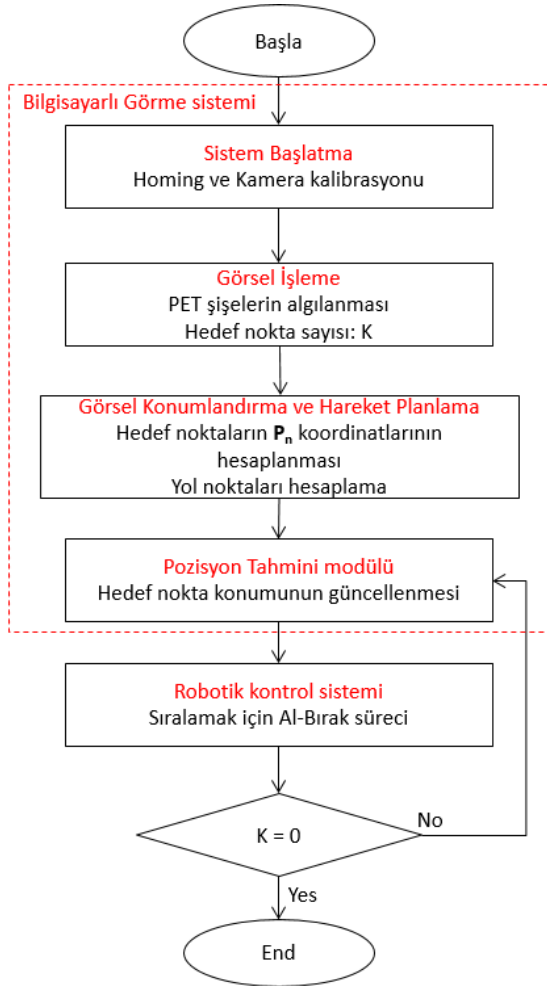


Şekil 3.17. Servo Axis'leri hareket ettirme.

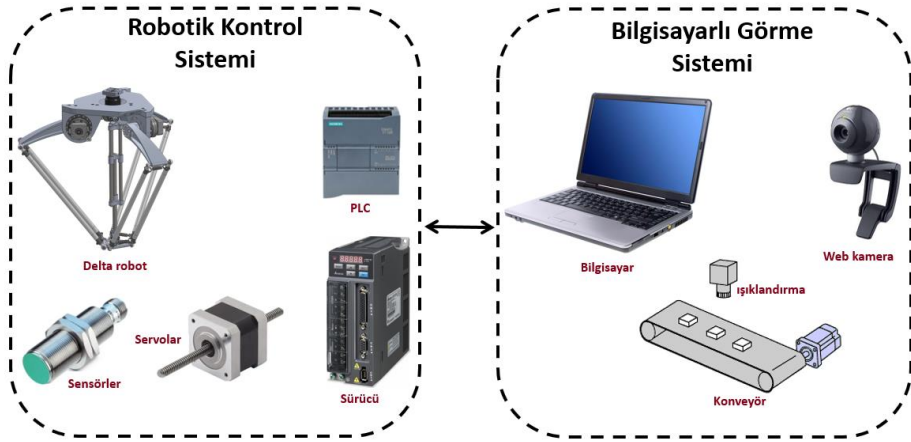
4. BİLGİSAYARLI GÖRÜ DESTEKLİ DELTA ROBOT UYGULAMASI

Paralel delta manipülatörün kinematik modeli oluşturulduktan sonra, bu sistemin bilgisayarlı görüş kısmı incelenmiştir. Bu bölümün temel amacı,

hareket planlamasında elde edilen sonuçları derin öğrenme tabanlı nesne algılama ve görüntü işleme algoritmalarıyla entegre etmektir. Konveyör üzerindeki hareketli nesnelerin eş zamanlı konum bilgilerini tahmin etmek için bir konum tahmin yöntemi (Şekil 4.1'de gösterildiği gibi) önerilmektedir, bu metot konveyör üzerindeki nesnelerin güncellenmiş konum bilgilerini sağlamak için kullanılmış ve böylece tespit edilen nesnenin anlık konumu belirlemek için kullanılmaktadır.



Şekil 4.1. Sistemin akış şeması.



Şekil 4.2. Tasarlanan robotik sistemin bileşenleri.

Şekil 4.1 ve 4.2'de gösterildiği gibi, genel sistem iki ana alt sisteme ayrılmaktadır:

Bilgisayarlı görü sistemi: dört modülden oluşmaktadır, sıralanacak gerekli nesnelere belirlemek için derin öğrenme ve Görüntü işleme algoritmaları ile nesne algılamayı kullanan bir **görsel işleme modülü**. Algılanan nesnelere bulmak ve konum koordinatlarını hesaplamak için bir **görsel konumlandırma modülü**. Kinematik model sonuçlarını kullanarak konum analizi yapmak için bir **hareket planlama modülü** ve hedef noktayı tahmin etmek ve konum bilgilerini güncellemek için bir **konum tahmin modülü**. Bu sistem robotik kontrol sistemi ile haberleşerek, mutlak yer değiştirmeyi hesaplar ve Servo motorların hedef nesnelere almak ve yerleştirmek için delta robotu hareket ettirmesini sağlamaktadır.

Robotik kontrol sistemi: Delta robotu kontrol etmek için PLC kullanılmaktadır. Bu şekilde robot, bilgisayarlı görü sistemi geri bildirimini kullanarak tanımlanan PET şişeleri ayırabilmektedir. Ardından robotun otomatik alma ve yerleştirme görevlerini gerçekleştirmek için nesnenin anlık konumuna hareket etmesini sağlamaktadır.

4.1 Bilgisayarlı görü sistemi

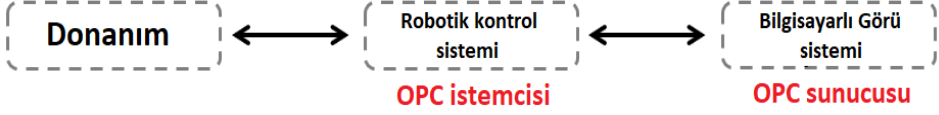
Kamera konveyör üzerindeki hareketli nesnelere yakaladığında, görüntü bilgisayarlı görü sisteminin görsel işleme modülü tarafından işlenmekte,

eğitilmiş YOLO nesne algılayıcısını kullanarak atık hattındaki diğer nesnelere arasında PET şişe sınıfı nesnelere tanımlanmaktadır. Nesne algılama deneysel sonuçları Şekil 4.3'de gösterilmektedir. Bundan sonra görsel konumlandırma modülü, tespit edilen nesnelere merkez noktasının koordinatlarını belirlemek için gerekli konum analizini yapmakta, ardından hareket planlama modülü global koordinatları hesaplamakta, konum tahmin modülü ise hareketli nesnenin gerçek zamanlı konumunu sağlamaktadır. Önerilen yaklaşıma göre konum bilgisi, konveyör bandının hızıyla aynı olan nesnelere hızına ve görsel işleme, konumlandırma ve hareket planlama modüllerinin kod yürütme süresine bağlı olarak güncellenmektedir.



Şekil 4.3. YOLO algoritması kullanılarak nesne algılama sonuçları.

Nesnenin konumlandırma bilgisini delta robot kontrol ünitesine iletmek için, oluşan ana sistemler arasında bir iletişim ara yüzü belirlenmektedir. Bu çalışmada TCP/IP protokolü üzerinden OPC kullanılmaktadır. Kontrol mimarisinde, Şekil 4.4'te gösterildiği gibi, bilgisayarlı görü sistemi sunucu (Server), robotik kontrol sistemi ise istemci (Client) rolündedir. Bilgisayarlı görü sisteminden gelen işaret, iletişimin başarısını garanti etmek için delta robot sisteminin iletişim protokolünü takip etmelidir.



Şekil 4.4. Bilgisayarlı görü ve robotik kontrol sistemleri haberleşme ara yüzü.

4.2 Bilgisayarlı görü sisteminin delta robot ile entegrasyonu

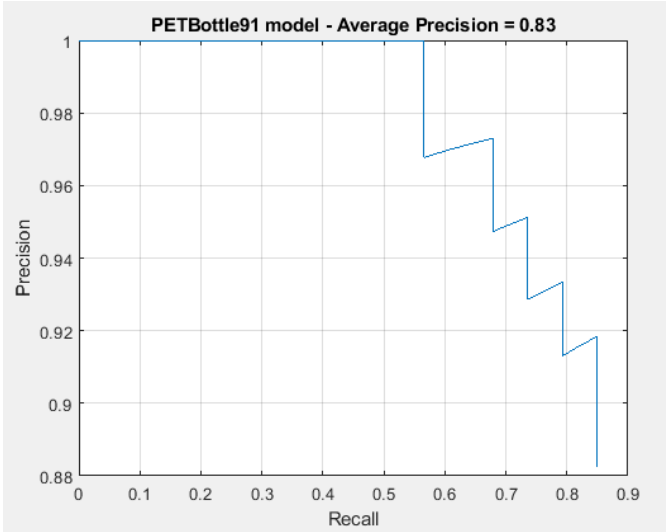
Bilgisayarlı görü sistemi, atık hattındaki PET şişeleri ayırmak üzere alma ve yerleştirme görevlerini yerine getirmek için bir delta robot ile entegre edilmiştir. Bu çalışmada, deltanın kelepçesi bir vakum emme vantuzu kullanılarak oluşturulmaktadır. İlk başta delta başlangıç noktasına ayarlanmakta, akış yolu başladığında, bilgisayarlı görü sistemi işlemeye başlamakta, nesne ekranda tam olarak görüldüğünde, görsel işleme ve konumlandırma modülü aracılığıyla merkez koordinatlarını hesaplamak için hareket planlaması yapılmakta, haberleşme ağları ile gerekli bilgileri delta robot kontrol ünitesine aktarmakta, PLC bilgileri aldıktan sonra, tespit edilen PET şişeleri diğer nesnelere göre işleme başlatmakta ve bunları sıralama alanına yerleştirmektedir. Çalışan sistemin sözde kodu Şekil 4.5'te verilmiştir. Şekil 4.6, bilgisayarlı görü sisteminin (Derin öğrenme tabanlı nesne algılama modeli) gerekli PET plastik şişeleri tespit etmek için ortalama hassasiyet sonuçlarını göstermektedir. Deneysel sonuçlar ortalama hatanın 0,83 olduğunu göstermektedir.

Algoritma Sistemin sözde kodu.

```
1: Başla
2: Sistemin başla
3: Homing etme
4: Kamera kalibrasyonu
5: while sonlandırma kriteri karşılanmazsa do
6:     Görüntü al
7:     Görüntü işle
8:     Nesne algıla
9:     Görsel konumlandırma
10:    K ← Hedef noktaları sayısı
11:    if K == 0
12:        continue
13:    end if
14:    for i ← 1 to K do
15:        Yol noktalarını planla
16:        Konum tahmin et
17:        Açı değerleri hesapla
18:        Hedef noktaların bilgilerini al
19:        Hareket_kontrolü()
20:    end for
21: end while

22: function Hareket_kontrolü ()
23: Sonraki hedef noktalarının bilgilerini oku
24: if Hedef nokta çalışma alanı içinde değil
25:     Adım 23'e git
26: end if
27: Hız değerlerini hesapla
28: Robot konumlandırma hesapla
29: PLC'nin data bloğuna yaz
30: end function
```

Şekil 4.5. Gerçekleştiren sistemin sözde kod.



Şekil 4.6. PET şişeleri algılamak için ortalama hassasiyet.

5. SONUÇ

Bu çalışma, incelenen bilgisayarlı görü destekli robotik sistemin plastik PET şişeleri bir atık hattından ayırtmak için kullanılabileceğini göstermektedir. Ayırtma işlemi, bir konveyör bant üzerinde hareket eden bir atık hattı için alma ve yerleştirme görevlerini yerine getirerek yapılmıştır. Görüntü işleme ve derin öğrenme tabanlı görüntü sınıflandırmasını gerçekleştirmek için MATLAB'da modüllerden oluşan bilgisayarlı görü sistemi geliştirilmiştir. Nesne algılama işlemleri için You Only Look Once (YOLO) yöntemleri üzerinde çalışılmıştır. Daha sonra, önerilen yöntemde dayalı olarak hedef konumlarını hesaplamak için hareket planlama ve konum tahmin yöntemleri kullanılmıştır. Son olarak, görsel işleme sonuçları, konum analizi ve hareket kontrolörü hareketleri, tespit edilen nesnelere otomatik olarak sıralamak için tek bir sisteme entegre edilmiştir.

Deneysel sonuçlar, incelenen sistemin PET şişeleri ayırtma amaçları için etkili olduğunu doğrulamaktadır. Ayrıca, paralel delta robotun performansı ile kullanılan nesne algılama tekniği arasındaki korelasyonun yüksek olduğu ve bu ikisinin birlikte iyi çalıştığını göstermektedir. Çalışılan sistemin etkinliğinin, kullanılan donanım, nesnelere karmaşıklığı, çalışma ortamının aydınlatılması ve kameranın kalitesi gibi bazı mekanik ve algılama hataları ile sınırlı olduğunu belirtmekte önemli bir husus vardır.

KAYNAKÇA

- [Clavel, 1991] Clavel, R., "Conception d'un robot parallèle rapide à 4 degrés de liberté," Ph.D. Thesis, EPFL, Lausanne, Switzerland, 1991.
- [Gasparetto, 2019], A. Gasparetto and L. Scalera, "From the Unimate to the Delta robot: the early decades of Industrial Robotics", Springer, DOI:10.1007/978-3-030-03538-9_23, 2019, <http://hdl.handle.net/11390/1142401> since 2020-06-04T15:13:57Z
- [Bonev, 2001] Bonev, I.: "The Delta parallel robot-the story of success"; (2001, May 6). Retrieved 28 December 2023, from <http://www.parallemic.org/Reviews/Review002.html>
- [Guglielmetti, 1994], Guglielmetti, Philippe, "Model-based control of fast parallel robots: a global approach in operational space", These No 1228, 1994
- [Nzue et al., 2013] Nzue, R. M. A., Brethé, J. F., Vasselin, E., Lefebvre, D.: "Comparison of Serial and Parallel Robot Repeatability Based on Different Performance Criteria" Mechanism and Machine Theory, 61, 2013, 136-155.
- [Erzincanli et al., 1997] Erzincanli, F., Sharp, J. M.: "Meeting the need for robotic handling of food products" Food Control, 8, 4, 1997, 185-190.
- [Li et al., 2005] Xu, Q., Li, Y.: "Dynamic analysis of a modified DELTA parallel robot for cardiopulmonary resuscitation" In 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2005, xxx-xxx.
- [Lin et al., 2016] Lin, C. J., Shaw, J., Tsou, P. C., Liu, C. C.: "Vision servo-based Delta robot to pick-and-place moving parts" In 2016 IEEE International Conference on Industrial Technology (ICIT), 2016, xxx-xxx.

- [Cong et al., 2023] Cong, V. D., Phuong, L. H.: “Design and development of a delta robot system to classify objects using image processing” In International Journal of Electrical and Computer Engineering (IJECE), 2023, xxx-xxx.
- [Schneider Electric, 2023] <https://www.se.com/za/en/product/VRKP4L0FNC00000/delta-3-robot-p4-35-axis-15-kg-permissible-load-01200-mm-working-envelop/>
- [Smirnova, 2023] Smirnova, Y.: “Delta-robot kinematic”, MATLAB Central File Exchange; Retrieved 29 December 2023, from <https://www.mathworks.com/matlabcentral/fileexchange/71119-delta-robot-kinematic>
- [Doğan, 2010] Doğan, E.: “Calculations of Delta Parallel Robots Forward and Inverse Kinematics” (Master's dissertations); Eskisehir Osmangazi University, Odunpazarı, Eskişehir, TR (2010).
- [Robert 2016], R.L. Williams II, “The Delta Parallel Robot: Kinematics Solutions”, Internet Publication, www.ohio.edu/people/williar4/html/pdf/DeltaKin.pdf, January 2016.
- [Delta, 2024] Online Documentation, Retrieved 17 August 2024, from <https://delta-ia-tips.com/2012/11/05/technical-announcement-pnp-connection-to-delta-servodrive/>
- [Siemens, 2024] Online Documentation, Retrieved 19 August 2024, from <https://support.industry.siemens.com/cs/mdm/107623221?c=73837491339&lc=en-AE>

BÖLÜM 4

YAPISAL DİRİLİŞ: BETONARME YAPILARDA KOLON ONARIM VE GÜÇLENDİRME YÖNTEMLERİ

Dr. Öğr. Üyesi Mahmut DURMAZ¹
(0000-0002-6060-4258)

Kübra KAYA²
(0009-0009-4065-0175)

DOI: <https://dx.doi.org/10.5281/zenodo.13825225>

¹ Siirt Üniversitesi, Mühendislik Fakültesi, İnşaat Mühendisliği Bölümü, Siirt, Türkiye, mahmutdurmaz@siirt.edu.tr

² Siirt Üniversitesi, Mühendislik Fakültesi, İnşaat Mühendisliği Bölümü Lisans Öğrencisi, Siirt, Türkiye, kbrakaya01344@gmail.com

1. GİRİŞ

Tarihin en eski çağlarından bu yana pek çok uygarlığa ev sahipliği yapmış olan Türkiye, konumu sebebiyle mühim fay hatları üzerinde yer aldığı için sürekli olarak deprem etkilerine maruz kalmıştır (Şirin, 2006). Türkiye bir deprem ülkesidir, bu durum bilindiği halde depremin olumsuz etkilerine karşı gerekli önlemler alınmadığı için, meydana gelen depremler sonucunda da çok sayıda mal ve can kayıpları meydana gelmiştir (Demirkan, 2014). Deprem sonrası yıkılan veya hasar gören her yapı da aslında bizlere standartlara uyulmadığını, yapının imalatında titizlikle davranılmadığını da göstermektedir (Akbaş & Çalışkan, 2023; Çırak, 2011). Bizler bu durumun önüne geçebilmek için depreme dayanıklı yapılar inşa etmek ve yapılarımızı gerektiği gibi güçlendirmek ile sorumluyuzdur. Günümüzde de şehirleşme ve yapılaşmanın hızla artması ve bununla beraber olarak yapı stoklarının yaşlanmasıyla, betonarme yapıların bakım ve onarımı mühendislik alanında giderek daha büyük bir önem kazanmaktadır (Atay, 2010). Bu bağlamda, yapıların taşıyıcı elemanlarından biri olan kolonlar, özellikle depremler, aşınma, kimyasal etkiler ve yapısal yorgunluk gibi faktörlere maruz kaldıkları için belirli bir dönemden sonra hasar görebilmektedir. Bu hasarlar, yapı güvenliği açısından ciddi riskler oluşturabilir ve zaman içinde artarak yapıyı daha da zayıflatabilmektedir (Keskin, 2005). Özellikle büyük şehirlerde, betonarme kolonların güvenilirliği ve dayanıklılığı önemli bir mesele haline gelmiştir. Betonarme kolonlardaki hasarlar, sadece yapısal bütünlüklerini etkilemekle kalmaz, aynı zamanda çevresel ve ekonomik etkileri de beraberinde getirmektedir. Bu bağlamda, hasarlı kolonları onarmak ve güçlendirmek, sadece mevcut binaların ömrünü uzatmakla kalmaz, aynı zamanda gelecekteki yapı projelerinin sürdürülebilirliğini artırabilmektedir (Demirkan, 2014).

Bu makalede, betonarme yapılarda karşılaşılan kolon hasarlarında uygulanması gereken onarım ve güçlendirme yöntemleri üzerinde durulacaktır. İlk olarak onarım ve güçlendirme terimlerinin ayrımından bahsedecek olursak; onarım hasar görmüş bir yapıyı veya yapı elemanını önceki durumuna geri döndürmek için (sağlamlık, dayanıklılık, mukavemeti artırmak vb.) yapılan işlemlerdir, güçlendirme ise bir yapının yük taşıma kapasitesini, sağlamlığını, dayanıklılığını veya stabilitesini önceki durumunun üstüne çıkarmak için yapılan değişikliklerdir (Özcan, 2018; Yiğit, 2002). Mevcut literatürdeki araştırmaların ve güncel gelişmelerin bir derlemesi yapılarak, bu alandaki güçlü

noktalar belirlenmiştir. Ardından, çeşitli güçlendirme yöntemleri, lifli polimer sargılar, betonarme mantolamalar, çelik mantolamalar, epoksi enjeksiyonu gibi onarım ve güçlendirmeler detaylı bir şekilde incelenmiş olup bu tekniklerin kapsamı hakkında gerekli bilgiler edinilmiştir.

Bu kullanılmakta olan yöntemler hem mevcut yapıların ömrünü uzatmayı hem de deprem gibi doğal felaketlere karşı direncini artırmayı amaçlamaktadır. Kolonların doğru bir şekilde onarılmaması veya güçlendirilmemesi, sadece bina sakinleri için değil, aynı zamanda çevrelerindeki topluluklar için de ciddi sonuçlar doğurabilmektedir. Bu nedenle, bu konuda bilinçli ve etkili bir yaklaşım benimsemek, yapıların dayanıklılığını artırmak ve gelecekte olası hasarlara karşı korunmak için önemli bir adımdır (Demirkan, 2014). Bu çalışmanın, mühendislik topluluğu ve inşaat sektörü profesyonelleri için önemli bir kaynak oluşturmasını amaçlamaktayız. Betonarme yapılardaki kolon onarımı ve güçlendirmesi, yapı güvenliği ve sürdürülebilirlik açısından kritik bir konu olup, bu derleme makalede sunulan bilgilerin sektördeki uygulayıcılara kılavuzluk etmesi de amaçlanmıştır.

2. MATERYAL VE METOD

Beton; iri agregası, ince agregası, çimento ve suyun, kimyasal ve mineral katkılarının yönetmeliğe uygun oranlarda eklenerek veya bu katkı maddeleri eklenmeden homojen olarak karıştırılmasıyla oluşturulan ve çimentonun hidrate olmasıyla gereken özelliklerin elde edildiği bir malzemedir. Türkiye Bina Deprem Yönetmeliği 2018'e göre yapılacak tüm betonarme yapılarda C25/30 betonundan daha düşük dayanımlı beton kullanımına izin verilmemektedir. Deprem bölgelerinde ise C30/37 beton sınıf düzeyine çıkılması ve bu sınıftan daha düşük betonların kullanılmasına izin verilmemesi gerekmektedir (TS EN 206; TBDY-2018; THBB, 2022).

Çelik; betonarme yapılarda kullanılmakta olan çekme gerilmesine karşın dayanıklı bir malzemedir. Betonarme yapılarda kullanılacak çeliğin TBDY-2018'e göre sadece B420C ve B500C çeliklerinin kullanılmasına doğrudan izin verirken S420 çeliğinin kullanımına koşullu izin vermektedir. Betonarme ise, çelik ile güçlendirilen beton anlamına gelmekte olan kompozit bir yapı malzemesidir. Çelik çekme gerilmelerine, beton ise yüksek basınç gerilmelerine karşı dayanıklıdır. Bu sebeple yapılarımızda betonarme tercih

edilmesinin sebebi, betonun basınç gerilmelerini; çeliğin ise çekme gerilmelerini karşılamasıdır (TS 708; TS 500; Meral, 2019).

Lifli malzemeler; çapları mikron seviyesinde olan ve boyca uzun olan bir tür kompozit malzemelerdir. Bu malzemeler, polimer matris içinde güçlendirici elyaflar içerir ve genellikle dayanıklılık, mukavemet ve hafiflik gibi özellikleri artırmak amacıyla kullanılır. Lifli kompozit malzeme, genellikle lif ve matris olmak üzere, en az iki malzemeden oluşmaktadır (Polat, 2014; Atay, 2010).

Epoksi reçinesi; iki bileşenli bir polimer sistemidir ve genellikle bir reçine (epoksi reçinesi) ve bir sertleştirici (sertleştirici veya kür) içerir. Bu iki bileşen, karıştırıldığında kimyasal reaksiyonla birleşir ve sert, dayanıklı bir malzeme oluşturur. Epoksi reçineleri genellikle çeşitli uygulamalarda kullanılır (Kan, 2020).

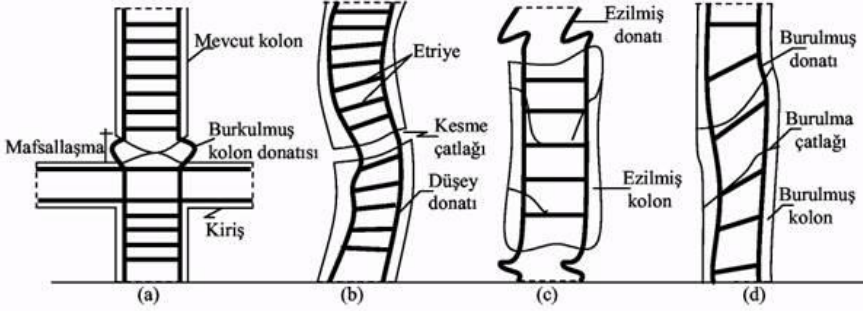
Çelik şeritler; 1,0-1,5 mm ebatlarında olan paslanmaz malzemeden meydana gelen malzemelerdir. Çelik güçlendirme öncesi beton yüzeyi zımpara ile düzeltilmektedir. Kalıntılar olmaması için yıkanıp kuruması beklenir. Uygulama sırasında kullanılan epoksi reçinesi çelik şeridin ve betonun yapışacak yüzeylerine sürülmektedir ve çelik şeritlerin yapışma işleminden sonra 24 saat boyunca basınç uygulama suretiyle yapışması sağlanır. Çelik şeritlerin yapışmasında püskürtme beton yöntemi de kullanılır (Kan, 2020; Demirkan, 2014).

Karbon fiberin; hafif yapıda olması, korozyon tetiklenmesi olmaması ayrıca büyük ebatlarda olması artı özellikleridir. Sürekli yük binmesi sonucu başlangıç mukavemetinde %15 ile %60'a varan oranda azalma olabiliyor. Bundan dolayı depreme karşı ve daimî olmayan yük güçlendirilmesinde yararlıdır. Kolon güçlendirme malzemeleri arasında en hafifi karbon fiberdir. Çelikten 4, 5 kat daha hafiftir. Bu da karbon fiberi 3 kat daha dayanıklı yapmaktadır (Keskin, 2005; Atay, 2010).

Bu çalışmada, betonarme yapılarda birçok etkiden dolayı meydana gelmekte olan kolon hasarlarına uygulanan onarım ve güçlendirme yöntemleri hakkında literatür taraması yapılmıştır ve literatürde yer almış özellikle son yıllara ait kaynaklar incelenmiştir. Güncel bilgiler incelendikten sonra sentezi yapılarak gerekli sonuçlar elde edilmiştir.

3. BETONARME YAPILARDA KOLON ONARIMI VE GÜÇLENDİRME YÖNTEMLERİ

Betonarme kolonlarda birçok etkene bağlı olarak çeşitli hasarlar meydana gelmektedir. Şekil 1’de gösterildiği üzere kolonlarda dört çeşit hasar türü vardır. Bunlar; kesme hasarı (Şekil 1b), basınç hasarı (Şekil 1c), eğilme hasarı (Şekil 1a) ve burulma hasarıdır (Şekil 1d). Hasar görmüş bir kolonun deprem etkilerini taşıyabilir duruma getirilmesi veya yatay yük taşıma kapasitesinin artırılması için kolonların onarılması ve/veya güçlendirilmesi gerekmektedir (Demirkan, 2014; Kan, 2020). Kolonlarda da hasar durumuna bağlı olarak çeşitli güçlendirme ve onarım yöntemleri bulunmaktadır (Yiğit, 2002).



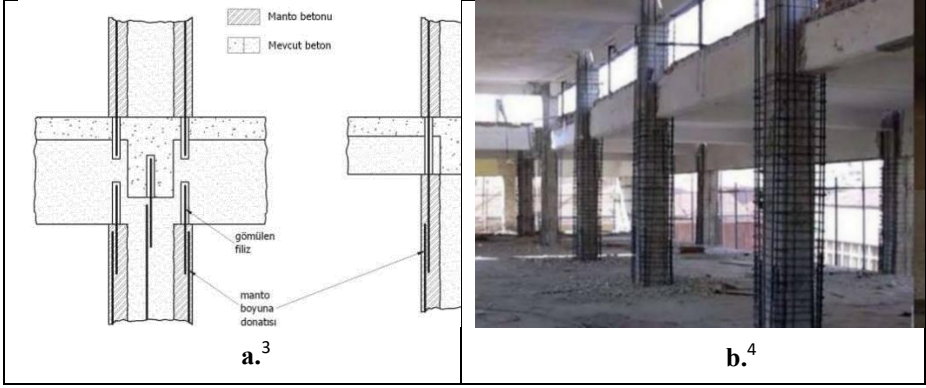
Şekil 1 a. Eğilme hasarı **b.** Kesme hasarı **c.** Basınç hasarı **d.** Burulma hasarı (Aytaç, 2011)

Kolonlardaki 0,1-5 mm genişliğindeki çatlaklar epoksi reçinesi ile onarılabilmektedir. Bunun için 20-100 cm mesafelerde açılan deliklerde aşağıdan yukarıya doğru epoksi reçinesi enjekte edilir. Çok daha büyük çatlaklar için çimento şerbeti enjeksiyonu uygun olabilir (Öztürk, 1998; Celep & Kumbasar, 2000). Ancak onarım veya güçlendirilecek olan bölgenin test edilerek kontrol ve doğrulanması gerekir. Kolonda yerel beton kırılması varsa bu alanın temizlenerek pürüzlülüğe göre etriye veya boyuna donatı ilavesi tavsiye edilir. Beton dökülmeden önce mevcut yüzeyin suya doyurulması eski ve yeni betonun bütünleşmesini sağlar. Bu amaçla özel kalıpların kullanılması, kesitin şişirilip beton dökülmesi veya tamir harcı kullanılması uygundur. Beton sertleştikten sonra fazlalık olan kısımlar kazınır (Keskin, 2005).

Betonarme kolonların güçlendirilmesinde en etkili ve yaygın yöntem ise mantolamadır. Bu yöntem kolon kesit alanını artıran bir kaplama yöntemidir. Bunun haricinde var olan hasarın durumuna göre çelik profil iskelet geçirme, çelik levha ile kaplama ve cam karbon lifli dokumalar ile de güçlendirme yöntemleri mevcuttur (Şirin, 2006).

3.1 Betonarme Mantolama

Kolonda, hasarın fazla olması veya mukavemetinin yetersiz olduğu gibi durumlarda genellikle mantolama teknikleri tercih edilmektedir. Betonarme kolonların onarımı/güçlendirilmesinde yaygın olarak kullanılan bir yöntemdir (Şirin, 2006). Kolon taşıma gücünün artırılması için yaygın olarak kullanılan bu yöntemde, mevcut kolona beton kesiti ve donatı eklemesi yapılır. Mantolama, mevcut sütunlara beton kesitler ve inşaat demiri eklenmesini içerir. Mevcut kolon ve yeni kolonlar arasında entegrasyonu sağlamak için mantonun en az 100 mm kalınlığında olması gerekir. Kolonlar uygulamaya bağlı olarak bir, iki, üç veya dört taraftan kaplanıp mantolanabilir. Ancak tüm çevreyi kaplayan bir mantolama yönteminin kullanılması tercih edilmelidir (Bayülke, 1995). Bu şekilde mevcut beton ve yeni beton arasında tam kuvvet aktarımı sağlanabilir. Dört kenarın mantolanması durumunda, mevcut kolonun boyuna donatısının ve etriyesinin ortaya çıkartılarak, yeni etriyelerin doğrudan veya bağ parçası ile bağlanıp kaynaklanması gerekmektedir. Dört taraflı mantolama durumunda genellikle yüzey pürüzlendirmesi yeterli olur ancak büyük etkiler var ise beton örtü tabakasını kaldırmak çok daha uygundur. Tek taraflı, çift taraflı veya üç taraflı mantolamada mevcut kolon ile manto kısmının entegre edilmesine özen gösterilmesi gerekir. Bu amaçla L şeklinde kenetlenme takviye donatıları kullanılabilir (Keskin, 2005; Yiğit, 2002). Mantolamanın asıl amacı kolona gelen düşey yük taşıma kapasitesini artırıp dikey yüklere karşı güvenlik payını yükseltmektir (Atay, 2010; Demirkan, 2014). Şekil 2a ve şekil 2b'de kolonda yapılmakta olan betonarme manto işlemleri görülmektedir.



Şekil 2 a. Kolonda betonarme mantolama b. uygulanan mantolama örneği

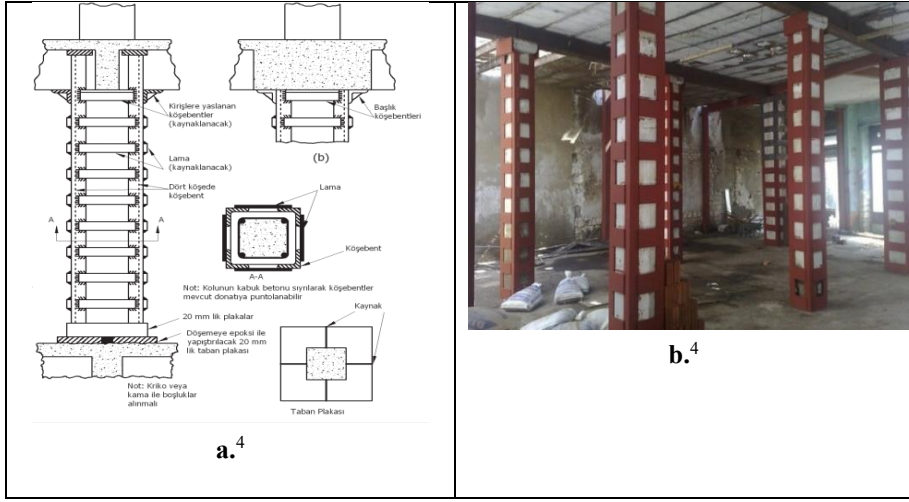
3.2 Çelik Mantolama

Yapıların çelik ile mantolanması, elemanlar ve katlar arasındaki moment aktarımından ziyade aksel yük kapasitesini artırmaya yönelik kullanılmakta olan bir yöntemdir. Çelik sargı, dikdörtgen kesitli betonarme bir kolonun köşelerine dört adet boyuna köşebent yerleştirilip ve bunların belirtilen aralıklar ile yatay plakalara kaynaklanmasıyla oluşturulmaktadır. Köşebentler ile betonarme yüzey arasında boşluk olmamalıdır (Tosun, 2009; Demirkan, 2014). Yatay plaklar dört taraftan sürekli olmalıdır. Çelik sargı, kolonun aksel yük kapasitesini artırabilmesi için korniyerlerin alt ve üst katlar arasında sürekli olması (boşlukların giderilmesi gerekir) ve döşemelere başlık plakaları ile basınç aktarımı sağlanmalıdır (Hakan, 2020). Basınç iletiminin sağlandığı matematiksel hesaplamalarla gösterilmelidir. Gerektiğinde köşebentlere ön yükleme yapılmı halinde mevcut betonarme kolon kesitlerindeki düşey yüklerden kaynaklanan aksel basınç yükü azaltılabilir (Gülmez, 2010). Şekil 3a'da kolonların çelik manto ile güçlendirilmesi ve şekil 3b'de ise çelik mantonun saha uygulaması görülmektedir.

3

https://eski.imo.org.tr/resimler/dosya_ekler/0f124ec125d030b_ek.pdf?tipi=1&turu=X&sube=10

⁴ <https://www.sanalsantiye.com/kolon-guclendirme-mantolama-yontemleri-nelerdir/>



Şekil 3 a. Kolonun çelik manto ile sarılması b. çelik mantolamanın saha uygulaması

Kolonun çelik ile mantolama uygulamasında; köşebentler aksenal yük kapasitesini artırır. Yatay lamalar dış tarafta sargı etkisi sağlar ve korniyerlerin burkulmasını önler. Bu uygulama sırasında bazı ayrıntılara dikkat edilmelidir. Bunlar (Demirkan, 2014);

- Korniyerler işkence ile kolona iyice oturmalıdır.
- Korniyerler üstte ve altta rijit çelik plakalara dayandırılmalı ve sıkıca sabitlemelidir.
- Bu amaçla kolonun her iki yanına simetrik olarak yerleştirilmiş, ortak yağ pompasına bağlı olan iki kriko kullanılabilir.
- Krikoyla kaldıramıyorsanız, altta bulunan plakanın altına çakılan kamalarla sıkıştırma yapılabilir.
- Korniyerler yeterince sıkıştırılmazsa çelik manto istenilen yükü taşıyamaz.

3.3 Lifli Polimer (LP) Sargı

LP tabakası kolonun çevresine, lifler enine donatıya paralel olacak şekilde sarılıp ve yapıştırılması ile sargılama işlemi yapılır. Lifli polimer sargısı, boyuna donatıların bindirme boyu yetersiz olduğunda betonarme kolonların süneklik kapasitesini, kesme dayanımını ve basınç dayanımının donatı kenetlenme dayanımını artırır. Lifli polimer sargılarla yapılan güçlendirmelerde tam sargı (kesit çevresinin tamamının sarılması) yöntemi kullanılmalı ve sargının sonunda en az 20 cm bindirme yapılmalıdır(Şekil 4a &

Şekil 4b). Lifli polimer (LP) sargısı dikdörtgen kolonlara kolonun köşeleri en az 3 cm yarıçapında yuvarlatılarak uygulanır. LP uygulaması üretici firmanın önerdiği yöntemlere uygun olarak yapılmalıdır (Demirkan, 2014; Gülmez, 2010).



Şekil 4 a. Kolonun lifli polimer (LP) ile sarılması b. LP örneği⁵

Bu teknik, genellikle betonarme kolonların mukavemetini ve dayanıklılığını artırmak amacıyla uygulanır. Lifli polimerler, genellikle karbon fiber, cam fiber veya bazen bazalt fiberleri gibi güçlü ve hafif malzemelerden yapılır.

İşte kolonun lifli polimer ile sarılmasının bazı temel avantajları:⁶

- **Mukavemet Artışı:** Lifli polimer sargı, betonarme kolonun mukavemetini artırabilir. Bu, yapı elemanının dayanıklılığını ve taşıma kapasitesini geliştirebilir.
- **Yorgunluk Direnci:** Lifli polimer sargı, kolonun yorgunluk direncini artırabilir. Özellikle depremler gibi dinamik yüklemelere karşı yapısal direnci artırabilir.
- **Korozyon Direnci:** Lifli polimerler genellikle metal olmayan malzemelerdir, bu nedenle metal donatıdaki korozyon riskini azaltabilirler.
- **Hafiflik:** Lifli polimerler genellikle hafif malzemelerdir, bu da yapı elemanının toplam ağırlığını azaltabilir ve taşıma sistemine yük getirebilir.

⁵ <https://www.besoglu.com/lifli-polimer-guclendirme-nedir-nasil-yapilir/>

⁶ <https://www.karadenizteknikproje.com/lifli-polimer-guclendirme-sistemlerinin-avantajlari/>

- Kolay Uygulanabilirlik: Lifli polimer sargılar genellikle kolayca uygulanabilir ve esneklik sağlar, böylece çeşitli geometrilerdeki kolonlara adapte edilebilir.

Ancak, lifli polimer sargılarının kullanımıyla ilgili bazı zorluklar da bulunabilir. Örneğin, doğru şekilde tasarlanması ve uygulanması önemlidir, aksi takdirde beklenen avantajlar elde edilemeyebilir. Ayrıca, maliyet ve malzeme uyumluluğu gibi faktörler de göz önünde bulundurulmalıdır. Bu nedenle, bu tür bir uygulama planlanırken dikkatlice mühendislik hesaplamaları ve tasarımlar yapılmalıdır.

4. BULGULAR

Betonarme yapılar yüksek dayanıklılıkları ve uzun servis ömürleri nedeniyle yaygın olarak kullanılmaktadır. Ancak, çeşitli faktörlerin etkisi altında kolonlarda meydana gelen hasarlar, yapının bütünlüğünü önemli ölçüde tehlikeye atabilmektedir ve meydana gelen hasarlar ile baş edebilmek adına birçok yöntem vardır (Kan, 2020). Bu çalışmamızda, betonarme kolonlarda uygulanan çeşitli onarım ve güçlendirme teknikleri hakkında literatür araştırmaları sonucu ile gerekli bilgileri derlemek amacıyla incelemeler gerçekleştirilmiştir (Demirkan, 2014). Yapılan literatür taramalarında, betonarme kolonların dayanıklılığını ve mukavemetini artırmak için kullanılan çeşitli yöntemlerin var olduğunu ortaya koymaktadır. Yapılan literatür araştırmalarından edinilen bilgilerde lifli polimer sargılar kullanılarak yapılan güçlendirmelerde, kolonların taşıma kapasitesinde belirgin bir artışın elde edilebileceğini göstermiştir (Aytaç, 2011). Karbon fiber sargılar, özellikle çekme mukavemetini artırarak, kolonun genel performansını iyileştirici etkidedirler. Yapılan simülasyonlar, lifli polimer sargıların deprem yüklemelerine karşı dayanıklılığı artırdığını ortaya koymuş ve sismik etkiler altında, sargıların kolonda meydana gelen deformasyonları kontrol altında tuttuğu ve çatlama minimumda tuttuğu gözlemlenmiştir (Çopur, 2013). Betonarme veya çelik ile yapılan mantolama yöntemlerinde, kolonların aksenal yükler altında mukavemetini arttırıcı etkisi gözlenmiştir. Mantolama işlemlerinin kolonun taşıma kapasitesini iyileştirdiği ve deformasyonları kontrol altında tuttuğu da yapılan araştırmalar sonucu elde edilen bilgilerdendir (Bayülke, 1995). Eğilme momentleri altında çelik plakalarda, kolonun eğilme direncini arttırdığı belirtilmiştir. Bu yöntem, yapı elemanının deprem ve diğer

dinamik yüklemelere karşı daha dayanıklı hale gelmesini sağlamaktadır (Kan, 2020). Epoksi enjeksiyonu, betonarme kolonlardaki çatlakların etkin bir şekilde onarılmasını sağlayan bir yöntemdir ve çeşitli çatlak genişlikleri üzerinde yapılmış olan epoksi enjeksiyonunun çatlak genişliklerini önemli ölçüde azalttığını göstermektedir. Sıkıştırılmış beton uygulamaları da, betonarme kolonların mukavemetini yeniden kazanmasına yardımcı olmuştur. Betonun homojenliğini artıran bu onarım yöntemi, yapı elemanının dayanıklılığını da önemli ölçüde artırmaktadır. Epoksi ve bu türden olan diğer sıvı reçineler (epoksi, poliester, poliüretan, akrilik vb.) uygulama için genel anlamda kullanılır (Atay, 2010). Reçineyi oluşturan kimyasal birleşenlerin çeşitlerine ve kimyasal yapısına, kullanılma oranına, eklenen maddelere vb. bağlı olarak pek çok çeşidi bulunmaktadır. Bu nedenle onarım üzerinde olumlu etki yaratabilmek için amaca uygun reçinenin seçilmesi önemlidir. Epoksi enjeksiyonu binalarda, köprülerde, barajlarda ve diğer betonarme yapılarda meydana gelen çatlakların onarılması için tercih edilen çok etkili bir yöntemdir. Ancak kolonda meydana gelen çatlak oluşumları için yapısal herhangi bir önlem alınmamış ise kolonlarda çatlaklar tekrar oluşabilmektedir (Keskin, 2005).

5. SONUÇ VE ÖNERİLER

Bu çalışma, betonarme kolonlarda uygulanan çeşitli onarım ve güçlendirme tekniklerinin yapılan literatür taraması sonucuyla bilgiler edinmeyi amaçlamıştır. Yapılan taramalar sonucu betonarme kolonlardaki mevcut yapısal sorunların çözümüne yönelik olarak kullanılan onarım ve güçlendirme yöntemlerin, kolonların dayanıklılığını ve mukavemetini artırmak için kullanılan farklı yöntemlerin etkili olduğunu ortaya koymaktadır. Betonarme yapıların temel elemanı olan kolonların onarım ve güçlendirilme sürecinde özverili ve dikkatli davranılmalıdır.

- Karbon ve cam fiber sargılar, güçlendirme amaçlı uygulamalarda kolonların taşıma kapasitesini belirgin bir şekilde artırdığı gözlemlenmiştir.
- Lifli polimer sargılar, deprem yüklemelerine karşı yapıda dayanıklılığı önemli ölçüde artırmış ve çatlak oluşumunu azaltmıştır.
- Çelik plakaların kullanımı, kolonların eksenel yükler ve eğilme momentleri altında mukavemetini arttırmaktadır. Bu plakalar,

kolonun dayanıklılığını ve deformasyon kontrolünü artırarak, özellikle dinamik yüklemelere karşı direncini de arttırmıştır.

- Mantolama yöntemi, eğilme momentlerine karşı dirençte kolona belirgin bir kazanım sağlanmıştır.
- Epoksi enjeksiyonu, betonarme kolonlardaki çatlakları etkili bir şekilde onarmaktadır ve çatlak genişliklerini azaltmak için etkili bir yöntemdir.
- Bu tarz güçlendirme işlemleri yapıldığında gerekli tekniklere dikkat edilmeli ve özenli çalışılmalıdır.

Yapılan bu çalışma, çeşitli güçlendirme ve onarım tekniklerinin, betonarme kolonların dayanıklılığını ve taşıma kapasitesini artırmada etkili olduğunu göstermiştir. Yapılan onarım ve güçlendirme yöntemleri, kolonların deprem ve diğer dış etkenlere karşı daha dirençli hale getirilmesini sağlamıştır.

Sonuç olarak, bu çalışmanın bulguları, betonarme kolonlarda meydana gelen hasarların onarımı ve güçlendirilmesi için kullanılan çeşitli tekniklerin yapı elemanlarının performansını önemli ölçüde iyileştirebileceğini göstermektedir. Ancak, her bir teknik seçilirken dikkatlice yapı mühendisliği prensipleri ve yerel koşullar göz önünde bulundurulmalıdır. Gelecekteki çalışmalar, bu tekniklerin uzun vadeli dayanıklılık, ekonomik etki ve çevresel sürdürülebilirlik açılarından daha fazla incelenmesine odaklanabilir.

Yapılmış olan literatür taramaları sonucu elde edilen bu bilgiler, inşaat sektöründe mühendislik uygulamalarını geliştirmek ve yapısal dayanıklılığı artırmak için önemli bir katkı sağlayabilir. Gelecekteki çalışmalarda, farklı materyallerin ve yöntemlerin kombinasyonları üzerine daha fazla araştırma yapılması, betonarme yapı elemanlarının daha etkili bir şekilde güçlendirilmesi için yeni stratejilerin geliştirilmesine olanak tanıyabilmektedir.

KAYNAKÇA

- Akbaş, A., & Çalışkan, Ö. (2023). Deprem Etkisinde Hasar Alan Betonarme Yapıların Düzensizlik Türleri Yönü İle İncelenmesi. *In International Conference on Scientific and Academic Research*(1), s. 428-435.
- Atay, H. (2010). *Depremde Hasar Görmüş Yapıların Güçlendirme Yöntemleri ve Güçlendirmede Kullanılan Malzemeler*. Yüksek Lisans Tezi, Sakarya Üniversitesi.
- Aytaç, E. (2011). *CFRP Güçlendirme Malzemesi ve Güçlendirme Teknikleri*. Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi.
- Bayülke, N. (1995). Depremlerde Hasar Gören Yapıların Onarım ve Güçlendirilmesi. *İnşaat Mühendisleri Odası İzmir Şubesi*.
- Celep, Z. ve Kumbasar, N., (2000). Deprem Mühendisliğine Giriş ve Depreme Dayanıklı Yapı Tasarımı İkinci Baskı, Beta Dağıtım, İstanbul.
- Çırak, İ. F. (2011). Damages Observed In Reinforced Concrete Buildings, Causes And Recommendations. *International Journal of Technological Sciences*, 3(3), s. 62-71.
- Çopur, A. (2013). *Lifli Polimer İle Sargılanmış Daire En Kesitli Ultra-yüksek Dayanımlı Beton Kolonların Davranışı* (Doctoral dissertation, Fen Bilimleri Enstitüsü).
- Demirkan, D. (2014). *Betonarme Yapılarda Onarım ve Güçlendirme Teknikleri*. Yüksek Lisans Tezi, Aksaray Üniversitesi Fen Bilimleri Enstitüsü.
- Gülmez, Ö., (2010). Depremde hasar gören betonarme yapıların güçlendirilmesi ve mimariye olan etkisi, Yüksek Lisans Tezi, Haliç Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul
- Hakan, S., (2020). Mevcut betonarme bir binanın sistem iyileştirilmesi ve taşıyıcı eleman güçlendirme yöntemlerine göre karşılaştırılması, K.D.Ü. Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi.
- Kan, A. (2020). *Yapı Onarım ve Güçlendirme*. Ders Notu, Atatürk Üniversitesi, Erzurum.

- Keskin, E. (2005). *Betonarme yapıların onarımı, güçlendirilmesi ve lifle güçlendirilmiş polimerler*. Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- Meral, E. (2019). Betonarme Binalarda Kısa Kolon Etkilerinin Araştırılması. *International Journal of Engineering Research and Development*, 11(2), s. 515-527.
- Özcan, Z. (2018). *Hasar Belirleme, Onarım ve Güçlendirme Ders Notları*. Sakarya Üniversitesi Mühendislik Fakültesi.
- Öztürk, H., (1998). Betonarme yapılarda onarım ve güçlendirme teknikleri ve çeşitli tekniklerle onarılp güçlendirilen betonarme kirişlerin davranışlarının karşılaştırılması olarak incelenmesi, Yüksek Lisans Tezi, K.T.Ü. Fen Bilimleri Enstitüsü, İstanbul.
- Polat, K. (2014). *Lifli polimerle güçlendirilen betonarme yapı elemanlarının kapasite hesabı için analitik yöntemler* (Doctoral dissertation, Bursa Uludag University (Turkey)).
- Şirin, C. (2006). *Yapılarda oluşan hasar biçimleri ve nedenleri ve yapıların onarım ve güçlendirilmesi teknikleri İle bir yapının güçlendirilmesi*. Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- Tosun, M., (2009). Perde duvarlı betonarme karkas binanın yapımı sırasında malzemelerinin mekanik özelliklerinin betonun dayanımına etkisi, kat perdesi deprem süresince hasar oluşmuş perde karkas şemalı bir binanın depreme dayanıklılığının ve güçlendirme metotlarının incelenmesi, Yüksek Lisans Tezi, Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Kayseri.
- THBB. (2022). *Hazır Beton Sektör Raporu*. İstanbul: Türkiye Hazır Beton Birliği.
- TBDY-2018. “Türkiye Bina Deprem Yönetmeliği,” Ankara. 2018.
- TS 500. “Betonarme Yapıların Tasarım ve Yapım Kuralları,” Ankara. 2000.
- TS EN 206. “Beton-Özellik, Performans, İmalat ve Uygunluk,” Ankara. 2002.

TS 708. ‘‘Çelik – Betonarme İçin – Donatı Çeliđi,’’ Ankara. 2016.

Yiđit, Y. (2002). *Betonarme Yapılarda Onarım ve Güçlendirme Yöntemleri ve Bir Güçlendirme Uygulaması* . Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü.

https://eski.imo.org.tr/resimler/dosya_ekler/0f124ec125d030b_ek.pdf?tipi=1&turu=X&sube=10

<https://tr.wikipedia.org/wiki/Betonarme>

<https://www.besoglu.com/lifli-polimer-guclendirme-nedir-nasil-yapilir/>

<https://www.karadenizteknikproje.com/lifli-polimer-guclendirme-sistemlerinin-avantajlari/>

<https://www.sanalsantiye.com/kolon-guclendirme-mantolama-yontemleri-nelerdir/>

<https://www.senmetal.com.tr/blog/kolon-guclendirme-malzemeleri/>

BÖLÜM 5

BETONARME YAPILARDA DÖŞEMELERDE DEPREM HASARLARINI ONARIM VE GÜÇLENDİRME YÖNTEMLERİ

Dr. Öğr. Üyesi Mahmut DURMAZ¹
(0000-0002-6060-4258)

Sefa KENCO²
(0009-0002-2933-0292)

DOI: <https://dx.doi.org/10.5281/zenodo.13825231>

¹ Siirt Üniversitesi, Mühendislik Fakültesi, İnşaat Mühendisliği Bölümü, Siirt, Türkiye, mahmutdurmaz@siirt.edu.tr

² Siirt Üniversitesi, Mühendislik Fakültesi, İnşaat Mühendisliği Bölümü Lisans Öğrencisi, Siirt, Türkiye, sefakenco@gmail.com

1. GİRİŞ

Günümüzde betonarme yapılar dünya çapında yaygın olarak kullanılmaktadır. Pek çok insanın yaşamını geçirdiği bu yapıların hasara karşı dayanıklı olması gerekmektedir. Yeterince sürdürülebilir olmayan ve hasar görmüş binalar mümkünse onarılmalı veya onarımı mümkün olmayan yapılar kullanılmamalıdır. Mevcut hasarsız yapıların hasar görme riski varsa bu yapıların hasar görmeden güçlendirilmesi ve daha güvenli hale getirilmesi gerekmektedir (Keskin E, 2005). Hasarlı yapıların onarılmasının temel amacı, onarım ve güçlendirme sonrasında yapının aynı ya da daha büyük depreme dayanabilmesidir. Onarımın konusu, binanın taşıyıcı olmayan sıva vb. yapı parçalarının onarılması ve kapatılması değil, yapının görünüm veya bu yapının işlevsel görünümü bozulan bir veya birkaç parçasının üzerinde çalışma ve değişiklik yapılması amacıyla yapının eski durumuna geri getirebilmektir (Şirin C, 2006). Takviye ise bir yapının rijitliğini, yük taşıma kapasitesini, mukavemetini ve ağırlığını daha önceki veya mevcut duruma göre artırmak amacıyla yapılmaktadır. Bazı durumlarda onarım maliyetleri yeniden inşa maliyetlerinin yarısını aşmaktadır. Çatlakların onarımı için genişliklerine bağlı olarak farklı yöntemler kullanılabilir (Bayülke N, 1984; Demir, 1992). Hasarın genişliğine bağlı olarak farklı yöntemler kullanarak çatlakları onarımı yapılabilir. Çatlakların onarılmasının nedeni, betonarme donatının zamanla korozyona uğramasına neden olan nemin bu çatlaklardan dışarı sızmasını önlemektir ve enine çatlaklı betonarme elemanların kesitteki rijitliği düşük ve yapının mekanik özellikleri değişmemektedir (Bayülke N, 1984).

Türkiye deprem kuşağında yer aldığı için özellikle son yıllarda çok sayıda deprem felaketi yaşanmaktadır. Bu afetler sonucunda birçok bina yıkılmış ve hasar görmüştür. Hasar gören yapıların kurtarılmasının mümkün olup olmadığına mümkün olduğu kadar çabuk karar verilmesi gerekmektedir. Depremde ciddi hasar gören ancak çökmeyen çok az sayıda bina da yıkılmıştır. Birçoğu tamir ve takviye ile tekrar kullanılabilir hale getirilmiştir. Özellikle son dönemde yaşanan "Kahramanmaraş Depremi" sonucunda sorun yeniden ortaya çıkmaktadır. Bu deprem sebebiyle çok sayıda yapı yıkıma uğradı ve pek çoğu da artçılar sonucu hasara uğramıştır. Farklı seviyedeki teknik çalışanların; yeni binaların hazırlanması ve inşaat koşulları konusunun önemini hatırlamalarını sağlamak, bina sahiplerinin farkındalığını arttırmak ve bu alandaki boşluğu doldurmak hedeflenmektedir (Rasathanesi B, 2000).

2. MATERYAL VE METOT

Çalışmada yapılan betonarme olan beton ve çelik avantajlar ve inşaatla kullanımı ile ilgili yazılı ve görsel kaynak taraması yapılmıştır. Yapılan bu çalışmada sürdürülebilir inşaat anlayışı içinde değerlendirilen ve literatürde yer almış farklı işleve sahip olan betonarme yapılar üzerine yapılan çalışmalar analiz edilmiştir. İncelenen çalışmalarda, betonarme yapılarda döşeme hasarının onarım ve güçlendirilmesinin hangi amaçla ve nerelerde kullanıldığı irdelenmiştir.

Beton; TBDY-2018'e göre yapılacak tüm betonarme binalarda C25'ten daha düşük dayanımlı beton kullanılmasına izin vermemektedir. Türkiye'deki mevcut beton kalitesi ve kontrolü göz önüne alındığında, C30 sınıfı betonun C20 sınıfı betona göre birçok avantajı bulunmaktadır (Tablo1). Bu nedenle özellikle 6 Şubat 2023'de meydana gelen deprem sonrası hala revize edilmekte olan deprem yönetmeliğimizde (TBDY2018) minimum beton sınıfının en az C30 olması gerektiği düşünülmektedir. C30 sınıfı beton kullanımının, betonda yeni bir aşamaya ulaştığı ve olumlu etkilerinin özellikle depreme dayanıklı binaların üretiminde gözle görülür hale gelmiştir (Çağatay İ. H. 2016).

Tablo1. Beton Sınıfları ve Dayanımları (TS500)

Beton Sınıfı	Silindir N/mm ²	Küp N/mm ²
C16/20	16	20
C20/25	20	25
C25/30	25	30
C30/37	30	37
C35/45	35	45
C40/50	40	50
C45/55	45	55
C50/60	50	60

Çelik; pürüzsüz yüzeyli 220 çeliği yalnızca döşemelerde veya sargı olarak kullanılabilir. Kiriş, perde ve kolon uçlarında ise yönetmelik gereğince kullanılması yasaklanmıştır. Uygulamada da kullanımına izin verilmemektedir.

Dayanımı 500 N/mm^2 olan çelikler döşemelerde, levhalarda ve istinat duvarlarında kullanılabilir. Kolon bölümünde kiriş ve perde uçlarında sadece 420 N/mm^2 mukavemetli nervürlü çelik kullanılmaktadır. Minimum çekme mukavemeti S220a sınıfı için 340 MPa , doğal sert S420a çeliği için 500 MPa ve diğer çelikler için 550 MPa 'dır (Topçu A, 2015).

Epoksi; matris içeren elyaf takviyeli kompozit malzemeler, yüksek mukavemet ($3,5 \text{ MPA}$) ve sertlikleri nedeniyle endüstride ileri teknoloji ürünü olarak yaygın olarak kullanılmaktadır. Günümüzde gürültü, titreşim, ısı yalıtım, yüksek dayanım ve korozyon direnci ve nedeniyle çok çeşitli uygulamalarda kullanılmaktadır (Demirci İ, 2017).

FRP (Fiber takviyeli plastikler); alternatif olarak, korozyon direnci, yüksek mukavemet/ağırlık oranı, iyi yorulma özellikleri ve kullanım kolaylığı gibi avantajları nedeniyle fiber takviyeli polimer (FRP) takviyesi kullanımı önerilmektedir (Lokman G, 2020).

CFRP (Karbon fiber takviyeli polimerler); polyester reçine içine gömülmüş, çapı 5 ila 10 mikrometre olan çok ince karbon fiberlerden oluşmaktadır. CFRP, çelikten 5.2 kat daha hafiftir. Şu anda inşaat mühendisliğinde yaygın olarak kullanılmasını engelleyen temel neden ise yüksek fiyatıdır ($\text{CHF } 25/\text{kg}$). Ancak hızlı teknolojik gelişme dikkate alındığında yakın gelecekte CFRP uygulamasının daha sık kullanılması için ekonomik hale gelmesi beklenmektedir (Aytaç E, 2011).

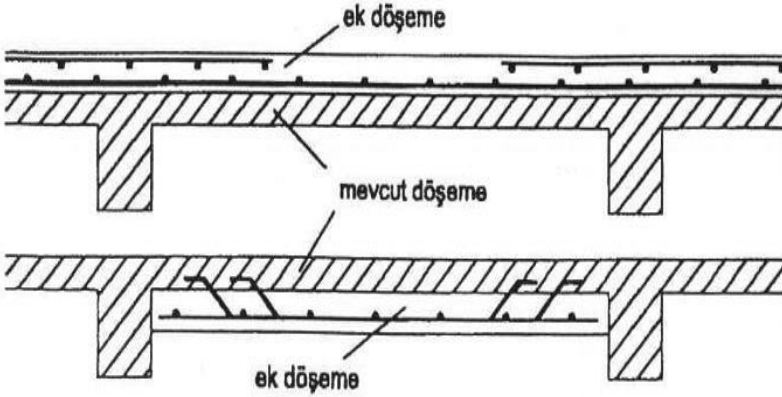
3. DÖŞEMENİN ONARIMI VE GÜÇLENDİRMESİ

Betonarme yapılarda döşemeler, dikey yükleri taşıyacak şekilde tasarlanmıştır. Ancak yatay yüklere maruz kaldığında zemin seviyesinde membran etkisine sahip olmalı ve yatay yükü destek sisteminin tüm yatay yük taşıyıcı elemanlarına aktarabilecek mukavemet ve rijitliğe sahip olması gerekmektedir. Döşemelerde hasar genellikle döşemenin yatay kuvvetlerin yoğunlaştığı bir duvara, döşeme boşluklarına ve merdivenlere doğrudan dayanması sonucu meydana gelmektedir. Hasarın onarılmasının yanı sıra döşemenin yetersiz kalınlığının artırılması ve yeni perdelerle yükün aktarılmasına sahip olmaktadır (Şirin C, 2006).

Çatlaklara epoksi, çimento harcı, tamir harçları ve ham beton enjekte edilerek lokal tamirler yapılmaktadır. Betonun takviye levhalarında ufalanması, kırılması veya bükülmesi durumunda hasarlı kısım çıkarılıp yenisi ile

değiştirilmektedir. Hasarlı kısım çıkarıldıktan sonra mevcut donatıya yeni donatı yerleştirilip kaynak yapılmalı ve beton dökülmelidir.

Döşemenin kalınlığı arttıkça parçanın eğilme mukavemeti artmaktadır. Döşemenin kalınlığı yukarıdan bir beton tabakasıyla veya aşağıdan püskürtme yöntemi kullanılarak arttırılabilmektedir (Keskin E. (., 2005).

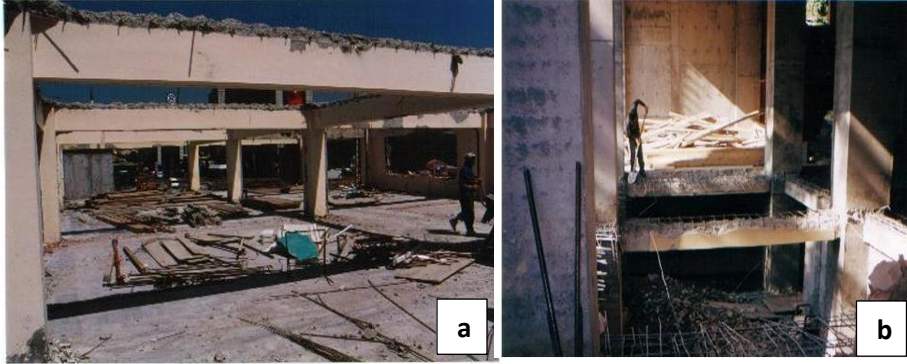


Şekil 1: Döşemelerin kalınlığını artırarak güçlendirilme (Ceritli, 2006)

Bu şekilde eklenen parçanın mevcut döşemenin aderansının sağlanabilmesi çok önem verilmelidir. Şekil 1'de gösterildiği gibi mevcut zemine epoksi ile çelik çubukların eklenmesi, profil parçalarının çelik pimlerle mevcut zeminin pürüzlendirilmesi, mevcut zemine sabitlenmesi ya da mevcut boşlukların açılmasıyla gerçekleştirilmektedir (Celep Z, 2000).

Döşemeyi güçlendirmenin bir başka yolu da onu çelik levhalarla güçlendirmektir. Çelik levhanın eğilme momenti mukavemetini arttırmak amacıyla çelik levhanın sadece alt yüzeyi epoksi gibi yapıştırıcılarla yapılmaktadır (Çevik M, 2003).

Döşemeyi sağlamlaştırmanın yanı sıra, diğer tamir ve güçlendirme yöntemlerine göre pratik olmasa da döşemenin yıkılıp yeniden yapılması daha ekonomiktir bir yöntemdir (Şekil 2). Ayrıca kat yüksekliğini düşürmemesi ve kullanım değişikliği gerektirmemesi nedeniyle daha kullanışlı bir yöntemdir. Mevcut döşemenin giriş destekleriyle birlikte kırılması durumunda mevcut donatı kullanılarak ilave donatı eklenir ve döşeme betonu istenilen kalınlığa kadar dökülmesi sağlanmaktadır (Şirin C, 2006).



Şekil 2 a) Döşemenin kırıp yeniden inşa edilmesi **b)** zemin kat döşemesinin kırıp yeniden inşa edilmesi (Çevik M Ö., 2003).

3.1. Fiber Takviyeli Plastikler (FRP) İle Döşeme Güçlendirme

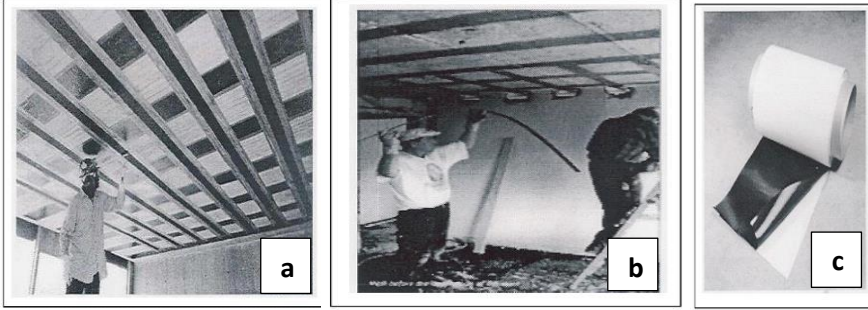
FRP ile uygulama takviyesi, şu anda bina restorasyon süreçlerinde kullanılan yaygın bir tekniktir. Diğer birçok kimyasal elemente, özellikle de klorür iyonlarına karşı mükemmel dirence sahiptir. Bu nedenle tavanda paslanmaya neden olmaz ve yüksek kalitede olup, mükemmel korozyon direncine sahip malzemelerdir. Bu sayede yıllarca hiçbir sorun yaşamadan kullanabilmektedir.

FRP ile döşeme takviyesinin bir diğer önemli avantajı ise çekme mukavemeti ve ağırlığının çok düşük olmasıdır. FRP malzemelerin teknik tasarımı, sertliği, dayanıklılığı, esnekliği, yüksek performansı ve diğer özellikleride, güçlendirme yöntemlerinde kullanımı giderek artmaktadır.

FRP uygulaması bir yönlü veya iki yönlü düzenlemeye sahip döşemelerde çalışma yükünü önemli ölçüde artırmaktadır (Şekil 3). Uygulama yerleri; inşaat ve sanayi binalarının döşemeleri, köprü tabliyeleri vb. Tipik güçlendirme ihtiyaçları (Atay H, 2010):

- Kullanım amacı değişikliği nedeniyle servis yükünün artması,
- Yeni merdiven veya asansör açılması,
- Döşeme duvarlarının yıkılması,

- Trafik yükünün artması,



Şekil 3 a) FPR döşemesinin iki doğrultuda yapıştırılması **b)** Mbrace FPR **c)** FPR laminat şekli (Ysl S., 2002)

3.2. CFRP (Karbon Fiber Takviyeli Polimerler) Plakalarla Döşeme Güçlendirme

Döşeme sistemindeki birincil bileşen olarak önerilen bu CFRP zemin paneli, binalardaki geleneksel beton zemin döşemesinin yerini almayı amaçlamaktadır. Yeni CFRP zemin paneli, hafifliği nedeniyle binalara kolaylıkla monte edilebilmekte ve tasarlanan şekli nedeniyle asma tavan, havalandırma ve aydınlatma sistemiyle kolayca entegre edilebilmektedir. Düşük karbonlu olması, inşaat sektöründe enerji tasarrufu ve CO₂ emisyonunu azaltma açısından potansiyel bir bina bileşeni sağlamaktadır (Şekil 4).



Şekil 4 a) CFRP ile güçlendirmek (Pehlivanlı M., 2023) **b)** Döşeme düzeyi **c)** CFRP ekledikten sonra (Hsu, 1756)

Epoksi matrisli ve karbon fiber levhalı döşeme; kolon, duvar ve kiriş yük taşıma kapasitesini arttırmaya yönelik dış takviye malzemelerdir. Bu paneller, yüksek mukavemetli epoksi reçine kullanılarak bileşenin gerilebilir yüzeyine bağlanmaktadır. Epoksi kuruduktan sonra yük panele aktarılmaktadır. Yüklendiğinde paneller akma noktasına kadar yüksek yüklere dayanabilmektedir. Elastik modüllerine bağlı olarak çeşitli karbon plaka türleri bulunmaktadır. Takviye edilecek elemanların yükleme şartlarına ve açıklıklarına bağlı olarak uygun levha tipi seçilmektedir. Yüksek elastik modüle sahip plakalar kullanılarak, donatı çubuklarının deformasyonunu ve çatlak genişliğini azaltmak ise mümkündür. Aşırı sehim nedeniyle takviye gerektiren elemanlar için yüksek modüllü plakalar kullanılması tercih edilmektedir. Karbon plakalar önerilmeli olarak da kullanılabilir (Atay H, 2010).

3.3. Döşemelerin Perde Duvarı İle Güçlendirilmesi

Döşemeler genellikle dikey yükler için yapılmaktadır. Düşey mesnetler (kolonlar, perde duvarlar vb.) üzerindeki yatay yüklerin etkisi altında bu yükleri taşıyabilecek mukavemet ve rijitliğe sahip olmaktadır. Hasar genellikle doğrudan giydirme cephe üzerine desteklendiği geniş boşluklu döşeme bölümlerinde meydana gelmektedir. Hasarın onarılmasının yanı sıra, yeterince kalın olmayan bir levhanın kalınlığının artırılması veya yeni bir giydirme cephe ile yük aktarımının sağlanması gerekmektedir (Oymael S., 2016). Güçlendirilecek deprem döşemesi, mevcut döşeme ile yeni ilave kalınlık arasındaki aderansın etkinliğine ve boyutlandırılacak ve demir hesabı yapılmaktadır (Keskin E, 2005). Sistem yeni bir giydirme cephe eklenerek güçlendirilirse, giydirme cephe ile ona çarpan kiriş arasındaki tüm boşlukların kaldırılması gerekmektedir. Bunun için harç kullanılmalıdır (Şekil 5). Yerel çatlaklar ve hasarlar epoksi reçine, çimento harcı veya püskürtme betonla onarılabilir. Özellikle betonun hasarlı kısımları çıkarılıp kırılan veya eğilen donatılar kesilerek kaynak yapılarak yeni donatılar eklenmektedir. Döşemenin kalınlığı arttıkça kesitinin eğilme mukavemeti de artmaktadır. Üstten yeni bir beton tabakası dökülerek veya alttan kaba beton uygulanarak levhanın mukavemeti artırılabilir. Yeni parçalara da uygun takviyeler yerleştirilmektedir. Yeni ve mevcut döşeme arasında kayma geriliminin akışını sağlamak için eski yüzeyin pürüzlendirilmesi veya çelik çubuklarla bağlanması yararlı olmaktadır. Yük taşıma kapasitesini yeniden sağlamak için direğin

onarılması veya güçlendirilmesi gerekmektedir. Kolonun mukavemeti, mantolama yöntemi kullanılarak kesitin artırılmasıyla sağlanabilmektedir (Oymael S., 2016).



Şekil 5. Perde/döşeme güçlendirme (Beşoğlu İ, 2019)

4. BULGULAR

Türkiye’de birçok betonarme yapı gerekli inşaat standartlarını karşılamamaktadır. Bu tür yapıların yıkılıp yeniden yapılması çok pahalı olacağından bu yapıların mümkünse onarılması veya güçlendirilmesi gerekmektedir. Ancak bu kararın deneyimli kişiler tarafından onaylanması gerekmektedir. Yapıları onarmak veya güçlendirmek çoğu zaman bir yapıyı yeniden inşa etmekten çok daha zordur (Keskin E. (., 2005). Hasarın seviyesini öğrenip binanın nerede ve nasıl onarılacağına ve güçlendirileceğine karar vermek gerekmektedir. Bu çalışmada, gerekli literatür araştırması yapılarak birçok bulgu ortaya çıkmıştır (Çetinkaya N, 2011). FRP malzemeleri kullanarak onarım/güçlendirme yöntemine karar verirken elemanın kesme dayanımının kontrol edilmesi gerekmektedir. Epoksi harcın yapışmasını arttırmak için FRP malzemenin uygulandığı yüzey pürüzlü olması gerekmektedir. Bu işlem matkap, kumlama ya da elektrik çelik tel fırça ile yapılabilir. Epoksi reçine çatlaklara püskürtüldüğünde, enjeksiyon dübeli için açılan deliğin hava ile iyice temizlenmesi gerekmektedir. Döşeme inşa edilirken mümkün olduğunca hafif malzemelerin kullanılmasına dikkat edilmelidir. Lifli polimer örtü; lifli polimer malzeme çelikten çok daha güçlü ve daha hafif ve yapıya ek bir gerilim katmamaktadır. Beton elemanların sismik etkilere ve düşey yüklere karşı taşıma kapasitesini arttırmaktadır. Kullanımı kolaydır ancak betonarme veya inşaat demirine göre daha pahalı malzemelerdir. Yangına dayanıklı değildir ve koruyucu malzeme ile kaplanması gerekmektedir (Al

Asadi S, 2022). Deprem öncesi döşeme durumunun araştırılması, mevcut döşemedeki eksiklikleri belirlemek için binanın düşey yüklere ve sismik etkiye (1.4G+1.6Q) maruz kaldığında hasar görme durumu analiz edilmeli, (G+Q+E) TS500 ve 1998 Deprem Yönetmeliği esas alınarak hazırlanması gerekmektedir (Çalışkan Ö., 2012).

5. SONUÇ VE ÖNERİLER

Çalışmada betonarme yapılarda döşemelerde deprem hasarlarını onarım ve güçlendirme yöntemleri en iyi yollarını araştırılmıştır. Döşemelerin onarımı ve güçlendirme için gerekli yöntemlerden bahsedilmiş ve bu yöntemlerin avantajlarından ve kullanım yolları hakkında gerekli bilgiler yapılan literatür taraması sonucu verilmiştir. Güçlendirme sorunları dikkatle ele alınmalıdır. Çünkü pek çok betonarme yapıda bu tarz işlemler gerekebilmektedir. Bina hasarlarının nedenleri ve sonuçları dikkatle düşünülmeli ve onarılacak her yapı ayrı ayrı değerlendirilmelidir. Bu alanda uzman olan insanlar ile döşemelerin öncelikle uygun tamir/takviye malzeme ve yöntemlerini belirlemesi gerekmektedir. Bundan sonra yapıların ve mimari elemanların onarımına yönelik hazırlıkların, yapıların korunmasının ve dayanıklılığının sağlanması konusunun değerlendirilmesi gerekmektedir. Prefabrik onarım malzemeleri ve takviyeleri kullanılarak yapılan çeşitli testlerin sonuçları ile üreticilerin iddia ettiği özellikler arasında çeşitli farklılıklar bulunmuştur. Bu nedenle hazır tamir harcını kullanmadan önce bazı özelliklerinin kontrol edilmesi ve amaca uygun olup olmadığının belirlenmesi gerekmektedir (Özturan T, 2003). Güçlendirme projesi hazırlanırken aşağıdaki hususlara dikkate alınmalıdır (Atay H, 2010):

- Tasarım öncesinde binanın detaylı bir mimari ve yapısal etüdü yapılmalıdır.
- Donatı spesifikasyonları doğru şekilde yapılmalıdır.
- Olası korozyon durumları dikkate alınmalıdır.
- Mimari planı bozmamak ve yapıya verilen zararı en aza indirmek için doğru montajın yapılması gerekmektedir.
- Simetrik güçlendirme yapılması ve elemanların düzenlenmesi gerekmektedir.

KAYNAKÇA

- Addleson, L., 1992. Building failures, butterworth- heinemann ltd., ingiltere
- Akgül, M., & Serkan, E. T. L. İ. (2023, April). 06 Şubat 2023 Kahramanmaraş (Pazarcık, Elbistan) Depremleri Sonrası Betonarme Binalarda Gözlenen Hasar Durumları. In International Conference on Scientific and Innovative Studies (Vol. 1, No. 1, pp. 309-318)
- Al Asadi s, (2022). Betonarme Yapılarında Kolonların Güçlendirme ve Onarım Yöntemleri. TAS Journal, 2(2), 25-34.
- Atay H, (2010). Depremde hasar görmüş yapıların güçlendirme yöntemleri ve güçlendirmede kullanılan malzemeler (Doctoral dissertation, Sakarya Üniversitesi (Turkey)).
- Aytaç E, (2011). CFRP güçlendirme malzemesi ve güçlendirme teknikleri (Doctoral dissertation, DEÜ Fen Bilimleri Enstitüsü).
- Bayülke N, "Depremlerde Hasar Gören Yapıların Onarımı ve Güçlendirilmesi", Yapı Malzemesi ve Deprem Araştırma Genel Müdürlüğü, Ankara, 1984
- Büyükcınar, Ö. F., 1984. Yapı Elemanı Tasarımında Gereksinim-Olanak Dengesi ve Malzeme Seçimi Açısından Bir Yöntem, Yüksek Lisans Tezi, Sosyal Bilimler Enstitüsü İ.T.Ü., İstanbul.
- Celep, Z, N., 2000. Kumbasar, Deprem Mühendisliğine Giriş ve Depreme Dayanıklı Yapı Tasarımı, Beta Dağıtım, İstanbul.
- Ceritli, Y. S. (2006). Depremde hasar görmüş yapıların mantolama ve perdelerle güçlendirilmesi.
- Çalışkan Ö., YILMAZ, S., & KAPLAN, H. (2012). Güçlendirme Ankrājlarının Kesme Dayanımının ACI318 ve TS500'e Göre Belirlenmesi. Kahramanmaraş Sütçü İmam Üniversitesi Mühendislik Bilimleri Dergisi, 14(3), 19-27.
- Çelebi, G., 1997. Yapı Malzemeleri Performans ve Yapıda Kullanım Özellikleri, Ankara.

- Çetinkaya N, Kaplan, H., & Şenel, Ş. M. (2011). Betonarme kirişlerin lifli polimer (frp) malzemeler kullanılarak onarım ve güçlendirilmesi. Pamukkale üniversitesi mühendislik bilimleri dergisi, 10(3), 291-298.
- Çevik M, Ö. (2003). Betonarme yapıların onarım ve güçlendirme yöntemlerinin ve bir konut yapısının güçlendirme projesinin hazırlanarak güçlendirme sonrası dinamik karakteristiklerindeki değişikliklerin incelenmesi (Doctoral dissertation, Fen Bilimleri En
- Demir., H, 1999. Betonarme Yapıların Onarım ve Güçlendirilmesi, Emlak Pazarlama Proje Yönetimi ve Servis A. Ş., İstanbul.
- Demirci İ, & Ahmet, A. V. C. I. (2017). Deniz suyu korozyon ortamında bazalt/epoksi kompozitlerin mekanik davranışlarının incelenmesi. Selcuk University Journal of Engineering Sciences, 16(2), 44-54.
- Duman, H., 2004. Malzemenin Kullanımına İlişkin Özellikleri ile Fiziksel Mekanik-Kimyasal Özelliklerinin Etkileşiminde Ürün Geliştirme Sürecinin Analizi, Yüksek Lisans Tezi, İ.T.Ü. Fen Bilimler Enstitüsü, İstanbul.
- Erten, E., 1996. Binaferda Nem Yalıtımı, Karadeniz Teknik Üniversitesi MühMim Fakültesi Fakülte Ders Notları, No:46, Trabzon.
- Esin, T., 1996. "Doğal Taş Malzemenin Fiziksel Özellikleri ve Dayanıklılık İlişkisi", Yapı, Sayı No 171, s.97-102
- Folic, R, J, 1991. Materials and Structures, 24, 286.
- Gökaltun E., 2001, Yapıların Zemine Oturan Döşemelerinde Ortaya Çıkan Nem Sorunları ve Yalıtım Çözümleri, TMMOB Makina Mühendisleri Odası Yalıtım Kongresi, Eskişehir
- Göntül İ.A. ve Çelebi G., 2003. Binalarda Zeminden Kaynaklanan Nemlenmeyi Önleme Yöntemleri, Gazi Üniv. Müh. Mim. Fak. Der. Cilt 18, No 4, 109-122, Ankara

- Göntül, İ.A., 2000. Yapılarda Zeminden Kaynaklanan Nemlenmeyi Önleme Yöntemlerinin Belirlenmesi, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- Gülkan, P, A. Yakut, H. Sucuoğlu, M. S. Yüçemen, E. Çıtıpıtıoğlu, 1994. Mühendislik Hizmeti Görmüş Yapılar
- Gürdal, E., 2009. Yapıda Hasar, İstanbul Teknik Üniversitesi Mimarlık Fakültesi Ders Notları, İstanbul.
- Hsu, W. L., Liu, C. C., Shiau, Y. C., & Lin, W. C. (2019). Discussion on the reinforcement of reinforced concrete slab structures. *Sustainability*, 11(6), 1756.
- <http://tr.wikipedia.org/wiki/Korozyon>, 04.Nisan.2009
- Keskin E, (2005). Betonarme yapıların onarımı, güçlendirilmesi ve lifle güçlendirilmiş polimerler (Doctoral dissertation, Fen Bilimleri Enstitüsü).
- Kömür, M, m. altan, 2005. İtü dergisi/d mühendislik, 4-43.
- Kumaran, M.K., Mitalas, G.P. ve Bomberg, M.T., 1994. Fundamentals of Transport and Storage of Moisture in Building Materials and Components, Moisture Control in Buildings, Cilt 18, Editör: Trechsel, H.R., ASTM, A.B.D., s.3-17
- Kurtuluş, T. (2009). Yapı döşemelerinde hasar analizi (Doctoral dissertation, Fen Bilimleri Enstitüsü).
- Kuşaslan, Deniz., Kasım 2002. Yapıda Durabilite ve Hasar Analizi Alçılı Kerpiç Yapı Örenği Üzerinde İncelenmesi, Yüksek Lisans Tezi, İTÜ. Fen Bilimleri Enstitüsü, İstanbul.
- Lokman G, E. M. İ., madenci, e., & özkılıç, y. o. (2020). Çelik, Cam FRP ve hibrit donatılı betonarme kirişlerin eğilme performansının incelenmesi. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 8(2), 1470-1483.

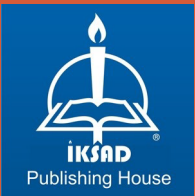
- Maeda, M; Matsukawa, K; Ito, Y. (2014, July). Revision of guideline for post-earthquake damage evaluation of RC buildings in Japan. In Tenth US National Conference on Earthquake Engineering.
- Onaran, K., 1993. Malzeme Bilimi, Bilim Teknik Yayınevi, İstanbul.
- Oxley, T. A. ve Gobert, E. G., 1985. Dampness in Buildings, Page Bros Ltd., İngiltere
- Oymael S., (2016), Yapı malzemesi ve yapı fiziği ilişkisi, Birsen yayın evi, İstanbul
- Özturan T., (2003), Betonarme yapıların restorasyonu ve onarım ve güçlendirme malzemeleri, İMO.
- Pehlivanlı M, (2023). Aynı profilde, farklı açılarda elle serim yapılarak üretilmiş kompozit parçaların incelenmesi (Master's thesis, Bartın Üniversitesi, Fen Bilimleri Enstitüsü).
- Rasathanesi B, Ü. K., & Enstitüsü, D. A. (2000). Depremde Hasar Gören Betonarme Yapıların Onarımı ve Güçlendirilmesi. Deprem Araştırma Bülteni.
- Şirin, C. (2006). Yapılarda oluşan hasar biçimleri ve nedenleri ve yapıların onarım ve güçlendirilmesi teknikleri ile bir yapının güçlendirilmesi (Doctoral dissertation, Fen Bilimleri Enstitüsü).
- Tankut, T. 1999. Deprem Güvenli Konut Sempozyumu, Ankara, 87.
- TBDY-2018, "Türkiye Bina Deprem Yönetmeliği ", Ankara, 2018.
- The Japan Building Disaster Prevention Association (JBDPA), 2001. Guideline for Post-earthquake Damage Evaluation and Rehabilitation of RC Buildings, Japan
- Topçu A, Ahmet, Betonarme I, Eskişehir Osmangazi Üniversitesi, 2015, <http://mmf2.ogu.edu.tr/atopcu>
- Üzümeri, Ş, M, T. Tankut, G. Özcebe, E. Atımtay, Uğur Ersoy, 1999 Symposium on Structural Engineering, Middle East Technical University Department of Civil Engineering, Ankara, 413.

Yılmaz & Umu, <http://w3.bilecik.edu.tr/insaat/wp-content/uploads/sites/28/2017/10/5-YAPILARDA-HASAR-BETONARME-YAPI-HASARLARI.pdf>

Yönetmelik, B. Y. K. H. (2007). Resmî Gazete Sayı: 26735. Access address: <https://www.resmigazete.gov.tr/eskiler/2007/12/20071219-2.htm>.

Ysl S., Gold, W. and Sinider, R., 2002. Composite Strenghtening System, wabombrace. Newyork..

Yüksel, İ. (2008). Betonarme binaların deprem sonrası acil hasar değerlendirmeleri. Erciyes üniversitesi fen bilimleri enstitüsü fen bilimleri dergisi, 24(1), 260-276.



ISBN: 978-625-367-840-1