

**T.C.**  
**FIRAT ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**



**GERÇEK ZAMANLI RAY TAKİBİ İÇİN OTONOM İHA**  
**ALGORİTMALARININ GELİŞTİRİLMESİ VE GÜRBÜZ DERİN**  
**ÖĞRENME TABANLI KUSUR TESPİTİ**

**Emre GÜÇLÜ**

Yüksek Lisans Tezi

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

Kuramsal Temeller Bilim Dalı

TEMMUZ 2022

**T.C.**  
**FIRAT ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

Bilgisayar Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

**GERÇEK ZAMANLI RAY TAKİBİ İÇİN OTONOM İHA**  
**ALGORİTMALARININ GELİŞTİRİLMESİ VE GÜRBÜZ DERİN**  
**ÖĞRENME TABANLI KUSUR TESPİTİ**

Tez Yazarı  
**Emre GÜÇLÜ**

Danışman  
Prof. Dr. Erhan AKIN

TEMMUZ 2022  
ELAZIĞ

**T.C.**  
**FIRAT ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

Bilgisayar Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

|                    |                                                                                                                         |
|--------------------|-------------------------------------------------------------------------------------------------------------------------|
| Başlığı:           | Gerçek Zamanlı Ray Takibi İçin Otonom İHA Algoritmalarının Geliştirilmesi Ve Gürbüz Derin Öğrenme Tabanlı Kusur Tespiti |
| Yazarı:            | Emre GÜÇLÜ                                                                                                              |
| İlk Teslim Tarihi: | 02.06.2022                                                                                                              |
| Savunma Tarihi:    | 04.07.2022                                                                                                              |

**TEZ ONAYI**

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına göre hazırlanan bu tez aşağıda imzaları bulunan jüri üyeleri tarafından değerlendirilmiş ve akademik dinleyicilere açık yapılan savunma sonucunda OYBİRLİĞİ ile kabul edilmiştir.

|           |                                                                   |                          |
|-----------|-------------------------------------------------------------------|--------------------------|
| Danışman: | Prof. Dr. Erhan AKIN<br>Fırat Üniversitesi, Mühendislik Fakültesi | <i>İmza</i><br>Onayladım |
|-----------|-------------------------------------------------------------------|--------------------------|

|         |                                                                   |           |
|---------|-------------------------------------------------------------------|-----------|
| Başkan: | Doç. Dr. İlhan AYDIN<br>Fırat Üniversitesi, Mühendislik Fakültesi | Onayladım |
|---------|-------------------------------------------------------------------|-----------|

|      |                                                                                                    |           |
|------|----------------------------------------------------------------------------------------------------|-----------|
| Üye: | Dr. Öğr. Üyesi İbrahim Rıza HALLAÇ<br>Alanya Alaaddin Keykubat Üniversitesi, Mühendislik Fakültesi | Onayladım |
|------|----------------------------------------------------------------------------------------------------|-----------|

Bu tez, Enstitü Yönetim Kurulunun ...../...../20..... tarihli toplantısında tescillenmiştir.

*İmza*  
Prof. Dr. Kürşat Esat ALYAMAÇ  
Enstitü Müdürü

## BEYAN

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım “Gerçek Zamanlı Ray Takibi İçin Otonom İHA Algoritmalarının Geliştirilmesi Ve Gürbüz Derin Öğrenme Tabanlı Kusur Tespiti ” Başlıklı Yüksek Lisans Tezimin içindeki bütün bilgilerin doğru olduğunu, bilgilerin üretilmesi ve sunulmasında bilimsel etik kurallarına uygun davrandığımı, kullandığım bütün kaynakları atıf yaparak belirttiğimi, maddi ve manevi desteği olan tüm kurum/kuruluş ve kişileri belirttiğimi, burada sunduğum veri ve bilgileri unvan almak amacıyla daha önce hiçbir şekilde kullanmadığımı beyan ederim.

04.07.2022

**Emre GÜÇLÜ**



# ÖNSÖZ

Bir demiryolu hattını denetlemek, hem yöntem hem de organizasyon açısından karmaşık bir işlemdir. Demiryolu denetiminde temel hedef, denetleme maliyetini ve harcanan zamanı en aza indirirken demiryolu trafiğinin güvenliğini sağlamaktır. Bu çalışmada, otonom İHA kullanarak demiryollarındaki kusurların tespiti için çeşitli algoritmalar önerilmiştir. Denetim işleminde ilk olarak, görüntü toplama işlemi için otonom İHA algoritmaları tasarlanmıştır. Görüntü toplamanın otonom İHA tarafından yapılması denetim sırasında rayları meşgul etmemesi ve görüntü toplama zamanını azaltması gibi avantajlar sağlamıştır. Toplanan demiryolu görüntülerinden demiryolu bileşenlerindeki kusurların tespiti yapılmıştır. Kusurların tespiti için makine öğrenmesi, derin öğrenme ve bulanık mantık yapısına dayalı yeni yöntemler önerilmiştir.

Bu çalışmada beni yönlendiren ve bana her türlü desteği veren danışman hocam sayın Prof. Dr. Erhan AKIN'a teşekkürlerimi sunarım. Çalışma süresince, şahsımdan hiçbir zaman desteğini esirgemeyen ve çalışmalarımda fazlasıyla emeği geçen hocam sayın Doç. Dr. İlhan AYDIN'a teşekkür ederim.

Bu tez çalışması, 120E097 numaralı TÜBİTAK 1001 araştırma projesi ve Fırat Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi (FÜBAP) tarafından MF.21.72 protokol numaralı proje ile desteklenmiştir.

**Emre GÜÇLÜ**  
ELAZIĞ, 2022

# İÇİNDEKİLER

Sayfa

|                                                                                                |           |
|------------------------------------------------------------------------------------------------|-----------|
| ÖNSÖZ.....                                                                                     | iv        |
| İÇİNDEKİLER .....                                                                              | v         |
| ÖZET .....                                                                                     | vii       |
| ABSTRACT .....                                                                                 | viii      |
| ŞEKİLLER LİSTESİ .....                                                                         | ix        |
| TABLOLAR LİSTESİ .....                                                                         | xiii      |
| SİMGELER VE KISALTMALAR .....                                                                  | xiv       |
| <b>1. GİRİŞ .....</b>                                                                          | <b>1</b>  |
| 1.1. İHA Türleri.....                                                                          | 5         |
| 1.2. Çok Rotorlu İHA'lar.....                                                                  | 6         |
| 1.3. Literatür Özeti .....                                                                     | 8         |
| 1.4. Demiryolu Bakımında Kullanılan Teknolojiler .....                                         | 11        |
| 1.4.1. Sensörler.....                                                                          | 11        |
| 1.4.2. Denetim Platformları.....                                                               | 13        |
| <b>2. MATERYAL VE METOT .....</b>                                                              | <b>16</b> |
| 2.1. Otonom Ray Takibi .....                                                                   | 16        |
| 2.1.1. HSV'ye Dayalı Ray Takip Algoritması .....                                               | 17        |
| 2.1.2. Gabor Filtresi ve Bulanık PID Tabanlı Ray Takibi .....                                  | 27        |
| 2.1.3. Hafif Ağırlıklı Çizgi Tespiti Yaklaşımına Dayalı Ray Takibi.....                        | 32        |
| 2.1.4. Yolact Tabanlı Ray Takibi .....                                                         | 34        |
| 2.2. Mask R-CNN Algoritması Tabanlı Travers Boşluklarının Tespiti .....                        | 37        |
| 2.2.1. Mask R-CNN .....                                                                        | 37        |
| 2.2.2. Maske Sınırlarına Dayalı Ölçüm Yöntemi .....                                            | 39        |
| 2.2.3. İkili Maskeye Dayalı Travers Eksikliklerinin Tespiti .....                              | 41        |
| 2.3. Görsel Kelime Çantası'na Dayalı Kusur Tespit Yöntemi.....                                 | 43        |
| 2.3.1. Görüntü Önileme Adımı ve İyileştirme .....                                              | 44        |
| 2.3.2. Bağlantı Elemanı Konumlarının Çizgi Yerel İkili Örüntü(ÇYİÖ) Tabanlı Tespiti .....      | 45        |
| 2.3.3. Özellik Çıkarımı ve Kusur Sınıflandırma.....                                            | 46        |
| 2.4. YOLOv4 ve Bulanık Mantık Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti .....     | 50        |
| 2.4.1. Bağlantı Elemanlarının Tespiti İçin YOLOv4 Algoritmasının Uygulanması .....             | 50        |
| 2.4.2. Kusur Tespiti İçin Tasarlanan Bulanık Mantık Yapısı .....                               | 52        |
| 2.5. DCGAN ve Siyam Sinir Ağı Kullanarak Bağlantı Elemanı Kusur Tespiti .....                  | 55        |
| 2.5.1. DCGAN ile eğitim görüntülerinin çoğaltılması .....                                      | 56        |
| 2.5.2. Siyam Ağı Modeli .....                                                                  | 58        |
| 2.6. Gürbüz Derin Öğrenme Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti .....         | 62        |
| 2.6.1. Popüler Düşman Saldırı Yöntemleri .....                                                 | 63        |
| <b>3. BULGULAR VE TARTIŞMA .....</b>                                                           | <b>66</b> |
| 3.1. Otonom Ray Takip Algoritmaları İçin DeneySEL Sonuçlar.....                                | 66        |
| 3.1.1. HSV'ye Dayalı Ray Takip Algoritmasının DeneySEL Sonuçları .....                         | 66        |
| 3.1.2. Gabor Filtresi Kullanılarak Ray Takibi İçin DeneySEL Sonuçlar .....                     | 67        |
| 3.1.3. Gabor Filtresi ve Bulanık PID Tabanlı Ray Takip Algoritmasının DeneySEL Sonuçları ..... | 69        |

|                                                                                                                   |           |
|-------------------------------------------------------------------------------------------------------------------|-----------|
| 3.1.4. Light-Weight Çizgi Tespiti Yaklaşımına Dayalı Ray Takip Algoritmasının Deneysel Sonuçları.....             | 73        |
| 3.1.5. Yolact'a Dayalı Ray Takip Algoritmasının Deneysel Sonuçları.....                                           | 74        |
| 3.2. Mask R-CNN Algoritması Tabanlı Travers Boşluklarının Tespiti için Deneysel Sonuçlar .....                    | 75        |
| 3.2.1. Maske Sınırlarına Dayalı Travers Eksikliklerinin Tespiti .....                                             | 75        |
| 3.2.2. İkili Maskeye Dayalı Travers Eksikliklerinin Tespiti .....                                                 | 76        |
| 3.3. Görsel Kelime Çantası'na Dayalı Kusur Tespit Yöntemi İçin Deneysel Sonuçlar.....                             | 78        |
| 3.4. YOLOv4 ve Bulanık Mantık Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti İçin Deneysel Sonuçlar ..... | 82        |
| 3.5. DCGAN ve Siyam Sinir Ağı Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti İçin Deneysel Sonuçlar ..... | 86        |
| 3.6. Gürbüz Derin Öğrenme Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti İçin Deneysel Sonuçlar .....     | 89        |
| <b>4. SONUÇLAR.....</b>                                                                                           | <b>92</b> |
| ÖNERİLER .....                                                                                                    | 93        |
| KAYNAKLAR.....                                                                                                    | 94        |
| ÖZGEÇMİŞ                                                                                                          |           |

## ÖZET

---

### Gerçek Zamanlı Ray Takibi İçin Otonom İHA Algoritmalarının Geliştirilmesi Ve Gürbüz Derin Öğrenme Tabanlı Kusur Tespiti

Emre GÜÇLÜ

Yüksek Lisans Tezi

FIRAT ÜNİVERSİTESİ  
Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Temmuz 2022, Sayfa: xiv + 98

---

Son yıllarda, demiryolu taşımacılığının kullanımı yüksek hızlı trenlerin de gelişimiyle önemli ölçüde artmıştır. Demiryolu ulaşımına olan bu talep sonucunda, dünyada ve ülkemizde demiryolu uzunluğu büyük ölçüde artırmaktadır. Bu nedenle, demiryolu bakımına olan talep de hızla artmış ve acil hale gelmiştir. Demiryolu bakımı için denetleme, demiryolu bileşenlerindeki kusurları aramak ve doğrulamak için önemli bir görevdir. Demiryolu bileşenlerinde meydana gelebilecek kusurlar, ulaşım güvenliğini önemli ölçüde etkiler. Hâlihazırda kullanılan denetim yöntemleri, temaslı ölçüm tekniklerine ve insan algılamasına bağlıdır. Bu nedenle geleneksel denetim, yavaş, öznel ve verimsizdir. Bu dezavantajların ve sınırlamaların üstesinden gelmek için bu çalışmada, otonom İHA (İnsansız Hava Aracı) bilgisayarlı görme tabanlı yeni bir yöntem önerilmiştir.

Demiryolu kusurlarının tespiti için İnsansız Hava Araçlarının kullanılması, geleneksel tekniklerden daha düşük denetim süreleri sunan uygun maliyetli bir yaklaşımdır. Son yıllarda, İHA'lar, işlem gücündeki gelişmeler ve sensörlerin ve bileşenlerinin küçülmesi hem askeri hem de sivil uygulamalar için ilgiyi artırmaktadır. Özellikle uçuş sürelerinin artması İHA'ların arıza tespit alanında kullanılabilecek seviyede olduğunu göstermektedir. Tezde, demiryolunun İHA tarafından otonom olarak takibi için algoritmalar geliştirilip İHA tarafından elde edilen görüntüler üzerinde derin öğrenme tabanlı kusur tespiti yapılmıştır.

**Anahtar Kelimeler:** İnsansız hava aracı, Demiryolu bakımı, Kusur tespiti, Derin öğrenme

## ABSTRACT

---

### Development of Autonomous UAV Algorithms for Real-Time Rail Tracking and Robust Deep Learning Based Fault Detection

Emre GÜÇLÜ

Master's Thesis

FIRAT UNIVERSITY  
Graduate School of Natural and Applied Sciences  
Department of Computer Engineering

July 2022, Pages: xiv + 98

---

In recent years, the use of rail transport has increased significantly with the development of high-speed trains. As a result of this demand for railway transportation, the length of the railway in the world and in our country is greatly increased. Therefore, the demand for railway maintenance has also increased rapidly and become urgent. Inspection for railway maintenance is an important task to look for and verify defects in railway components. Defects that may occur in railway components significantly affect transportation safety. Control methods currently used depend on contact measurement techniques and human perception. This is why traditional auditing is slow, subjective, and inefficient. In order to overcome these disadvantages and limitations, a new autonomous UAV (Unmanned Aerial Vehicle) computer vision-based method is proposed in this study.

The use of Unmanned Aerial Vehicles for the detection of railway defects is a cost-effective approach that offers lower inspection times than traditional techniques. In recent years, UAVs, improvements in processing power, and the shrinking of sensors and their components have increased interest for both military and civilian applications. In particular, the increase in flight times shows that UAVs are at a level that can be used in the field of fault detection. In the thesis, algorithms were developed for autonomous tracking of the railway by the UAV and deep learning-based defect detection was performed on the images obtained by the UAV.

**Keywords:** Unmanned aerial vehicle, Railway maintenance, Flaw detection, Deep learning

# ŞEKİLLER LİSTESİ

|                                                                                                          | Sayfa |
|----------------------------------------------------------------------------------------------------------|-------|
| Şekil 1.1. Demiryolunun genel yapısı .....                                                               | 2     |
| Şekil 1.2. Rayın yapısı.....                                                                             | 3     |
| Şekil 1.3. Balast yatağı .....                                                                           | 3     |
| Şekil 1.4. Travers.....                                                                                  | 4     |
| Şekil 1.5. Bağlantı Elemanı .....                                                                        | 4     |
| Şekil 1.6. (a) Çok rotorlu İHA (b) Sabit kanatlı İHA .....                                               | 6     |
| Şekil 1.7. Çok motorlu bir uçağın koordinat sistemi .....                                                | 7     |
| Şekil 1.8. Pervanelerdeki dönüş hızının drona etkileri .....                                             | 8     |
| Şekil 1.9. Görüntü toplama sistemi .....                                                                 | 9     |
| Şekil 1.10. İHA ile görüntü toplama.....                                                                 | 10    |
| Şekil 1.11. Denetim aracı .....                                                                          | 14    |
| Şekil 1.12. Karayolu raylı aracı.....                                                                    | 14    |
| Şekil 1.13. Hat kayıt aracı .....                                                                        | 15    |
| Şekil 2.1. Otonom ray takibi için blok diyagramı .....                                                   | 16    |
| Şekil 2.2. HSV renk uzayı .....                                                                          | 18    |
| Şekil 2.3. Çizgiye en kısa mesafe ve oluşan açı .....                                                    | 20    |
| Şekil 2.4. Tek nokta ile ifade.....                                                                      | 21    |
| Şekil 2.5. 3 × 3 kare çekirdek görüntü kullanarak açmanın etkisi .....                                   | 21    |
| Şekil 2.6. 3 × 3 kare çekirdek görüntü kullanarak genişletmenin etkisi .....                             | 22    |
| Şekil 2.7. Önerilen yöntemin akış diyagramı .....                                                        | 23    |
| Şekil 2.8. a) Alınan görüntü b) HSV formatlı görüntü .....                                               | 23    |
| Şekil 2.9. a) Eşikleme işlemi sonucu b) Morfolojik işlemler sonucu .....                                 | 24    |
| Şekil 2.10. a) Görüntüye uygulanan kenar çıkarma sonucu b) Hough Dönüşümü .....                          | 24    |
| Şekil 2.11. a) RGB renk uzayı b) HSV renk uzayı.....                                                     | 25    |
| Şekil 2.12. Ray konumlarının histogram tabanlı tespiti a) İkili görüntü b) Histogram pik noktaları ..... | 26    |
| Şekil 2.13. İHA'nın bulunması gereken konum.....                                                         | 27    |
| Şekil 2.14. İHA'nın bulunduğu konum ile istenen konumun belirlenmesi .....                               | 27    |
| Şekil 2.15. Ufuk noktası tahmini için uygulanan algoritma .....                                          | 29    |
| Şekil 2.16. PID kontrol parametrelerini ayarlamak için kullanılan bulanık sistemin blok şeması .....     | 30    |
| Şekil 2.17. Hata (e) ve hatanın türevi (ed) için üyelik fonksiyonları.....                               | 30    |
| Şekil 2.18. a) Kp için üyelik işlevleri b) Ki için üyelik işlevleri c) Kd için üyelik işlevleri.....     | 31    |

|                                                                                                                                                                                                                |    |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Şekil 2.19. Hat segmentasyon ağının mimarisi .....                                                                                                                                                             | 32 |
| Şekil 2.20. Katman yapıları .....                                                                                                                                                                              | 33 |
| Şekil 2.21. Farklı modeller için çizgi sonuçları .....                                                                                                                                                         | 33 |
| Şekil 2.22. COCO veri kümesinde çeşitli örnek segmentasyon yöntemleri için hız-performans karşılaştırması .....                                                                                                | 35 |
| Şekil 2.23. Yolact mimarisi .....                                                                                                                                                                              | 35 |
| Şekil 2.24. Sonuçlar a) Maskelerin elde edilmesi b) İkili maskenin oluşturulması .....                                                                                                                         | 36 |
| Şekil 2.25. Orta nokta ve referans noktasının işaretlenmesi .....                                                                                                                                              | 36 |
| Şekil 2.26. Önerilen yöntemin tasarımı .....                                                                                                                                                                   | 37 |
| Şekil 2.27. Mask R-CNN mimarisi .....                                                                                                                                                                          | 39 |
| Şekil 2.28. Travers eksikliklerinin tespiti için önerilen algoritma .....                                                                                                                                      | 39 |
| Şekil 2.29. a) Mask R-CNN algoritmasının çıktısı b) Ölçüm için sınırların belirlenmesi .....                                                                                                                   | 40 |
| Şekil 2.30. Referans görüntüde iki nokta arası ölçüm sonucu .....                                                                                                                                              | 41 |
| Şekil 2.31. Mask R-CNN sonucu elde edilen görüntü a) Maske, kutu, etiket bilgisi b) İkili maske .....                                                                                                          | 42 |
| Şekil 2.32. x ekseninde piksel yoğunluğu toplamının gösterimi .....                                                                                                                                            | 42 |
| Şekil 2.33. Histogramın yeniden çizilmesi .....                                                                                                                                                                | 43 |
| Şekil 2.34. Önerilen yöntemin akış diyagramı .....                                                                                                                                                             | 44 |
| Şekil 2.35. Bağlantı elemanı görüntüsünde yerel ikili desen .....                                                                                                                                              | 45 |
| Şekil 2.36. Farklı n değerleri için ÇYİÖ uygulaması .....                                                                                                                                                      | 46 |
| Şekil 2.37. Kelime çantası modeli akış diyagramı. ....                                                                                                                                                         | 48 |
| Şekil 2. 38. Kusurlu bağlantı elemanı örnekleri .....                                                                                                                                                          | 50 |
| Şekil 2. 39. Yapılan işlemlerin akış diyagramı .....                                                                                                                                                           | 50 |
| Şekil 2. 40. a) Orijinal görüntü b) Etiketli görüntü .....                                                                                                                                                     | 51 |
| Şekil 2. 41. YOLOv4 ağı giriş çıkış örneği .....                                                                                                                                                               | 51 |
| Şekil 2. 42. YOLOv4 mimarisi .....                                                                                                                                                                             | 52 |
| Şekil 2. 43. Bulanık mantık yapısı .....                                                                                                                                                                       | 53 |
| Şekil 2. 44. Giriş ve çıkış için oluşturulan üyelik fonksiyonları a) Girişler için üyelik fonksiyonları b) Çıkış için üyelik fonksiyonları .....                                                               | 54 |
| Şekil 2. 45. DCGAN Yapısı .....                                                                                                                                                                                | 57 |
| Şekil 2. 46. Eğitim aşamasında devir sayısına göre DCGAN ile üretilen bazı deforme bağlantı elemanı görüntüleri .....                                                                                          | 57 |
| Şekil 2. 47. Oluşturulan deforme veri kümesi a) Görüntü üretimi için kullanılan veri kümesinden bazı deforme bağlantı elemanı görüntüleri b) Eğitim için kullanılmak üzere DCGAN ile üretilen görüntüler ..... | 58 |
| Şekil 2. 48. Siyam sinir ağı yapısı .....                                                                                                                                                                      | 60 |

|                                                                                                                                                                                                |    |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| <b>Şekil 2. 49.</b> Rastgele seçilen bazı görüntü çiftleri .....                                                                                                                               | 60 |
| <b>Şekil 2. 50.</b> Siyam sinir ağı modelinde kullanılan katmanlar ve parametreler .....                                                                                                       | 61 |
| <b>Şekil 2. 51.</b> Rakip örneklerin kullanılması .....                                                                                                                                        | 63 |
| <b>Şekil 3.1.</b> Ray takibi algoritmasının sahada test edilmesi (b) İHA'nın istenen değeri yakalaması.....                                                                                    | 66 |
| <b>Şekil 3.2.</b> a) Demiryolu sahasında ray hattını takip eden İHA'nın orta noktayı yakalaması b) İHA'ya gönderilen dönme değerleri .....                                                     | 67 |
| <b>Şekil 3.3.</b> Sağ ve sol çizgilerin oluşturulması.....                                                                                                                                     | 68 |
| <b>Şekil 3.4.</b> Ufuk noktasının kırmızı nokta ile işaretlenmesi.....                                                                                                                         | 68 |
| <b>Şekil 3.5.</b> Ufuk noktası ile merkez konumun gösterimi .....                                                                                                                              | 69 |
| <b>Şekil 3.6.</b> Referans noktasına göre PID çıkışı .....                                                                                                                                     | 69 |
| <b>Şekil 3.7.</b> Kp'nin değişimi.....                                                                                                                                                         | 70 |
| <b>Şekil 3.8.</b> Ki'nin değişimi.....                                                                                                                                                         | 70 |
| <b>Şekil 3.9.</b> Kd'nin değişimi.....                                                                                                                                                         | 70 |
| <b>Şekil 3.10.</b> Referans değerinin bulanık PID .....                                                                                                                                        | 71 |
| <b>Şekil 3.11.</b> Hata(e) ve hatanın türevinin(ed) sıfıra yakınsaması.....                                                                                                                    | 71 |
| <b>Şekil 3.12.</b> Ufuk noktasının bulanık-PID ile yakalanması.....                                                                                                                            | 72 |
| <b>Şekil 3.13.</b> Ufuk noktasının İHA tarafından yakalanması.....                                                                                                                             | 72 |
| <b>Şekil 3.14.</b> Deney esnasında hatanın (e) ve hatanın türevinin (ed) sıfıra yakınsaması .....                                                                                              | 73 |
| <b>Şekil 3.15.</b> Ray ve traverslerin tespiti.....                                                                                                                                            | 73 |
| <b>Şekil 3.16.</b> İHA'nın istenen pozisyonu yakalaması.....                                                                                                                                   | 74 |
| <b>Şekil 3.17.</b> PID denetleyici sonucu.....                                                                                                                                                 | 74 |
| <b>Şekil 3.18.</b> Test verilerinin sınıflandırılmasından elde edilen karmaşıklık matrisi a) Eşik değeri 70 cm için sonuç b) Eşik değeri 75 cm için sonuç c) Eşik değeri 80 cm için sonuç..... | 76 |
| <b>Şekil 3.19.</b> Otsu Metodu için sınıflandırma sonucu .....                                                                                                                                 | 77 |
| <b>Şekil 3.20.</b> Mask R-CNN için sınıflandırma sonucu .....                                                                                                                                  | 77 |
| <b>Şekil 3.21.</b> Görüntü ön işleme sonucu.....                                                                                                                                               | 78 |
| <b>Şekil 3.22.</b> Görüntü üzerinde ray ve traverslerin tespit edilmesi .....                                                                                                                  | 79 |
| <b>Şekil 3.23.</b> ÇYİÖ uygulaması ve bağlantı elemanı tespiti. ....                                                                                                                           | 79 |
| <b>Şekil 3.24.</b> Farklı özellik çıkarım yöntemlerine göre başarımlar.....                                                                                                                    | 80 |
| <b>Şekil 3. 25.</b> Sağlam bağlantı elemanı için elde edilen sonuçlar a) Giriş görüntüsü, b) YOLOv4 çıktısı, c) Sonuç d) Sonuç grafiği .....                                                   | 83 |
| <b>Şekil 3. 26.</b> Deforme bağlantı elemanı için elde edilen sonuçlar a) Giriş görüntüsü, b) YOLOv4 çıktısı, c) Sonuç d) Sonuç grafiği .....                                                  | 83 |
| <b>Şekil 3. 27.</b> Eksik bağlantı elemanı için elde edilen sonuçlar a) Giriş görüntüsü, b) YOLOv4 çıktısı, c) Sonuç d) Sonuç grafiği .....                                                    | 84 |

|                                                                                                             |    |
|-------------------------------------------------------------------------------------------------------------|----|
| <b>Şekil 3. 28.</b> Deney sonucunda oluşan model doğruluk grafiği .....                                     | 86 |
| <b>Şekil 3. 29.</b> Deney sonucunda oluşan model hata grafiği .....                                         | 86 |
| <b>Şekil 3. 30.</b> Eğitilen siyam ağına göre iki görüntünün benzerlik oranına göre karşılaştırılması ..... | 87 |
| <b>Şekil 3. 31.</b> Test görüntülerinin eşleştirilmesi sonucu oluşan karmaşıklık matrisi .....              | 87 |
| <b>Şekil 3. 32.</b> Gürültü eklenmeden önce sınıflandırma sonuçları .....                                   | 89 |
| <b>Şekil 3. 33.</b> Gürültü eklendikten sonra sınıflandırma sonuçları.....                                  | 89 |
| <b>Şekil 3. 34.</b> Önerilen yöntemin akış diyagramı .....                                                  | 90 |
| <b>Şekil 3. 35.</b> Kullanılan veri kümesinden bazı görüntüler .....                                        | 90 |



## TABLÖLAR LİSTESİ

|                                                                                                         | Sayfa |
|---------------------------------------------------------------------------------------------------------|-------|
| <b>Tablo 2.1.</b> Ray tespiti için eşik değerleri.....                                                  | 25    |
| <b>Tablo 2.2.</b> Kp ve Ki için kurallar .....                                                          | 31    |
| <b>Tablo 2.3.</b> Kd için kurallar .....                                                                | 32    |
| <b>Tablo 2.4.</b> Kurallar.....                                                                         | 54    |
| <b>Tablo 3.1.</b> Diğer ölçütler için performans sonuçları .....                                        | 75    |
| <b>Tablo 3.2.</b> İki farklı travers segmentasyon yöntemi için sonuçların karşılaştırılması .....       | 77    |
| <b>Tablo 3.3.</b> Veri kümesinden rastgele seçilen bağlantı elemanı örnekleri .....                     | 78    |
| <b>Tablo 3.4.</b> Beş ayrı ölçüt için ORB, SURF ve HOG performans değerleri.....                        | 80    |
| <b>Tablo 3.5.</b> Literatürde bulunan diğer yaklaşımlar ile karşılaştırma.....                          | 80    |
| <b>Tablo 3.6.</b> Karmaşıklık matrisi ölçüm değerleri için parametreler .....                           | 84    |
| <b>Tablo 3.7.</b> Bağlantı elemanı kusur tespiti için karmaşıklık matrisi.....                          | 84    |
| <b>Tablo 3.8.</b> Dört ayrı ölçüt için performans değerleri.....                                        | 85    |
| <b>Tablo 3.9.</b> Önerilen yöntemin performans sonuçları.....                                           | 87    |
| <b>Tablo 3.10.</b> Normal veri kümesi için sonuçlar .....                                               | 90    |
| <b>Tablo 3.11.</b> Rakip görüntülerden oluşan veri kümesi için sonuçlar .....                           | 91    |
| <b>Tablo 3.12.</b> Tekrar eğitildikten sonra normal veri kümesi için sonuçlar.....                      | 91    |
| <b>Tablo 3.13.</b> Tekrar eğitildikten sonra rakip görüntülerden oluşan veri kümesi için sonuçlar ..... | 91    |

## SİMGELER VE KISALTMALAR

### Simgeler

---

|            |                                  |
|------------|----------------------------------|
| $L_{cls}$  | : Sınıflandırma kayıp fonksiyonu |
| $L_{reg}$  | : Sınıflandırma kayıp fonksiyonu |
| $L_{mask}$ | : Maske kayıp fonksiyonu         |

### Kısaltmalar

---

|       |                                              |
|-------|----------------------------------------------|
| İHA   | : İnsansız Hava Aracı                        |
| R-CNN | : Region Based Convolutional Neural Networks |
| SIFT  | : Scale-invariant feature transform          |
| UAS   | : İnsansız Uçak Sistemleri                   |
| 3B    | : Üç Boyutlu                                 |
| HSV   | : Hue, Saturation, Value                     |
| ROI   | : Region of Interest                         |
| ÇYİÖ  | : Çizgi Yerel İkili Örüntü                   |

# 1. GİRİŞ

Demiryolu taşımacılığının, yolcu ve yük taşımacılığında önemli bir payı vardır. Bu yönüyle ülkelerin ekonomik kalkınmasında için büyük öneme sahiptir. Demiryollarının, yüksek hızlı trenlerin kullanıldığı bölgelerde hâlâ büyük hız avantajları ile yolcu taşımacılığı için kullanılmaktadır. Ayrıca demiryolları, diğer yük taşıma yöntemleriyle karşılaştırıldığında ham petrol gibi ağır yükleri uzun mesafeler boyunca taşımak için çok uygun maliyetli bir yol sağlar. Demiryollarının bu avantajları, birçok ülkede raylı sistemlere yapılan yatırımların artmasını sağlamıştır. Yapılan yatırımlarda mevcut sistemlerin arızalanmasını engellemek için alınan önlemlerin yeri oldukça fazladır. Demiryolu şirketleri, her yıl kaza sayısını azaltmak için hat bakımına oldukça fazla para harcamaktadır. Demiryolu altyapısı yoğun bir şekilde baskıya ve yanal kuvvete maruz kalmaktadır. Bu baskıların oluşturabileceği arızaların erken tespiti için raylı sistemler, etkin ve düzenli olarak izlenmeli ve değerlendirilmelidir. Demiryolu altyapısının zamanında denetlenmesi ve önleyici bakım, demiryolu altyapısını güvenli ve çalışır durumda tutmak için iki ana görev olarak kabul edilmektedir.

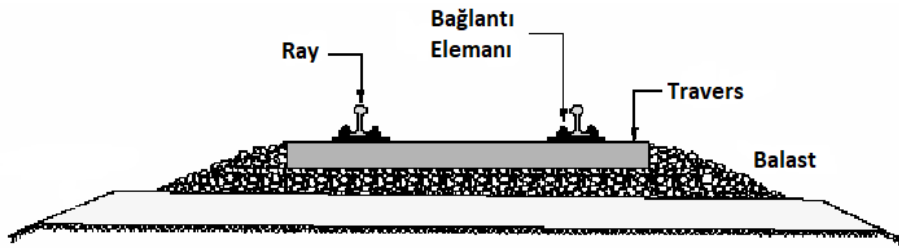
Ray denetiminin geçmişi, denetimin görsel olarak gerçekleştirildiği yüz yıldan daha eskiye dayanmaktadır. Geçmişten beri, demiryolu altyapısının durumunu denetlemek veya demiryolu kusurlarına karşı önlem almak için çeşitli denetim yöntemleri kullanılmaktadır. Demiryolları farklı bileşenlerden oluşur. Demiryolu bileşenlerinden birinin bile arızalanması diğer bileşenler için yıkıcı sonuçlar oluşturur. Bu nedenle, bakım ve denetim çalışmalarının sık yapılması sistemin uzun vadeli çalışmasını sağlar. Ancak, günümüzde bakım ve denetim çalışmaları sıklıkla yapılamamaktadır. Bunun sebebi, denetimi gerçekleştirmek için gerekli ekipmanın kullanılmasının pahalı olması gibi nedenlerden kaynaklanmaktadır. Bu nedenle, demiryolu bağlantı elemanlarını tespit etmek için otomatik bir yöntem acilen ihtiyaç duyulmaktadır.

Demiryolu denetimi için alternatif yöntemler üzerinde çalışılmaktadır. Özellikle, yüksek hızlı trenlerin gelişmesiyle ve demiryolu ağının genişlemesiyle birlikte demiryolu bakımı eskiye göre daha zor hale gelmiştir. Genellikle denetim, kusurları tespit etmek için demiryolu ağı boyunca yürüyen eğitimli çalışanlar tarafından gerçekleştirilir. Ancak bu denetim biçimi yavaştır ve denetleme yapan elemanın olası anormallikleri tespit etme yeteneğine bağlıdır. Günümüzde kullanılan diğer yöntem ise temaslı ölçüm teknikleridir. Ancak bu teknikler, düşük hassasiyet ve düşük doğruluk oranı ile sonuçlanmaktadır. Ayrıca, kullanılan mevcut tekniklerin erken tespit oranı, güvenilirliği düşüktür ve tespit sistemlerinin maliyetleri fazladır. Otomatik ve temassız denetim sistemlerinin geliştirilmesi, temaslı incelemenin dezavantajlarının üstesinden gelebilmek için önemlidir. Bu nedenle, otomatik ve temassız denetim yöntemleri acilen geliştirilmelidir. Bu amaçla, kusurların olabilecek en erken biçimde tespit edilmesi için etkili ve hızlı tahribatsız yöntemler geliştirmek için çok sayıda araştırma yapılmaktadır.

Tahribatsız yöntemler, denetlenen malzemeye zarar vermeden malzemelerin ve yapıların tekdüzeliği, kalitesi ve hizmet verilebilirliğinin tüm yönlerinin incelenmesiyle ilgili çeşitli yöntem ve prosedürlerin genel adıdır. Tahribatsız testler, bileşenlere temas etmeden ve gelecekteki kullanışlılığını etkilemeden gerçekleştirilen yöntemdir. Temassız incelemede özellikle kullanılan en yeni yöntemlerden bazıları bilgisayarla görüye [1] dayanırken, farklı durumlarda darbe akustik emisyonları [2], ultrason [3] ve girdap akımı [4] gibi diğer otomatik inceleme yöntemleri kullanılabilir. Bu tür denetimler, genellikle kalifiye personel tarafından yapılır.

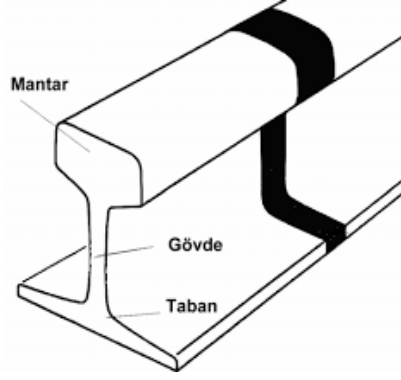
Otomatik ve temassız denetim sistemlerinde demiryolundan görüntü toplanması önemli bir görevdir. Bu görev için genellikle bir tren altına kamera ve ışık kaynakları yerleştirilerek görüntü toplanır veya görüntü toplamak için demiryolunda ilerleyebilen bir tekerlekli araç yapılabilir [5]. Son yıllarda İHA'ların gelişmesiyle demiryolunu denetleme ve görüntü toplama görevleri İHA'lar ile yapılmaya başlanmıştır. Özellikle, kusur tespiti uygulamalarında otonom İHA'lar için uygulamaların araştırılması ve geliştirilmesinde muazzam bir artış görülmektedir. İHA'lar üzerindeki sensörlerin gelişmesiyle otomatik denetim uygulamalarında İHA tabanlı çözümlerin önemli bir pazar payı kazanması beklenilmektedir. Bu tez, demiryolu altyapı denetimi için İHA tabanlı, sürdürülebilir ve düşük maliyetli bir çözüm oluşturmaya katkıda bulunmaktadır. Bu çözümle, altyapı denetimi sırasında zamandan tasarruf etmeyi ve operatörlerin ve inşaat işçilerinin güvenliğini artırmayı umuyoruz. Sürekli altyapı bakımının sağlanması, hasarlı yapılardan kaynaklanan kazaları önleyecek ve kamu güvenliğini sağlayacaktır.

Demiryolları, Şekil 1.1'de gösterildiği gibi ray, bağlantı elemanları, traversler ve balasttan oluşur. Ray, trenleri destekleyen demiryolu bileşenlerinden biridir. Trenin güvenli bir şekilde hareket edebilmesi için ölçüsü, eğimi, çapraz seviyesi ve rayların hizalaması belirli sınırlar içinde tutmalıdır. Bunlar hattın kritik geometri parametreleridir.



Şekil 1.1. Demiryolunun genel yapısı

Raylar, bir demiryolunu oluşturan temel unsurdur. Tren tekerleklerinin temas ettiği ilk elemanlardır. Bu nedenle trenleri destekler ve yönlendirirler. Ray parçaları, travers adı verilen bir yatak yapısının üzerine paralel olarak yerleştirilmiştir. En çok kullanılan profil, Şekil 1.2'de gösterilen profildir. Karakteristik şekli, yüksek bir dikey atalet ve kesme gerilmelerinin uygun değerlerde dağılımını sağlar. Temel özelliği, lineer metre başına kütlesidir.



**Şekil 1.2.** Rayın yapısı

Hat, bir demiryolu hattında kalıcı yoldur. Balast, "traverslere veya taşıyıcılara dikey ve yanal destek sağlamak alt zemin üzerine yerleştirilmiş, belirtilen özelliklere sahip tek boyutlu granüler bir malzemedir. Balast yatağı, gevşek iri taneli malzeme tabakasından oluşur (Şekil 1.3).



**Şekil 1.3.** Balast yatağı

Balastın temel amacı, doğrudan yer ile temas etmesi gereken ağır yapılar için sağlam ve sabit bir destek tabanı sağlamaktır. Topraklar; ıslak, verimli ve bitki örtüsü açısından zengin olabilmeleri ve özellikle sel, rüzgâr vb. nedenlerle önemli değişikliklere uğrayabilirler. Bu da zeminin dayanıklılığını azaltır. Bu nedenle, yapıları doğrudan desteklemek için genellikle uygun değildir. Küçük ve orta boy taşlardan oluşan bileşimi sayesinde balast, yapıyı nemden korur ve suyun altından geçmesine izin verir, böylece daha sonra buharlaşabilir ve sağlam bir yapı elde edilmiş olur.

Traversler, demiryolu altyapısı için önemli bileşenlerdir. Demiryolunda kullanılan traversler, rayın geometrisini korumak için sağ ve sol ray arasında bulunur (Şekil 1.4). Ray ve ray sabitlemesine dikey ve yanal destek sağlayan enine bir kiriştir [6]. Demiryolu altyapısında beton, plastik, ahşap, çelik olmak üzere dört ayrı travers kullanılır. Çoğunlukla tahta (kereste) ve beton traversler kullanılır. Günümüzde, bazı ülkelerde ahşap traverslerin yerini çoğunlukla beton

traversler almıştır. İki travers arasındaki mesafe genellikle 60 cm'dir. Ray, traverslere rayın dönmesini ve hareketini önlemek için kullanılan tespitlerle sabitlenir.



**Şekil 1.4.** Travers

Bağlantı elemanları, çelik rayları ve demiryolu traverslerini birleştirmek için kullanılan ray bileşenleridir [7]. İşlevleri, demiryolu traverslerine çelik rayları sabitlemek, ölçüyü korumak ve rayların traverslere göre uzunlamasına ve yanlamasına hareket etmesini önlemektir. Bağlantı elemanları, çelik rayın demiryolu traversine sabitlenmesinde rol oynayan hayati bir ray bağlantı elemanıdır. Bağlantı elemanının fonksiyonları şu şekildedir; kenetleme kuvveti üretir ve bunu ray hattına iletir, iki ray arasındaki güvenilir bağlantıyı korur, demiryolu hattının bütünlüğünü mümkün olduğunca korur. Ayrıca bağlantı elemanları, lokomotifin ürettiği darbeyi emebilir.

Ray ve travers arasındaki bağlantı yıllar içinde çeşitli değişikliklere uğramıştır. Başlangıçta, rayı sabitlemek için, çift raylı bir ray için tahta bir kama kullanıldı ve tekerleklerle temas eden parça aşındığında, ray aşınmayan kısmı kullanmak için döndürülebilirdi. Günümüzde bağlantı sistemi, iyi esneklik ve daha hızlı süreler sağlamak için mekanize edilmiştir. Sabitleme, travers ile rayı otomatik olarak birbirine bağlayan mekanik makinelerle yapılır. Şekil 1.5'te bir bağlantı elemanı örneği gösterilmiştir.



**Şekil 1.5.** Bağlantı Elemanı

İnsansız Uçak Sistemleri (UAS) olarak da bilinen insansız hava araçlarının rolü, modern yüksek teknoloji toplumunda giderek artmaktadır. İHA'lar, ticari ve özel alanda sıklıkla kullanılmaktadır. Bu nedenle, İHA'lar için geliştirici ve üretici sayısındaki keskin artış görülmektedir. Hızla gelişen pazar göz önüne alındığında, İHA teknolojisi önümüzdeki yıllarda hızla gelişecek ve daha düşük maliyetler ortaya çıkacaktır. Bu nedenle, insansız hava aracı teknolojisine daha geniş erişim sağlanacaktır. Uluslararası alanda, İnsansız hava sistemlerini araştırmaya yönelik büyük bir ilgi vardır. Bu ilgi, İHA'ların gelişmesine kullanılabileceği yeni uygulama alanlarının ortaya çıkmasına neden olmaktadır.

Günümüze kadar insansız hava araçlarının çoğunlukla askeri alanda kullanımları vardı. Genellikle gözetleme ve dinleme görevleri için kullanılırdı. Ancak son yıllarda İHA'lar maliyetindeki düşüş ve teknolojik gelişmeler sayesinde sivil alanda da yaygın olarak kullanılmaktadır. Mevcut büyüme rakamları sivil kullanım için muazzam bir potansiyele işaret etmektedir. Üzerlerinde pilot bulundurmamalarından dolayı insanlar için tehlikeli olabilecek görevlerde sıklıkla kullanılırlar. Gelecekte İHA kullanımının, katlanarak büyümeye devam etmesi beklenmektedir. İHA'ların uygulamaları hızla artıyor. Günümüzde ticari olarak tarım, anket, hayvan gözlemi, emlak ve diğer birçok alanda kullanılmaktadır. İHA'lar bu kullanımların yanı sıra video çekimi, haritacılık, 3d modelleme gibi alanlarda da kullanılır. Ayrıca, petrol ve gaz gibi endüstriler izleme ve bakım faaliyetlerinde kullanılır. İHA'lar irtifa aralığı, dayanıklılık ve ağırlığa göre sınıflandırılır ve askeri ve ticari uygulamalar dâhil olmak üzere çok çeşitli uygulamaları destekler.

### **1.1. İHA Türleri**

İHA'ların sınıflandırılmaları, yapılarına göre yapılmaktadır. Buradaki ana ayırt edici özellik, rotorların düzeni ve gövdenin yapısıdır. İnsansız hava araçlarının klasik temel formları, kanatlar ve rotorlardan oluşur. İHA'lar görev ihtiyaçlarına, kullanım alanlarına veya taşıyacakları yük kapasitesine göre farklı tiplerde tasarlanmaktadır. İHA'lar, uçuş menzili ve manevra kabiliyeti açısından sınıflandırılırlar. Kullanıcılar seçimlerini İHA'nın kullanılacağı göreve göre seçmektedirler. İHA'lar, uçuş mimarisine göre iki ana kategoriye ayrılabilir. Bunlardan birincisi rotor tabanlı İHA'lardır (Şekil 1.6.a). Rotor tabanlı İHA'lar; küçük, hafif ve kontrolü kolaydır. Rotor tabanlı İHA'lar, dikey ve aynı zamanda yatay olarak uçuş kabiliyetini sağlar. Piyasada dört rotorlu (quadrocopter), altı rotorlu (hexacopters), sekiz rotorlu (octocopters) veya daha fazla rotorlu sistemler bulunabilir. Bunlar arasından quadrocopter en yaygın olanıdır. Tüm türler için kullanılan multicopter terimi, bir düzlemde ikiden fazla rotora sahip tüm türleri içerir. Bu tür bir yapı, helikopter gibi klasik bir döner kanatlı hava aracına çok benzer. İkinci tür ise sabit kanatlı İHA'lardır (Şekil 1.6.b). Sabit kanatlı İHA olarak adlandırılan İHA, kanatlar üzerindeki hava akışı ile birlikte kaldırma ve yatay uçuş yönü sağlayan dikey olarak düzenlenmiş bir pervane tarafından

hareket ettirilir. Sabit kanatlı İHA'nın yapısı klasik bir uçağa benzer. Sabit kanatlı İHA'lar daha yüksek irtifada uçar, daha hızlıdır ve ağır yükler taşıyabilirler. Ancak, sabit kanatlı İHA'lar ileri yönde hareket etmeden uçamazlar. Bu nedenle havada asılı kalamazlar. Ayrıca, sabit kanatlı İHA'lar kalkış ve iniş için pistlere veya geniş açık alanlara ihtiyaç duyar. Tercih edilme sebepleri, yük taşıma kapasitelerinin fazla olması ve uzun süre havada kalabilmeleridir.



(a)



(b)

**Şekil 1.6.** (a) Çok rotorlu İHA (b) Sabit kanatlı İHA

Rotor tabanlı İHA'lar, düşük fiyatları ve kullanım kolaylığı sayesinde en yaygın olanıdır. Bu cihazlar, üzerinde bulunan sensörler sayesinde görsel görüntüleme gibi görevleri yapabilecek kapasitededir. Görsel görüntülemede İHA'ların kullanılmasının birçok avantajı vardır. Bu avantajlar şunları içerir:

- Zaman tasarrufu ve iş gücü verimliliği: İHA'lar, görsel inceleme gibi görevleri, geleneksel yaklaşımlardan daha hızlı ve daha ucuz bir şekilde tamamlayabilir;
- Artan erişilebilirlik ve güvenlik: İHA'lar, ulaşması zor olan veya güvenlik riskleri oluşturan yerlere risk oluşturmadan erişim sağlayabilir;
- Maliyet ve taşıma: maliyetleri ve düşük nakliye maliyetleri nedeniyle, boyutlarının küçük olmasından dolayı ihtiyaç duyulduğunda farklı bölgelere sevk edilebilir;
- Gelişmiş veri toplama: İHA'lar, verilerin eşzamanlı olarak ve daha hızlı bir şekilde toplanmasına izin verir. Bu durum, arızalar meydana gelmeden önce müdahalenin gerçekleştirilmesine olanak sağlar;
- Doğruluk: Görsel inceleme yapılan materyalden İHA görüntüleme yoluyla alınan ölçümler, manuel yöntemlere göre gelişmiş doğruluk sağlayabilir.

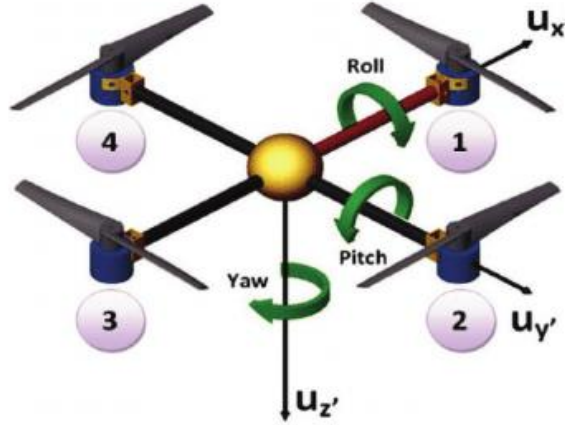
## 1.2. Çok Rotorlu İHA'lar

Çok rotorlu İHA'lar, hobi amaçlı kullanım veya profesyonel kullanımda en çok tercih edilen tasarımıdır. Çok rotorlu İHA'lar, dikey kalkış ve iniş, havada asılı kalma ve mekanik basitlik gibi özellikleri nedeniyle yaygın olarak kullanılmaktadır. Geleneksel çok motorlu İHA tasarımları, maksimum uçuş süresi için optimize edilmiştir. Bu İHA'ların, genellikle üzerlerinde kamera

bulunur. Çok rotorlu tasarımlarda uçuş prensibi nedeniyle İHA'yı kontrol eden kişi, İHA üzerinde çok iyi kontrole sahip olabilir. Bu İHA'lar, stabiliteleri ve manevra kabiliyetleri ile bilinir. Farklı kontrolörler ve otomatik pilotlar mevcuttur ve dronlar birçok şekil ve boyutta gelir.

Çok rotorlu İHA'lar, helikopterler ile benzer prensiplerle çalışır. İnsanlı bir helikopter, havayı aşağı doğru iterek ve havada asılı kalır. Bu işlemi gerçekleştirmek için büyük rotor kanatları kullanır. Rotorun yüksek dönme enerjisi gövdede bir tork yaratır. Ayrıca, helikopteri hizalı tutmak için uçağın kuyruğunda da küçük bir rotor bulunur. Çok rotorlu İHA'da ise hareketli tek parça yatay olarak yönlendirilmiş sabit rotorlardır. Bu parçalar dönerken havayı helikopter gibi aşağı doğru iterler. Havada manevra yapmak için ise, her bir rotorun hızı, İHA'yı hedeflenen biçimde yönlendirecek şekilde ayarlanır.

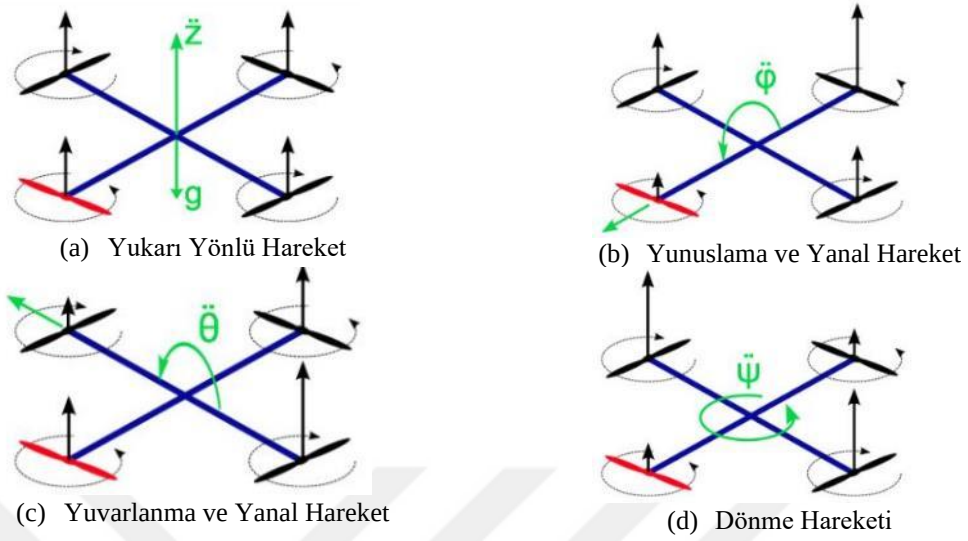
Multirotor İHA'ların koordinat sistemleri için kullanılan sistem, Şekil 1.7'de gösterildiği gibi 3 eksen etrafındaki yönelim ile tanımlanır. Bunlar; yunuslama(pitch), yuvarlanma(roll) ve dönmedir(yaw) [8]. Yuvarlanma(roll), Sağ ve sol tarafın birbirine göre konumunu ayarlar. Sağ taraf aşağı, sol taraf yukarı doğru itilirse İHA sağa hareket edecektir. Ters durumda ise sola doğru hareket edecektir. Yunuslama (pitch), Ön ve arka tarafın birbirine göre konumunu ayarlar. Öne doğru hareket için dronun ön tarafı aşağı, arka tarafı yukarı doğru hareket ettirilmelidir. Dönme (yaw), dronun dönüşünü ifade eder. Ön, arka ve her iki taraf düzeltilirse, konum değiştirmeden sapma eksenini etrafında dönebilir.



Şekil 1.7. Çok motorlu bir uçağın koordinat sistemi

Çok motorlu pervane konfigürasyonu genellikle dört pervaneden oluşur. Bu sistem quadcopter olarak adlandırılır. Diğer çok motorlu İHA'lar farklı sayıda pervaneden de oluşabilir. Bunlar pervane sayısına göre; tricopter, hexacopter ve octocopter olarak adlandırılır. Pervane sayısındaki bu farklılık; uçuş süresi, çeviklik, stabilite ve yük kapasitesi açısından performans farklılıklarına sahiptir [9]. Pervaneler, dronu yukarıya kaldırmak için yeterli kaldırma kuvveti oluştururlar. Aynı şekilde ileri ve geri hareketi sağlamak için itme gücü sağlarlar. Dronun hareketi

genel olarak pervanelerin farklı hızda dönmeleri ile sağlanır. Şekil 1.8, pervanelerin dönüş hızına göre dronun hareketinin değişimini göstermektedir [10].



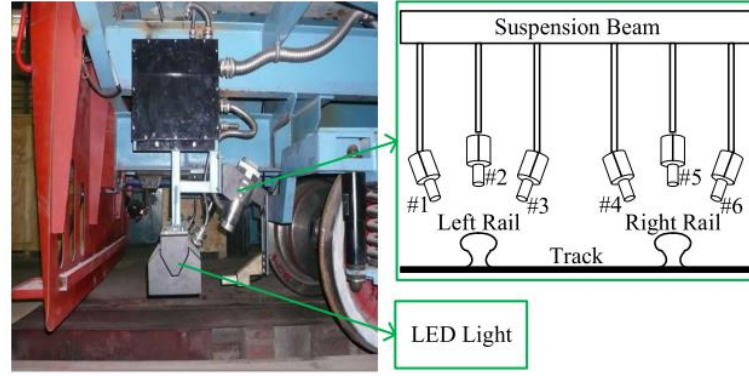
Şekil 1.8. Pervanelerdeki dönüş hızının drona etkileri

### 1.3. Literatür Özeti

Demiryolu bileşenlerinin sağlık durumu, demiryolu taşımacılığının güvenli ve istikrarlı çalışmasını sağlamak için kritik öneme sahiptir. Bu nedenle, demiryolu güvenliğinin sağlanması için sık sık demiryolu denetimi yapmak önemli bir görev haline gelmiştir. Demiryolu bileşenlerinde, tren tekerlekleri ve ray arasındaki temas sürtünmesi, ağırlık ve oluşan titreşim etkisiyle, çeşitli kusurların meydana gelmesi muhtemeldir. Oluşan kusurların tespiti için temassız ve otomatik demiryolu denetimi için çeşitli araştırmalar yapılmıştır.

Aytekin ve arkadaşları [11], yüksek hızlı lazer mesafe bulucu kamera ile gerçek zamanlı bir demiryolu bağlantı elemanı tespit sistemi sunmuşlardır. Önerilen algoritma, yüksek hızlı 3 boyutlu bir kamera demiryolu denetimi için tasarlanmış bir vagonunun altına yerleştirilerek gerçek zamanlı olarak çalıştırılmıştır.

Liu ve arkadaşları [12], çok sayıda bağlantı elemanı örneğini otomatik olarak toplayabilecek ve not ekleyebilecek sistem için çevrimiçi şablon eşleştirme tabanlı sınıflandırma yöntemini önermişlerdir. Uygulamada, dengesiz veri kümesi sorununu çözmek için benzerlik tabanlı bir derin ağ kullanılmıştır. Demiryolu görüntüsü, bir denetim treninin altına monte edilen altı adet DALSA Spyder2 S2 1x-01k40 hat tarama kamerası tarafından yakalanmıştır. Şekil 1.9, görüntü alma cihazının konfigürasyonunu göstermektedir [12].



Şekil 1.9. Görüntü toplama sistemi

Zhang ve arkadaşları [13], görüntü toplama ve görüntü işleme olmak üzere iki adımlı bir kusur tespit yöntemi önermişlerdir. Çalışmada, görüntü elde etmek için rayın üzerine bir kamera yerleştirilip kameranın her iki tarafına ışık kaynağı cihazları eklemiştir. Böylece tren ray üzerinde ilerlerken görüntü alımı sağlanmıştır. Ardından, daha iyi algılama gereksinimleri elde etmek için iki tek aşamalı derin öğrenme algoritmasını birleştirmişlerdir.

Yang ve arkadaşları [14], kurdukları görüntü toplama yöntemi ile ray yüzeyinden görüntü toplamışlardır. Çalışmada, ray yüzeyi ile arka plan arasındaki kontrastı arttırmak için kameranın etrafına ışık kaynakları eklenmiştir. Görüntü toplam işleminden sonra kenar piksellerinin standart sapmasının kararlılığı tabanlı ray çıkarımı yapılmıştır. Ardından, diferansiyel kutu sayımı (DBC) ve GrabCut algoritması, karmaşık kusurların tespit doğruluğunu artırmak üzere kusur segmentasyonu için birleştirilmiştir. Son aşamada, kusur tespiti için YOLOv2 algoritması kullanılmıştır.

Ziukun Wei ve diğerleri [15]'te, görüntü işlemeye dayalı geleneksel bağlantı elemanı konumlandırma yöntemini uygulamıştır. Ek olarak, Dense-SIFT özelliklerini kullanan yeni bir bağlantı elemanı kusur saptama ve tanımlama yöntemi önerilmektedir. Çalışmanın devamında VGG16, bağlantı elemanı kusur tespiti ve tespiti için eğitilmiştir. Sonuç, bağlantı elemanlarının kusur tespitini CNN ile gerçekleştirmenin mümkün olduğunu göstermektedir. Çalışmanın son aşamasında, daha hızlı R-CNN, tespit oranını ve verimliliğini artırmak için bağlantı elemanı kusur tespiti için kullanılmıştır.

Junwen Chen ve diğerleri [16]'da, bağlantı elemanlarının kusur tespitinde derin evrişimli sinir ağlarını uygulayan yeni bir görmeye dayalı yöntem önermektedir.

Ashwani Kumar Dubey ve arkadaşları [17]'de, demiryolu hatlarında kusurlu kısmın tespiti, işaretlenmesi ve görselleştirilmesi için yeni bir görsel inceleme tekniği sunmaktadır. Çalışmada, toplanan demiryolu hattı görüntülerinde ray yüzeyindeki hasarlı bölgelerin geometrik özelliklerini tanımlamak ve görselleştirmek için maksimum stabil uç bölge tekniği kullanılmıştır.

[18]'de, görüntü işleme ve analiz yöntemleri kullanılarak otomatik demiryolu hattı inceleme algoritması sunulmuştur. Çalışmada, eksik bağlantı elemanlarını bulma sorununu ele alınmıştır.

[19]'da Ying Li ve arkadaşları, 20 fps kare hızında 16 km / sa hızda denetimler gerçekleştiren gerçek zamanlı otomatik görmeye dayalı bir ray denetim sistemi sunmuştur. Sistem; bağlantı elemanları gibi önemli ray bileşenlerini yüksek doğruluk ve verimlilikle sağlam bir şekilde algılamaktadır. [20]'de, kısmen aşınmış veya tamamen eksik bağlantı elemanlarını tespit etmek için bir yöntem önerilmiştir. Önerilen yöntem, orijinal test bağlantı elemanı görüntüsünün birinci ve ikinci simetri örneğinden yararlanıp bunları, gelişmiş temsile dayalı bağlantı elemanı tanıma için entegre etmektedir. Xavier Gibert ve diğerleri [21]'de, çok görevli bir öğrenme çerçevesi içinde birden çok dedektörü birleştirerek algılama performansının iyileştirilebileceğini göstermişlerdir. Bu yaklaşımın, demiryolu bağları ve bağlantı elemanlarındaki kusurları tespit etmek için geliştirilmiş doğruluk ile sonuçlandığı gösterilmiştir.

Son yıllarda, İHA'ların gelişmesiyle demiryolunu denetleme ve görüntü toplama görevleri İHA'lar ile yapılmaya başlanmıştır. İHA'lar ile denetleme yapabilmek için İHA'ların hat üzerinde hareket ettirilmesi gerekir. Bu nedenle ray takip algoritması kullanılmalıdır. İHA'lar çeşitli takip uygulamalarında kullanılabilir [22][23]. Bu nedenle ray takibi için de kullanıma uygundur [24].

Wu ve arkadaşları [25], Şekil 1.10'da gösterildiği gibi, üzerinde kamera bulunan bir İHA ile rayların görüntü veya videolarını toplamışlardır. Ardından, toplanan ray görüntülerini analiz etmek için özelleştirilmiş bir görüntü işleme yazılımı kullanılmıştır.



**Şekil 1.10.** İHA ile görüntü toplama

Bojarczak ve arkadaşları [26], demiryolu teşhisi için İnsansız Hava Araçlarından (İHA'lar) alınan görüntüleri kullanmışlardır. Çalışmada, rayı görüntü segmentasyonu ile çıkarmak için Tensorflow ortamında uygulanan FCN-8 derin öğrenme ağını kullanılmıştır.

[27]'de, sabit kanatlı bir İHA için yerleşik görüş tabanlı bir yol izleme sisteminin geliştirilmesi sunulmaktadır. Uygulamada demiryolu takibi üzerinde çalışılmıştır. Bu göreve uygunlukları açısından üç yol izleme kontrolörü incelenmiştir. [28]'de, İHA'ların demiryollarında kullanımını ve demiryolu altyapısının bilgisayarla görüş tabanlı izlenmesini araştırmaktadır. Demiryollarında oluşabilecek kusurların tespiti için dronların kullanılması için uygun maliyetli ve güvenilir görsel denetim üzerinde durulmuştur. Drone üzerine yerleştirilen kamera ile rayların ve demiryolu altyapısının görüntüleri alınıp bu görüntülerde rayların belirlenmesi için kenar ve özellik çıkarma yöntemleri uygulanmaktadır. [29]'da, İHA görüntülerinden ray hatlarının tespiti ve 3 boyutlu modellenmesi incelenmiştir.

Quadrotorlar için kontrol yöntemleri, iyi belgelenmiş çalışmalarla popüler bir araştırma konusu olmuştur. Bu çalışmada PID denetleyici kullanılmıştır. Yapılan denemelerde PID denetleyici kullanmanın düzgün uçuş dengesi ve güvenilir kontrol gösterdiği sonucuna varılmıştır [30,31,32]. PID denetleyiciler, endüstride yaygın olarak kullanılmaktadır.

#### **1.4. Demiryolu Bakımında Kullanılan Teknolojiler**

Hâlihazırda, demiryolu taşımacılığı genellikle ya eğitimli bir insan tarafından ya da manuel makineler tarafından denetlenmektedir. İnsan denetimciler tarafından yapıldığında, denetim yapan kişi demiryolu boyunca yürür ve ölçümleri manuel olarak yapar [33-34]. Demiryolu altyapısı, geniş bir alanı kaplar ve genellikle bakımı zor olan birçok alt bileşenden oluşur. Bu nedenle, manuel denetim yoğun emek gerektirir ve oldukça fazla zaman gerektirir. Ayrıca, denetim sonuçları, büyük ölçüde denetim personelinin yeterliliğine ve kişisel deneyimine bağlı olan öznelir [35]. Son yıllarda, demiryolu denetimi için kullanılan çeşitli araçlar, yavaş yavaş insan denetçilerinin yerini almaktadır. Demiryolu denetim araçlarındaki son gelişmeler, denetim hızını artırmıştır. Raylar, köprüler, tüneller ve katenerlerin denetimi için kullanılmak üzere çeşitli araçlar kullanılabilir. Denetimde muayene araçları kullanılırken, demiryolu muayenesi tahribatsız muayene yöntemleri kullanılarak gerçekleştirilir. Bu sayede muayene prosedürü daha standart ve objektif olabilir. Bununla birlikte, muayene araçlarının kullanılmasına rağmen bazı denetim teknikleri hala zaman alıcıdır. Ayrıca, muayene araçlarını satın alma ve çalıştırma maliyeti yüksektir. Bu nedenle, muayene arabalarını kullanarak ray denetimi yapma sıklığı genellikle düşüktür.

##### **1.4.1. Sensörler**

Sensörlerin ve veri işleme teknolojisinin gelişmesiyle birlikte endüstriyel araçlar, tehlikeli ve zor işlerde kullanılmaya başlanmıştır. Demiryolu denetimi için çeşitli sensörler kullanılmaktadır. Bunlar arasında görsel sensörler [36-37], Dijital Görüntü Korelasyonu (DIC) [38], ultrasonik sensörler [39-40], lazer görüş sistemleri [41], Işık Algılama ve Menzil (LiDAR)

sensörleri [42-43], ivmeölçer [44], termal sensör [45] gibi sensörler ve yöntemler bulunmaktadır. Kullanılacak sensör türü, algılama hedefine, denetlenebilecek kusur türüne göre değişebilmektedir. Seçim, kullanılacak yönteminin avantajlarına ve sınırlamalarına göre yapılmaktadır.

### **Görüntü Sensörleri**

Görüntü sensörleri, yüksek hızlı ve yüksek çözünürlüklü kameralardır. Bu kameralar, endüstriyel süreçleri ve kararları otomatikleştirmek, görüntüleri kaydetmek, malzemenin kalitesini gözlemlemek için kullanılır. Kameralar genellikle ray kusuru, bağlantı elemanı eksiklikleri ve katener hasarı gibi anormallikleri belirlemek için kullanılırlar. Alınan görüntünün işlendiği algoritma, algılama hızını ve doğruluğunu belirleyen önemli bir parçadır.

### **LiDAR**

LiDAR, bir nesnenin dünya yüzeyinden değişken mesafelerini hesaplamak için kullanılan optik bir lazer ölçüm teknolojisidir. Havadaki sistem tarafından toplanan bilgilerle bir araya getirilen bu ışık darbeleri, 3B temsilleri olan nesnelerin yüksek çözünürlüklü haritalarını üretir. Demiryolu denetiminde, genellikle ray açıklığını kontrol etmek için kullanılır.

### **Ultrasonik Sensörler**

Ultrasonik sensörler, genellikle raylardaki çatlakları algılamak için kullanılırlar. Rayın farklı bölümlerini algılamak için farklı derecelerde ultrasonik dalgalar oluşturularak B tarama görüntüleri olarak adlandırılan dalga görüntüleri üretirler. B-tarama görüntü analizi, ray denetimi için sıklıkla kullanılır. Ancak bu yöntemde analiz zaman alıcıdır ve denetim yapan kişinin deneyimine bağlıdır.

### **Termal Sensörler**

Bir sistemin veya alanın sıcaklığını ölçen bir cihaza termal sensör denir. Termal sensörler, anormal ısı modellerini gösteren kızılötesi radyasyon kullanarak görüntüler üretir. Demiryollarında genellikle katenerin durumunu ölçmek için kullanılır. Termal kameralarla görüntü alındıktan sonra sinyal işleme kullanılarak kusur tespiti yapılır.

### **Lazer Mesafe Ölçerler**

Lazer mesafe ölçerler genellikle ışının yansıma süresi ilkesine göre çalışır. Lazer projektör, ray yüzeyine doğru dar bir ışın ışığı yayar ve darbenin ray yüzeyinden yansıyıp sensöre geri dönmesi için geçen süreyi ölçer. Demiryolu hattının geometri parametreleri lazer mesafe ölçerler kullanılarak ölçülebilir.

### **Girdap Akımı**

Girdap akımları, bir akımdaki dönen girdaplar gibi iletkenlerde dolaşan akımlardır. Bu yöntem, manuel olarak veya trenlere monte edilerek raylardaki çatlakları tespit etmek için kullanılır.

#### **1.4.2. Denetim Platformları**

Demiryolu denetim platformları genellikle ray üzerinde hareket etse de sabit olan platformlar da vardır. Sabit platformlar hattın belirli yerine sabitlenir. Amaç, geçen trenlerin tekerlek yükü gibi bilgilerinin araştırılmasıdır. Hareketli platformlar ise demiryolunu gezinerek bilgi toplar ve demiryolunu yüksek hızda incelemek için tasarlanmıştır. Örneğin, İHA'lar, geniş bir bölgede kilometrelerce alanı inceleyebilir. Hareketli platformlarda kullanılan sensörler geliştikçe ve boyut olarak küçüldükçe daha fazla sensörün tek platformda kullanılması sağlanır. Böylece hem denetim hızı artar hem de daha fazla bileşen incelenir.

Yaygın olarak bulunan hareketli muayene üniteleri, insansız hava araçları, denetim arabası, karayolu raylı aracı ve hat kayıt aracı dâhil olmak üzere muayene yöntemlerine göre kategorize edilebilir.

#### **İnsansız Hava Araçları (İHA)**

İHA'lar, üzerinde denetim için çeşitli sensörleri bulunduran bir uçan platformdur. Uçuş yer istasyonu tarafından uzaktan kontrol edilir. Yol programlama veya otonom kontrol ile insan müdahalesini en az gerektirecek şekilde kontrol edilebilir. İHA'lar, kullanımının kolay olması, hareket kabiliyeti ve yüksek hız avantajları nedeniyle demiryolu denetimi için önemli bir araç haline gelmiştir. İHA'ların kullanımıyla insan denetleyiciler tehlikeli yerlerden uzak durabilir ve böylece riskleri azaltabilir. İHA tabanlı bir demiryolu denetimi için kurulan deney ortamı, Şekil 1.10'da gösterilmektedir.

İHA'lar, demiryolundan görüntüleri toplar ve analiz için verileri uzaktan kontrol merkezlerine gönderir. Veriler genellikle İHA'lara entegre edilen termal sensörler ve yüksek çözünürlüklü kameralar tarafından toplanır.

#### **Denetim Arabası**

Denetim arabası, üzerinde ölçüm sistemlerini ve sensörleri bulunduran platformdur. Denetim arabaları, raylar üzerinde çalışan manuel tekerlekli arabalardır. Hafif ve küçük boyutta yaygındırlar ve bu nedenle rayları ve irinleri takmak veya çıkarmak kolay olabilir. Normatif demiryolu standartlarına uygun olarak, araba, Şekil 1.11'de gösterildiği gibi, rayın üzerine veya dışına kolayca yerleştirilebilen basit bir yapıdır [46]. Denetim arabası, insan gücüyle ray üzerinde hareket ettirilebilir. Hafif ve küçük boyutludurlar. Bu özelliğiyle ray üzerine yerleştirmek kolaydır. Denetim arabalarında ölçüm için genellikle çeşitli sensörler ve lazer ölçüm sistemleri bulunur. Bu

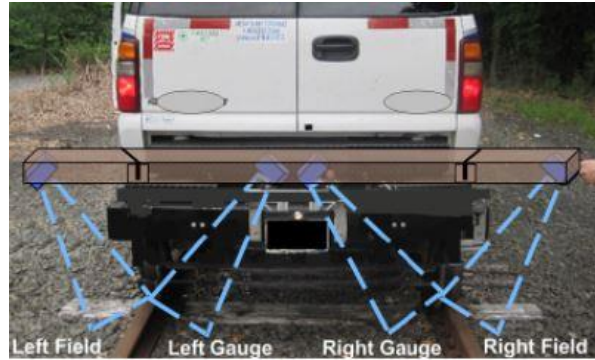
araçların dezavantajı, düşük ölçüm hızıdır. Özellikle, demiryolunu gezinme hızı İHA'lara göre çok düşüktür.



Şekil 1.11. Denetim aracı

### Karayolu Raylı Aracı

Karayolu raylı araçları, demiryollarında çalışabilecek şekilde dönüştürülmüş araçlardır. Bu araçlar bir dizi flanşlı çelik tekerlekle donatılmış normal bir motorlu araçtır. Karayolu ve demiryolunda kullanılabilirler. Araç çelik tekerlekleri indirerek, demiryolu raylarında kullanılabilir. Vizyon lazer sistemi, ultrasonik sensörler, girdap akımı sensörleri ve EMAT sensörleri gibi ray kusurlarını tespit etmek ve hat geometrisini ölçmek için yüksek raylı araçlarda birçok sensör donatılabilir. Karayolu raylı araçları, önceden ayarlanmış program tarafından otonom olarak veya bir operatör tarafından kontrol edilebilir. Denetim arabalarından ve İHA'lardan daha hızlıdır. Şekil 1.12, bir karayolu raylı aracını göstermektedir [47].



Şekil 1.12. Karayolu raylı aracı

### Hat Kayıt Aracı

Hat kayıt araçları, lazer görüş sistemleri, yüksek çözünürlüklü kameralar, ultrasonik ve girdap akımı sensörleri gibi gelişmiş sensörlerle donatılmış özel olarak yapılmış trenlerdir. Denetim sonucunda ray kusurlarının adları, rayların konumları, hat geometrisi izi elde edilebilir.

Demiryolu denetiminde yaygın olarak kullanılan bir hat kayıt aracı Şekil 1.13'te gösterilmektedir [48]. Hat kayıt araçları, çoğunlukla demiryolu endüstrisinde sabit bir sıklıkta

kullanılır. Denetim sonuçları, hat kalitesini deęerlendirmek ve hat bakımını planlamak için kullanılır. Muayene arabaları ve yüksek raylı araçlarla karşılaştırıldığında, hat kayıt araçları daha yüksek muayene hızında ve daha yüksek doğrulukta ölçüm yapabilir. Ancak, hat kayıt araçları yüksek maliyetlidir. Bu nedenle denetim uzun aralıklarla yapılır.

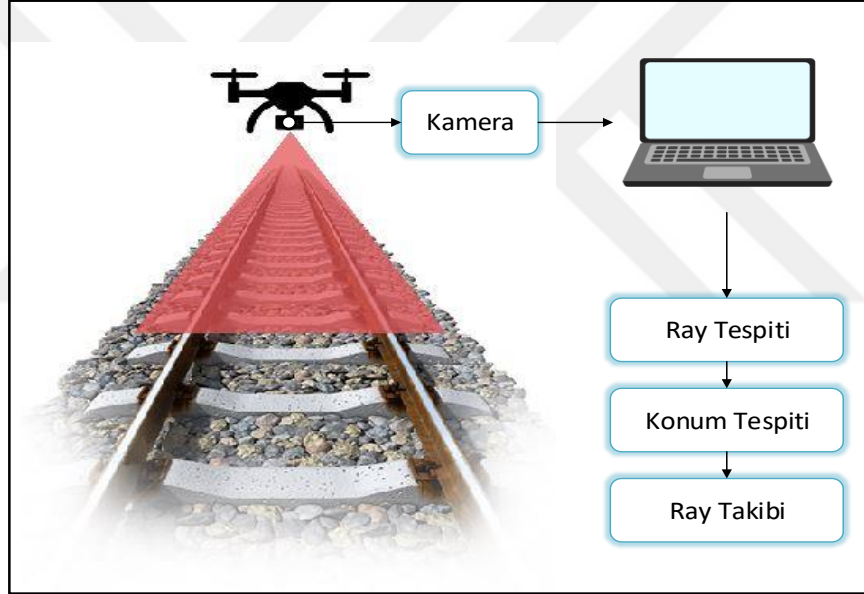


**Şekil 1.13.** Hat kayıt aracı

## 2. MATERYAL VE METOT

### 2.1. Otonom Ray Takibi

Otonom İHA'lar insan müdahalesi olmadan hareket edebilen araçlardır. Otonom İHA'lar çevreyi algılamak için çeşitli sensörlere sahiptir. Sensörlerden alınan verilere göre yol çıkarımı ve çevre algılaması yapabilirler. Bu sayede, herhangi bir insan müdahalesi olmadan çeşitli görevleri yerine getirebilirler. Bu çalışmada kullanılacak İHA, demiryolu boyunca otonom hareket edecektir. Bu şekilde insan yerine bir yazılım tarafından kontrol edilerek demiryolundan görüntü alınması sağlanacaktır. Önerilen algoritma üç bölümden oluşmaktadır. İlk aşamada alınan görüntülerden ray tespiti yapılır. İkinci aşamada ray hatları kullanılarak konum tespiti yapılır. Son aşamada ise, bulunan ufuk noktası kullanılarak demiryolunun otonom takibi sağlanır. Sistemimizin blok diyagramı Şekil 2.1'de verilmiştir.



Şekil 2.1. Otonom ray takibi için blok diyagramı

Otonom İHA'nın kontrolünde ray çizgilerini algılama önemli bir kısımdır. İHA demiryolunda ilerlerken ray çizgilerinin gerçek zamanlı olarak algılanması önemlidir. Ayrıca, ray çizgileri gerçek zamanlı tespit edilirken ray tespit algoritması hızlı ve güvenilir bir şekilde çalışmalıdır. Ray tespiti bilgisayarlı görmeye dayanan bir yöntemdir. Ray çizgileri, İHA'nın ön tarafında bulunan kamera sensörü ile tespit edilir. Kamera tarafından yakalanan karelerin genellikle yüksek bir kaliteye ihtiyacı vardır. Çünkü alınan görüntülerin canlı olması, yanlış bilgi alma olasılığını en aza indirecektir. Bu nedenle ray tespiti için en az 720p çözünürlüklü kamera içeren drone kullanılmalıdır.

### 2.1.1. HSV'ye Dayalı Ray Takip Algoritması

Yol çizgilerini algılama, yoldaki şeritleri tespit etmek ve her şeridin doğru konumunu ve şeklini bulma işlemidir. Yol çizgilerini algılama işlemi otonom sürüş sistemlerini mümkün kılan temel tekniklerden biri olarak kabul edilmektedir. Bu bölümde, demiryolu raylarının tespit edilebilmesi için bir görüntü işlemeye dayalı yöntem önerilmiştir. Önerilen yaklaşımda, raylar başarıyla tespit edilmiştir. Ray takibi iki ayrı drone ile yapılmıştır. Kullanılan dronlar Parro Anafi 4K ve DJI Tello'dur. Ray tespiti için dronların üzerinde bulunan kameradan alınan görüntüler kullanılmıştır. DJI Tello kullanarak ray takibi yapılırken alınan görüntüye, görüntü ön işleme adımları uygulandıktan sonra Hough dönüşümü ile yol çizgileri bulunmuştur. Tespit edilen ray çizgileri ileriki bölümde yol takibi için kullanılmıştır. Parrot Anafi'de ise kamera 90 derece aşağı bakacak şekilde ayarlanarak iki ray çizgisi belirlenip rayların orta noktası kullanılarak ray takibi yapılmıştır.

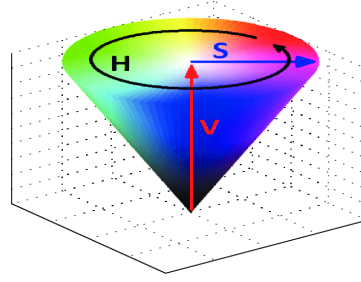
Demiryolları, traversler üzerine bağlanan rayların uç uca eklenmesiyle oluşur. Demiryollarında, rayları diğer elemanlardan ayıran bazı özellikleri vardır. Örneğin rayların rengi diğer elemanlardan farklıdır ve tekdüzedir. Görüntü işleme tabanlı bölütlemelerde renk özellikleri sıklıkla kullanılır. Ayırt edici özelliklerin olmaması, şerit algılama algoritmalarının benzer yerel görünüme sahip diğer nesnelerle karıştırılmasına neden olur. Ayrıca, bir raydaki tutarsız renk değerleri yol çizgilerinin bulunmasını zorlaştırır.

### Görüntü Bölütleme

Belirli bir bölgenin çıkarılması görüntü analizinde önemli bir işlemdir. Bu işlem genellikle görüntü bölütleme ile yapılır. Görüntü bölütleme, bir görüntüyü tutarlı bölgelere ayırma işlemidir. Tutarlılık genellikle piksel değerlerinin benzerliğine göre değerlendirilir. Piksel değerleri arasındaki benzerlik genellikle renk yoğunluğuna göre bulunur.

Önerilen bu yöntemde, görüntü işleme tabanlı yol çizgilerini bulma yöntemi geliştirilmiştir. Giriş verileri, hareket eden bir İHA'dan alınan renkli bir görüntü dizisidir. İHA üzerinde bulunan kamera; demiryolunu ve yol kenarında bulunan nesneleri görüntülemektedir. Alınan görüntü işlenmek üzere bilgisayara gönderilecektir. Orijinal görüntüler, çok fazla arka plan bilgisi ve gürültü içerdiğinden doğrudan ray tespiti için kullanılamaz. Bu nedenle, görüntüleri bazı işlemlerden geçirmek gerekmektedir. Hesaplama kolaylığı ve algoritma hızı açısından ilk olarak alınan görüntüye yeniden boyutlandırma uygulanmıştır. Görüntü bölütleme işlemi ile renge dayalı yol tespiti yapılmıştır. Görüntü piksellerinin renk özelliğine dayanan renkli görüntü bölütleme, görüntü analizinde sıklıkla kullanılan yöntemdir. Dronun alınan görüntü RGB formatındadır. RGB renk uzayı, kırmızı, yeşil ve maviden oluşan 3B Kartezyen koordinat sistemidir. Bu modelde diğer tüm renkler ana renkler kırmızı, yeşil ve mavi eklenerek üretilebilir. Renk bilgisi ve yoğunluk bilgisi RGB renk uzayında karıştırılır. Bu nedenle, renk bazlı görüntü analizi ve segmentasyon için

RGB kullanılmaz. HSV’de renkler, renk özü, doygunluk ve parlaklık olmak üzere üç boyutlu koordinat sisteminde tanımlanır (Şekil 2.2).



Şekil 2.2. HSV renk uzayı

HSV renk uzayı, görüntü segmentasyonu, nesne tanıma, özellik algılama ve görüntü analizinde yaygın bir biçimde kullanılmaktadır [49] [50]. RGB formatındaki görüntüyü HSV formatına dönüştürmek için görüntünün R, G ve B bileşenlerinin çıkarılması gerekir. RGB’den HSV’ye dönüşüm için gerekli denklemler aşağıda verilmiştir.

$$H = \arccos \frac{\frac{1}{2}(2R - G - B)}{\sqrt{(R - G)^2 - (R - B)(G - B)}} \quad (2.1)$$

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \quad (2.2)$$

$$V = \max(R, G, B) \quad (2.3)$$

### DJI Tello ile Otonom Ray Takibi

Çalışmanın bu bölümünde ilk olarak, RGB formatlı görüntü HSV formatına dönüştürülmüştür. Eşik seçimi, renge göre bölütleme işleminde önemli bir faktördür. Görüntü HSV renk uzayına dönüştürüldükten sonra, yalnızca ilgilenilen renge göre algılama yapılması gerekmektedir. Bu nedenle yol çizgilerinin, üst ve alt renk sınırları belirlenip renge göre maskeleme işlemi yapılmıştır.

Renge göre maskeleme işleminin ardından ikili görüntü elde edilmiştir. İkili görüntünün kullanım açısından birçok avantajı vardır. Renkli bir görüntü, hedef nesne dışında, arka planda bulunan nesneleri de içerir. Ancak, renge göre maskeleme işleminden sonra elde edilen ikili görüntüde raylar 1’e (beyaz) rayların çevresi ise 0’a (siyah) dönüştürülmüştür. Böylece bu

aşamadan sonra yalnızca ikili görüntüler üzerinde işlem yapılacak ve bu da hesaplama hızını artırarak ray çizgilerini belirlemenin olası bir yolu olacaktır.

Elde edilen görüntüde eşikleme işleminin ardından yol çizgilerinin daha belirgin olabilmesi için bazı morfolojik işlemler uygulanmıştır. Morfolojik işlemler, genellikle ikili görüntülere uygulanır. İkili görüntüler çok sayıda kusur içerebilir. Morfolojik işlemler, görüntünün biçimini ve yapısını hesaba katarak bu kusurları ortadan kaldırmak için kullanılır. Bu çalışmada, genişletme (dilation) ve açma (opening) işlemleri uygulanmıştır. Morfolojik işlemlerin ardından elde edilen görüntüden kenar çıkarımı yapılmıştır.

### Canny Kenar Çıkarım Algoritması

Kenarlar, bir görüntünün önemli bilgilerini gösterir. Kenar algılama sonuçları, görüntü analizini doğrudan etkileyecektir. Bu nedenle görüntü işlemede çok önemli bir konudur. Yapılan çalışmalarda çok sayıda kenar algılama tekniği önerilmiştir [51], [52], [53]. Roberts operatörü, Prewitt operatörü, Sobel operatörü gibi birçok kenar algılama operatörü vardır. Bu operatörler; basit hesaplama, hız ve uygulamada kolaylık gibi birçok avantaja sahip olsalar da, gürültüye karşı daha hassastırlar ve mühendislik uygulamalarında mükemmel değildir.

Canny algoritması, John F. Canny tarafından 1986'da önerilen, aşamaları olan popüler bir kenar algılama algoritmasıdır. Genel kenar algılama algoritmasıyla karşılaştırıldığında, çoğu durumda Canny algoritması en iyi performansa sahiptir [54].

Algoritmada ilk aşama görüntü üzerindeki gürültünün azaltılması işlemidir. Bu işlem görüntüdeki gürültüyü 5x5 Gauss filtresi ile gidermektedir. Daha fazla Gauss filtresi ile daha az kenar algılanır. Algoritmanın ikinci aşaması ise görüntünün yoğunluk gradyanını bulma işlemidir. Gauss filtresi uygulanan görüntü daha sonra, birinci türevi bulmak için hem yatayda ( $G_x$ ) hem de dikeyde ( $G_y$ ) bir Sobel çekirdeği ile filtrelenir. Bu iki görüntüden, her piksel için kenar gradyanı ve yönü Denklem 2.4 ve 2.5'teki gibi bulunur.

$$\text{Edge Gradient}(G) = \sqrt{G_x^2 + G_y^2} \quad (2.4)$$

$$\text{Angle}(\theta) = \tan^{-1} \left( \frac{G_x}{G_y} \right) \quad (2.5)$$

Gradyan büyüklüğü ve yönü bulunduğundan sonra, hesaplanan açı değeri bilgisi kullanılarak kenarı oluşturamayabilecek istenmeyen pikselleri kaldırmak için tam bir görüntü taraması yapılır. Ardından ikili eşikleme ile kenarlar daha belirgin hale getirilir.

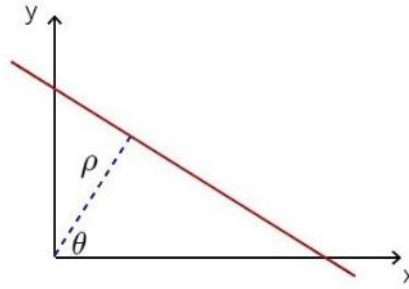
Bu çalışma için uygulanan ve değerlendirilen tüm kenar dedektörlerinden en iyi kenar görüntülerini üreten kenar bulma algoritması, Canny kenar bulma algoritmasıdır. Canny operatörü,

daha düşük hata oranlarına, daha yüksek hassasiyete ve tek yanıtı sahiptir. Bu nedenle, kenar çıkarım işlemi için "Canny" kenar çıkarma algoritması kullanılmıştır.

### Hough Dönüşümü

Hough dönüşümü, görüntü analizinde kullanılan bir özellik çıkarma yöntemidir. Hough dönüşümü, çizgiler, daireler, elipsler gibi herhangi bir düzenli eğrinin özelliklerini belirlemek için kullanılabilir.

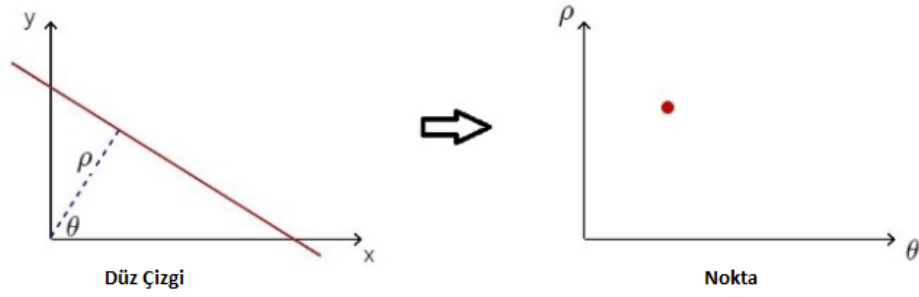
Görüntü uzayındaki çizginin denklemi  $y = mx + c$  biçimindedir, burada  $m$  eğimdir ve  $c$  doğrunun  $y$  kesişimidir. Ancak, bu denklem dikey çizgileri bulamaz. Bu nedenle kutupsal sistemde  $(\rho, \theta)$  çifti kullanılarak doğru açık bir şekilde tanımlanmalıdır. Burada  $\rho$ , başlangıçtan çizgiye en kısa mesafedir.  $\theta$  ise,  $x$  eksenine ile mesafe çizgisi arasındaki açıdır (Şekil 2.3). Böyle temsil etmenin faydalarından biri, dikey çizgileri  $\rho$  ve  $\theta$  ile tanımlayabilmemizdir. Kartezyen sistemde sadece  $(a, b)$  parametrelerini kullanarak böyle bir tanımlama mümkün değildir. Belirli bir doğru için, belirli  $\rho$  ve  $\theta$  değerlerini belirlenir. Ardından, bu çizgiye ait her bir  $x, y$  noktası için Denklem 2.6'daki eşitlik sağlanır.



Şekil 2.3. Çizgiye en kısa mesafe ve oluşan açı

$$\rho = x \cos(\theta) + y \sin(\theta) \quad (2.6)$$

Çizgi, belirlenen  $\rho, \theta$  koordinatlarıyla Hough uzayında tek bir nokta olarak ifade edilir (Şekil 2.4). Bu şekilde görüntü uzayında ortak bir noktada kesişen başka çizgiler de varsa bu noktalar sinüs yapısı oluşturur. Hough uzayında bir sürü sinüs yapısı elde edilir. Bu yapıların kesişimi ile düz çizgiler tespit edilir.



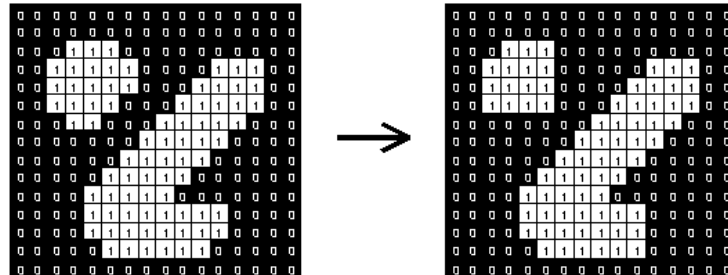
Şekil 2.4. Tek nokta ile ifade

### Morfolojik İşlemler

İkili görüntüler çok sayıda kusur içerebilir. Bu kusurları azaltmak için morfolojik operatörler sıklıkla kullanılır. Morfolojik işlemler, görüntünün biçimini ve yapısını hesaba katarak bu kusurları ortadan kaldırma işlemini yapar. Morfolojik görüntü işlemede iki temel özellik vardır. Bunlar; açma ve kapama işlemleridir. Bu operatörler genellikle ikili görüntülere uygulanırlar.

### Morfolojik Açma

Morfolojik açma, ikili görüntü üzerinde hem bir erozyon hem de bir genişleme işlemi gerçekleştirir. Genellikle gürültü kaldırma işlemleri için kullanılır. Açma operatörü, açma işlemi uygulanacak ikili görüntü ve çekirdek görüntüyü kullanarak işlem yapar. Açma işleminin etkisini çekirdek görüntü belirler. Şekil 2.5'te  $3 \times 3$  kare çekirdek görüntü kullanarak açmanın etkisi gösterilmiştir.

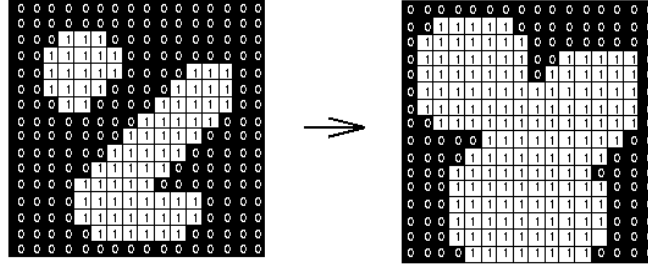


Şekil 2.5.  $3 \times 3$  kare çekirdek görüntü kullanarak açmanın etkisi

### Morfolojik Genişletme

Morfolojik genişletme, görüntü işleme alanında sıklıkla kullanılan operatördür. Genellikle ikili görüntülere uygulanır. Ancak gri tonlamalı görüntülerde çalışan sürümleri de vardır. Operatörün ikili görüntü üzerindeki temel etkisi, beyaz piksel bölgelerinin sınırlarını kademeli olarak genişletmektir. Genişletme operatörü, giriş olarak 2 veri alır. Bu verilerden birincisi genişletme işlemi uygulanacak ikili görüntüdür. Diğeri ise genişletme operatörünün etkisini

belirleyen çekirdek görüntüdür. Şekil 2.6'da  $3 \times 3$  kare çekirdek görüntü kullanarak genişletmenin etkisi gösterilmiştir.

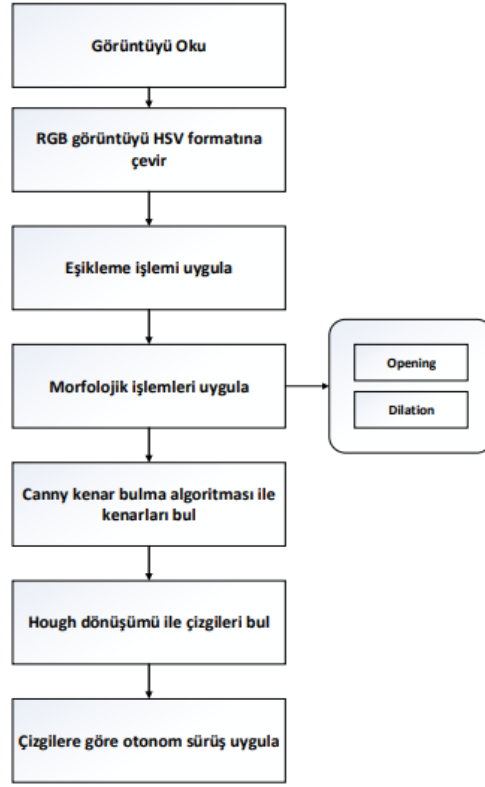


Şekil 2.6.  $3 \times 3$  kare çekirdek görüntü kullanarak genişletmenin etkisi

### DJI Tello ile Ray Takibi İçin Önerilen Yöntem

Bu çalışmada, ray tespiti için histogram eşiğine dayalı yöntem kullanılmıştır. Bu renk tabanlı yöntem, HSV renk uzayına ve histogram eşiğine dayanmaktadır. İlk olarak İHA üzerinde bulunan kameradan yol görüntüsü alınmıştır. Alınan renk tabanlı görüntü bölütleme işlemi için HSV renk formatına dönüştürülmüştür.

Demiryolundan alınan görüntüler RGB renk uzayındadır. Önerilen yaklaşımda, RGB renk uzayındaki ray görüntüsü HSV renk uzayına dönüştürülür ve ardından dönüştürülen görüntü yoğunluk ve renge bağlı olarak üç farklı bileşene bölünür. Her üç bileşen için (ton, doygunluk ve değer) histogram hesaplanır. Ardından, en düşük ve en yüksek eşik değeri seçilir ve eşik değerine göre ray görüntüsü çıkarılır. Ancak eşikleme işlemi tek başına yeterli olmayabilir. Elde edilen görüntü gürültülü bir görüntü olabilir. Böyle durumlar için morfolojik işlemler uygulanması gerekir. Morfolojik işlemlerden sonra maske elde edilir. Önerilen yaklaşımın akış diyagramı Şekil 2.7'de verilmiştir. Bu yöntemin ray görüntüsünün çıkarılmasında etkili olduğu gösterilmiştir.

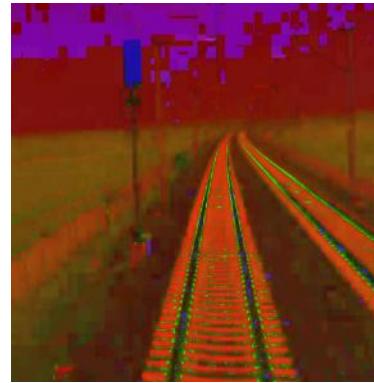


**Şekil 2.7.** Önerilen yöntemin akış diyagramı

Yapılan çalışmada İHA üzerine bağlı kameradan alınan görüntüler üzerinde işlemler yapılmıştır. Şekil 2.7’de verilen akış diyagramına göre raylar tespit edilmiştir. Alınan görüntü örneği Şekil 2.8.a’ verilmiştir. Alınan görüntü HSV formatına çevrilmiştir (Şekil 2.8.b). HSV formatlı görüntüye yapılan eşikleme işlemi sonuçları Şekil 2.9.a’da verilmiştir. Eşikleme sonucu elde edilen görüntüye morfolojik işlemler uygulanmıştır (Şekil 2.9.b).



a)



b)

**Şekil 2.8.** a) Alınan görüntü b) HSV formatlı görüntü



Şekil 2.9. a) Eşikleme işlemi sonucu b) Morfolojik işlemler sonucu

Morfolojik işlemlerden sonra kenar çıkarma işlemi uygulanmıştır. Sonuçları Şekil 2.10.a'da gösterilmiştir. Kenar çıkarma Hough dönüşümü için yapılmaktadır. Hough dönüşümü, bir görüntüdeki belirli bir şeklin özelliklerini göstermek için kullanılan bir tekniktir. Hough dönüşümü, yaygın olarak çizgiler, daireler, elipsler gibi düzenli eğrilerin saptanması için kullanılır. Hough dönüşümü, bu çalışmada ray çizgilerini belirlemek için kullanılmıştır. İşlem sonrası Şekil 2.10.b'de verilen sonuç alınmıştır.



Şekil 2.10. a) Görüntüye uygulanan kenar çıkarma sonucu b) Hough Dönüşümü

### Ufuk Noktası Tespiti

Ufuk noktası tespiti, bilgisayarlı görme alanında önemli bir araştırma konusudur ve otonom bir İHA'nın görsel navigasyon sistemi için önemli bir bileşendir. Ufuk noktası tespiti, robot navigasyonu, hedef yeniden yapılandırma, kamera kalibrasyonu gibi alanlarda kullanılabilir. Ufuk noktası tespit edilerek, İHA'ların otonom navigasyon sistemlerinin en önemli parçası tamamlanabilir. Bu işlem için İHA'dan alınan görüntü kullanılarak ufuk noktası tahmin edilmelidir. Ufuk noktası tahmini sonrasında İHA'nın yönlendirilmesi ufuk noktasına göre yapılmalıdır.

Literatür incelendiğinde, ufuk noktası tahmini için kullanılan çeşitli teknikler vardır. Ancak kullanılan tekniklerin birçoğu yüksek hesaplama maliyetleri nedeniyle gerçek zamanlı kullanıma

uygun değildir. Bu nedenle, değer çalışmalarda kıyaslandığında hızlı çalışan çizgilerin kesişimine dayalı yöntem kullanılmıştır. Rayların kesintisiz çizgi oluşturması kenar tabanlı ufuk noktası algılama yönteminin uygulanmasını kolaylaştırmaktadır. Önerilen yöntemde amaç, ray çizgilerine ait özellikleri bulmaktır.

## Parrot Anafi ile Otonom Ray Takibi

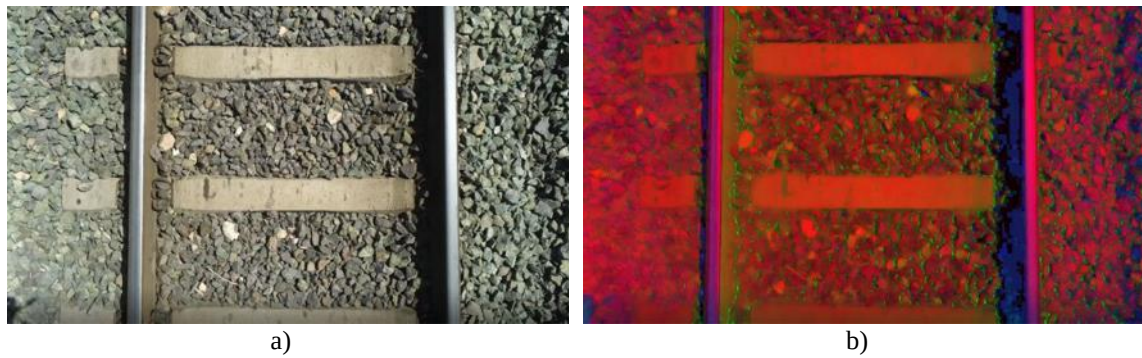
### Ray çizgilerinin tespiti için önerilen yaklaşım

Rayların tespiti için HSV'ye dayalı renk bölütleme kullanılmıştır. İHA tarafından alınan RGB görüntü 90 derece açıdan alındığı için sadece demiryolunu içermektedir. Böyle bir görüntüde, HSV tabanlı ray tespiti iyi sonuçlar vermektedir ve hız açısından avantajlıdır. Bu nedenle alınan görüntü ilk olarak HSV renk uzayına dönüştürülmüştür. Elde edilen sonuçlar Şekil 2.11'de gösterilmiştir. Ardından, raylara ait ikili maske elde edilmiştir.

Renk özü, doygunluk ve parlaklık değerlerine ait düşük eşik ve yüksek eşik değerleri uygulanmıştır. Uygulanan eşik değerleri, Tablo 2.1'de verilmiştir. Eşikleme işlemi uygulandıktan sonra rayın renk değerini içeren pikseller 0 olarak belirlenecek, kalan pikseller ise 1 olacaktır.

**Tablo 2.1.** Ray tespiti için eşik değerleri

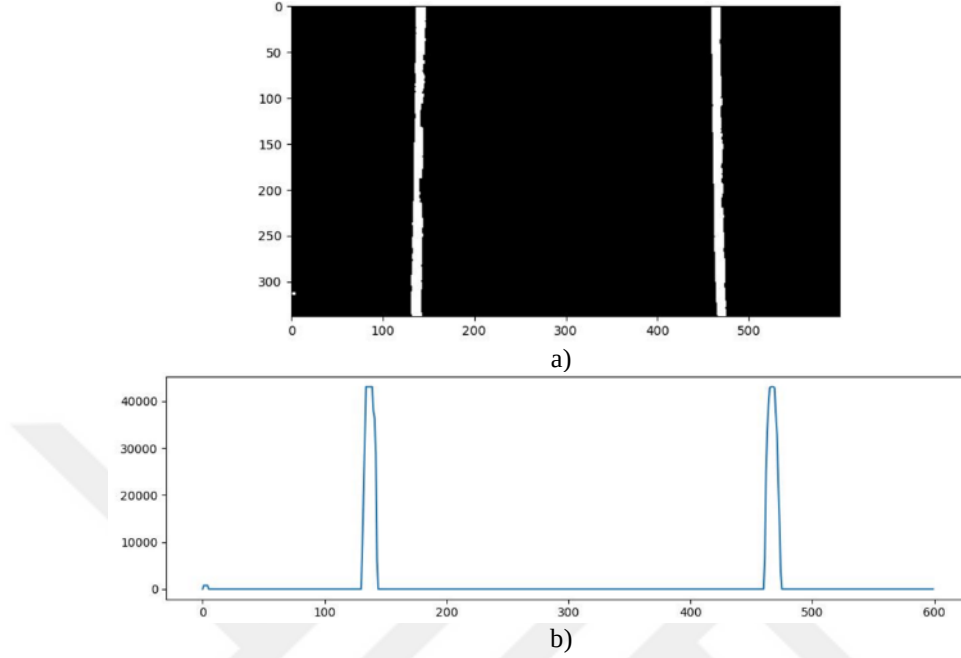
| Kanal    | H   | S   | V   |
|----------|-----|-----|-----|
| En Küçük | 85  | 0   | 61  |
| En Büyük | 121 | 255 | 255 |



**Şekil 2.11.** a) RGB renk uzayı b) HSV renk uzayı

Alınan görüntüye eşikleme yapıldıktan sonra, Şekil 2.12.a'da görüldüğü gibi ray çizgilerini içeren net bir şekilde öne çıktığı ikili bir görüntü elde edilmiştir. Bu aşamadan sonra, ikili maskede bulunan beyaz piksellerin hangisinin sağ raya hangisinin sol raya ait olduğunun belirlenmesi gerekmektedir. Bu nedenle, beyaz pikselleri algılamak için bir histogram oluşturulmuştur. Şekil

2.12.b’de görüldüğü gibi histogram çizildiğinde iki tepe noktası elde edilecektir. Bu tepe noktaları, sol ray çizgisine karşılık gelen bir sol zirve ve sağ şerit çizgisine karşılık gelen bir sağ zirveyi belirtmektedir.



Şekil 2.12. Ray konumlarının histogram tabanlı tespiti a) İkili görüntü b) Histogram pik noktaları

### İHA konumunun belirlenmesi

Parrot Anafi 4K’nın kamerası raya 90 derece ayarlandığı için alınan görüntüde raylar dikey bir biçimde görünmektedir. Bu durumda İHA’nın bulunması gereken konum iki rayın orta noktası olarak hesaplanabilir. Şekil 2.12.b’teki histogram grafiğinde görüldüğü gibi raylar iki adet pik noktası oluşturmaktadır. Elde edilen pik noktalarından sağ ve sol rayı belirlemek için ekranın yatay uzunluğu ikiye bölünmüştür. Yatay uzunluğun yarısı referans nokta olarak alınmıştır. Referans noktasının solunda kalan pik noktası sol rayı, sağında kalan pik noktası ise sağ rayı ifade etmektedir. İHA’nın bulunması gereken noktanın hesaplanması için aşağıdaki denklemler kullanılarak hesaplanmaktadır. İlk olarak sağ pik noktasının x koordinatı ile sol pik noktasının x koordinatı arasındaki fark Denklem 2.7’deki gibi hesaplanmıştır.

$$\text{fark} = \text{sagx} - \text{solx} \quad (2.7)$$

Ardından, aradaki farkın yarısı bulunarak sol pikselin x koordinatına eklenmesi gerekmektedir. Bunlar için denklemler aşağıda verilmiştir.

$$\text{orta} = \text{fark}/2 \quad (2.8)$$

(2.9)

$\text{konum} = \text{sol} + \text{orta}$

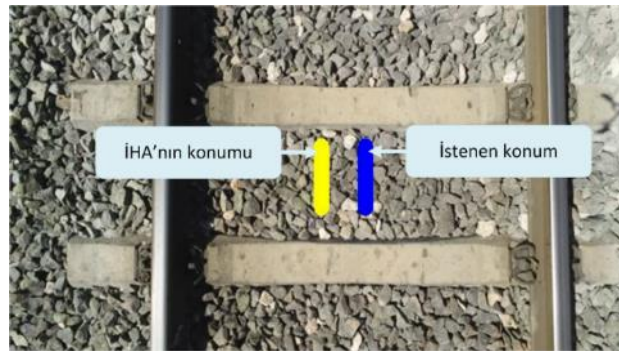
Şekil 2.13'te görüldüğü gibi İHA'nın o anda bulunması gereken konum işaretlenmiştir. Rayların orta noktası ve İHA'nın anlık konum bilgisi arasındaki fark PID denetleyiciye aktarılacaktır. PID denetleyici ile her görüntü çerçevesi için oluşan hata değerine göre İHA kontrolü sağlanacaktır.



Şekil 2.13. İHA'nın bulunması gereken konum

### Ray takibi

İHA'nın rayı takip etmesi, İHA'nın bulunduğu konumun x koordinatı ile istenen konum arasındaki mesafenin sıfıra indirilmesi ile sağlanacaktır. İHA'nın bulunduğu konum, kameradan alınan görüntünün x koordinatının orta noktasıdır. İstenen konum denklem 2.9'da hesaplanan konumdur. Hesaplanan konumlar, her görüntü çerçevesi üzerinde Şekil 2.14'te gösterildiği gibi işaretlenmektedir. İstenen konum ile İHA'nın konumunun farkı sıfırda tutulurken İHA, ileri yönlü sabit hızla ilerleyecektir. Demiryolu takibi bu şekilde sağlanacaktır.



Şekil 2.14. İHA'nın bulunduğu konum ile istenen konumun belirlenmesi

### 2.1.2. Gabor Filtresi ve Bulanık PID Tabanlı Ray Takibi

Ray görüntüsüne uygulanan Hough dönüşümü sonucunda elde edilen çizgiler ufuk noktasında kesişecektir. Gürültü gibi etkenler nedeniyle bu kavşakların tamamı aynı noktada

olmayacaktır. Bu nedenle, bu kesişim noktalarından hangisinin gerçek ufuk noktası olduğunu belirlemek zordur. Ufuk noktası tahmininin amacı bu kesişimlerden tek bir nokta elde etmektir. Bu nedenle, çizgilerin kesişiminde en küçük kareler çözümü kullanılmalıdır [55].

2 boyutlu bir çizgi üzerindeki nokta  $a = (a_0, a_1)^T$  olarak ifade edilir. Yön vektörü  $n = (n_0, n_1)^T$  olarak ifade edilir. Böylece çizgi  $p = a + tn$  olarak gösterilir. Diyelim ki K-hattımız var. Tüm bu çizgiler ufuk noktasında kesişecek. Dolayısıyla birden fazla kesişme noktası. Tek kesişme noktası, en küçük kareler yöntemi kullanılarak bulunacaktır. Kare mesafelerin toplamı en aza indirilecektir. K çizgilerinin mesafesi şu şekilde hesaplanır:

$$D(p; A, N) = \sum_{i=1}^K (p; a_i, n_i) = \sum_{i=1}^K (a_i - p)^T (I - n_i n_i^T) (a_i - p) \quad (2.10)$$

$$p' = \text{argmin}_p D(p; A, N) \quad (2.11)$$

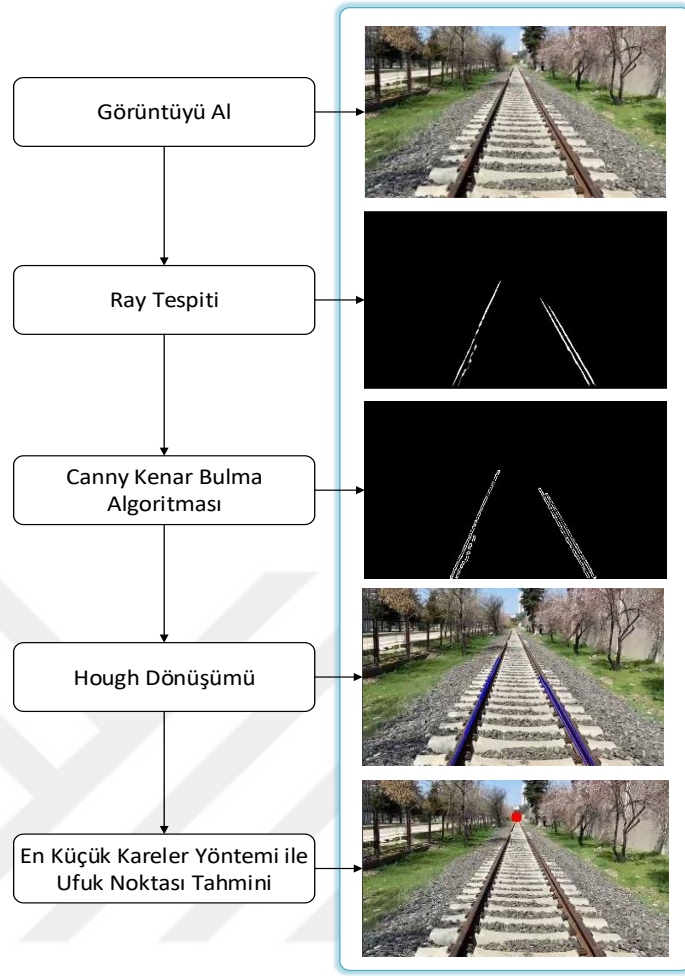
$p'$ 'ye göre türev alınırsa aşağıdaki ifade elde edilir:

$$\frac{\partial D}{\partial p} = \sum_{i=1}^K -2(I - n_i n_i^T) (a_i - p) = 0 \quad (2.12)$$

Aşağıdaki doğrusal denklem yeniden düzenlenerek elde edilir.

$$R = \sum_{i=1}^K (I - n_i n_i^T), q = \sum_{i=1}^K (I - n_i n_i^T) a_i \quad (2.13)$$

Ufuk noktası tahmini için uygulanan algoritma Şekil 2.15'te verilmiştir.



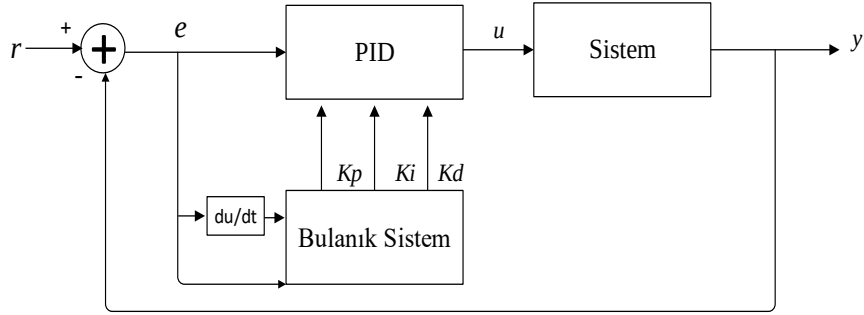
Şekil 2.15. Ufuk noktası tahmini için uygulanan algoritma

### Bulanık PID Tabanlı Ray Takibi

Uygun PID parametrelerinin seçimi ile PID kontrolörün başarısı artırılabilir. Doğru PID kontrol parametrelerinin seçilmemesi performansı büyük ölçüde etkileyecektir. Bu nedenle, PID kontrol parametrelerinin ayarlanırken dikkat edilmelidir. PID kontrol parametreleri genellikle deneyimli uzmanlar tarafından belirlenir. Bu çalışmada, PID kontrol parametrelerini ayarlamak için bulanık mantık kullanılmıştır. Bulanık sistem için ilk olarak kurallar belirlenmiştir. Daha sonra, bu kurallar, PID kontrol parametrelerini çevrimiçi olarak ayarlamak için kullanılmıştır. Bulanık-PID denetleyicisi, PID denetim parametrelerini düzenlemek için bulanık sistem çıktısını kullanmaya dayanır. Bu yönüyle, geleneksel bir PID denetleyiciden farklıdır. Bu teknikle optimize edilmiş bulanık PID denetleyicisi, sistemdeki ani değişikliklere gerçek zamanlı olarak hızlı dinamik yanıt almayı sağlar.

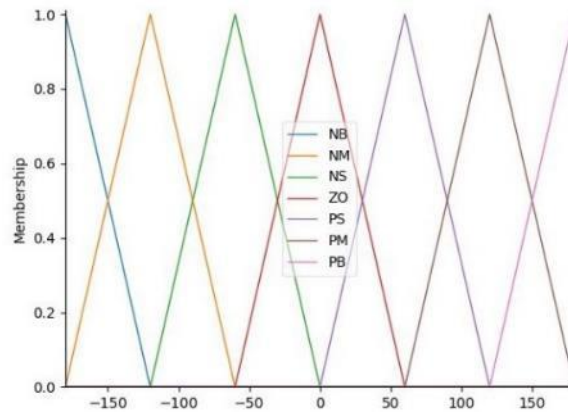
Kontrol parametrelerinin kendi kendine ayarlanmasıyla, sistemin çıktısına göre optimum değerler belirlenir. Bulanık PID denetleyici, sistemin mevcut hata değerine göre kontrol parametrelerini çevrimiçi olarak ayarlayabilir. Bu işlem, kontrolün hassasiyetini artırır. Bu

nedenle, geleneksel PID denetleyiciden daha iyi bir performans sağlanır. Sistemin blok diyagramı Şekil 2.16'daki gibidir.

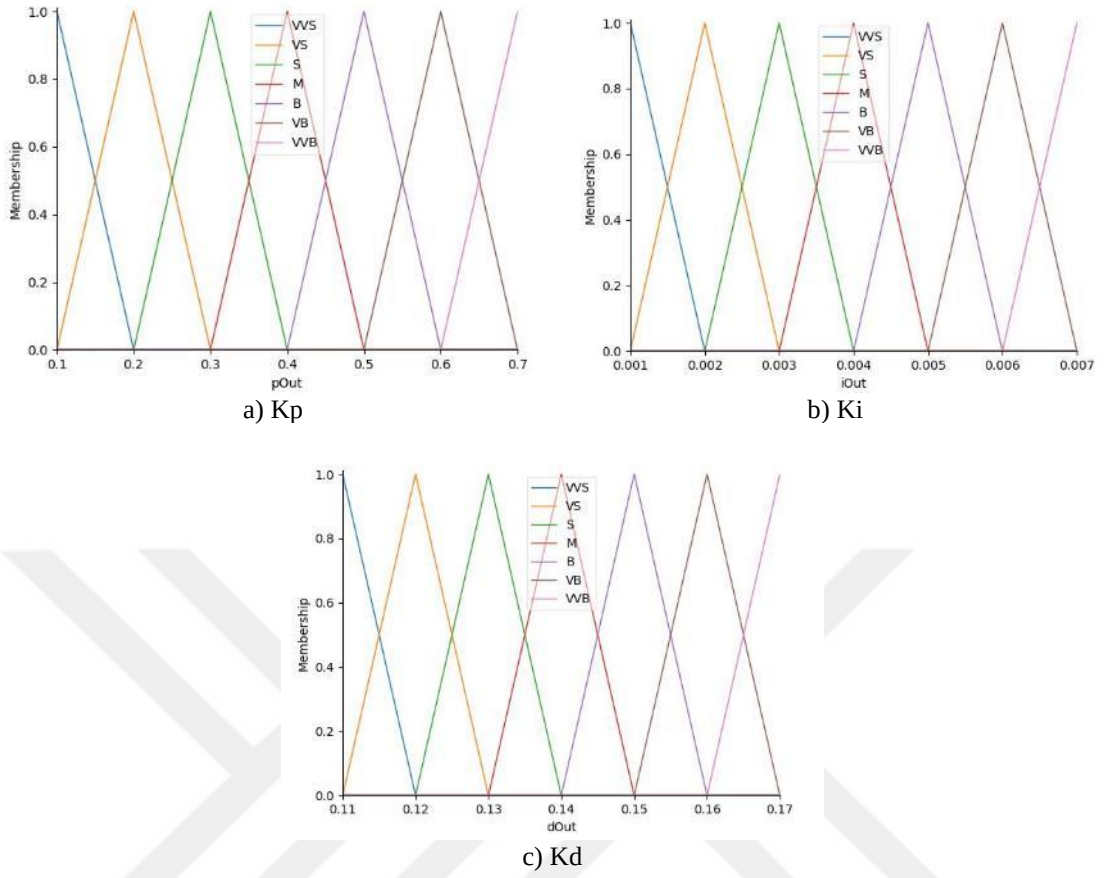


**Şekil 2.16.** PID kontrol parametrelerini ayarlamak için kullanılan bulanık sistemin blok şeması

Bulanık sistem, hata (e) 'yi ve hatanın (ed) türevini girdi olarak alır. Hatanın türevi ile hata arasındaki ilişkiye göre  $K_p$ ,  $K_i$  ve  $K_d$  değerleri için çıktı verir. Sistemde hata (e) ve hatanın türevi (ed) [-180,180] aralığında seçilmiştir. Bu aralığın seçilmesinin nedeni, İHA'dan alınan görüntünün yeniden boyutlandırılmasıyla ilgilidir. Çıktı olarak  $K_p$  [0.1, 0.8],  $K_i$  [0.001, 0.008] ve  $K_d$  [0.11, 0.18] seçilmiştir. Şekil 2.17, hata (e) ve hatanın türevi (ed) için üyelik fonksiyonlarını göstermektedir. Şekil 2.18,  $K_p$ ,  $K_i$  ve  $K_d$  değerlerinin çıktısı için üyelik fonksiyonlarını gösterir. Ek olarak, kurallar Tablo 2.2 ve Tablo 2.3'te gösterilmektedir.



**Şekil 2.17.** Hata (e) ve hatanın türevi (ed) için üyelik fonksiyonları



Şekil 2.18. a) Kp için üyelik işlevleri b) Ki için üyelik işlevleri c) Kd için üyelik işlevleri

Tablo 2.2. Kp ve Ki için kurallar

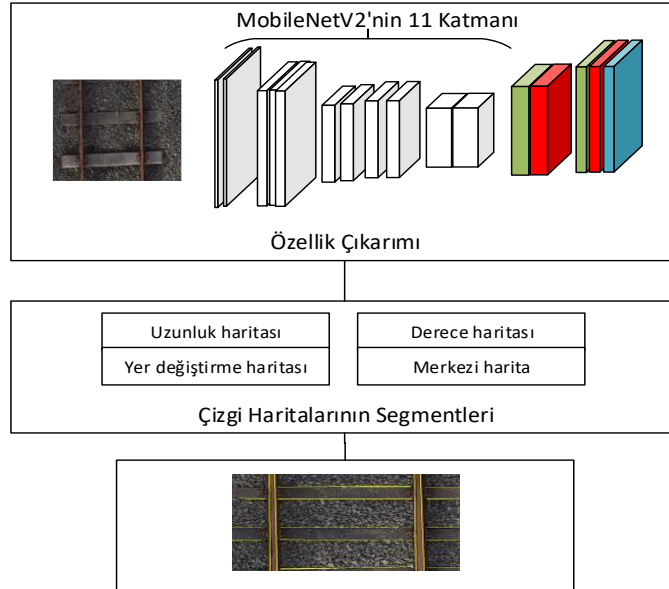
| Kp,Ki             |      |    |    |     |    |    |     |
|-------------------|------|----|----|-----|----|----|-----|
| Hatanın<br>Türevi | Hata |    |    |     |    |    |     |
|                   | NB   | NO | NK | N   | PK | PM | PB  |
| NB                | O    | K  | ÇK | ÇÇK | ÇK | K  | O   |
| NO                | B    | O  | K  | ÇK  | K  | O  | B   |
| NK                | ÇB   | B  | O  | K   | O  | B  | ÇB  |
| N                 | ÇÇB  | ÇB | B  | O   | B  | ÇB | ÇÇB |
| PK                | ÇB   | B  | O  | K   | O  | B  | ÇB  |
| PO                | B    | O  | K  | ÇK  | K  | O  | B   |
| PB                | O    | K  | ÇK | ÇÇK | ÇK | K  | O   |

**Tablo 2.3.** Kd için kurallar

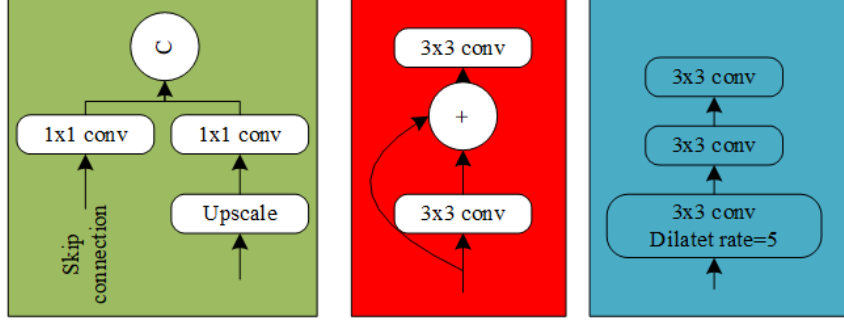
| Kd                |      |    |    |     |    |    |     |
|-------------------|------|----|----|-----|----|----|-----|
| Hatanın<br>Türevi | Hata |    |    |     |    |    |     |
|                   | NB   | NO | NK | N   | PK | PO | PB  |
| NB                | O    | B  | ÇB | ÇÇB | ÇB | B  | O   |
| NO                | K    | O  | B  | ÇB  | B  | O  | K   |
| NK                | ÇK   | K  | O  | B   | O  | K  | ÇK  |
| N                 | ÇÇK  | ÇK | K  | O   | K  | ÇK | ÇÇK |
| PK                | ÇK   | K  | O  | B   | O  | K  | ÇK  |
| PO                | K    | O  | B  | ÇB  | B  | O  | K   |
| PB                | O    | B  | ÇB | ÇÇB | ÇB | B  | O   |

### 2.1.3. Hafif Ağırlıklı Çizgi Tespiti Yaklaşımına Dayalı Ray Takibi

Ray çizgilerinin tespiti için hafif bir çizgi segmentasyon yöntemi kullanıldı. Bu yöntemin omurgası, hafif bir MobileNetV2 ve azaltılmış bir hat tahmin sisteminden oluşur [56]. Önerilen mobileNetv2 sistemi, kodlayıcı-kod çözücü mimarisine dayanmaktadır. Kodlayıcı ağı, onu daha da hafif hale getirmek için MobileNetV2'nin parçalarını kullanır. Kodlayıcı kısım, özellik çıkarımı için MobileNetV2 mimarisinin ilk 11 katmanını kullanır. Bu katmanlarda darboğaz blokları kullanıldığından, parametre sayısı orijinal MobileNetV2'deki parametrelerin yaklaşık %7'si (0.25MB) kadardır. Şekil 2.19, önerilen hat segmentasyon ağının mimarisini göstermektedir.

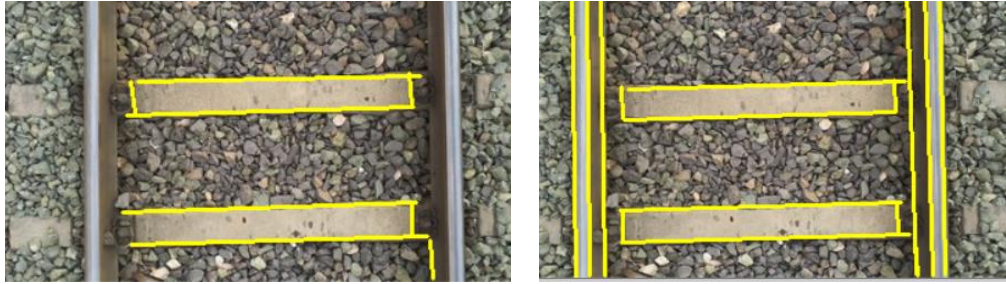
**Şekil 2.19.** Hat segmentasyon ağının mimarisi

Dekoder ağı, A, B ve C blok tiplerinin bir kombinasyonudur. Kod çözücü bloğu 3 tip katmandan oluşmaktadır. Bunlardan ilki, atlama ve lüks bağlantıları birleştirir. İkinci blok artık bir bağlantıdan oluşur. Son katman, iki 3x3 evrişim katmanından ve ardından 1x1 evrişim katmanından oluşur. Kullanılan katman yapıları Şekil 2.20'de verilmiştir.



Şekil 2.20. Katman yapıları

Kod çözücü blokları ayrıca MobileNetV2'nin darboğaz katmanlarını kullanır. Bu katmanların eklenmesiyle ağıdaki parametre sayısı 0,56 milyon olur. Daha sonra çizgi segment haritaları oluşturulur. Bu çalışmada hem travers hem de ray için elde edilen hatlar iki farklı amaç için kullanılmıştır. Ray için elde edilen hatlar ray takibi için kullanılmaktadır. Her iki ray için de birçok hat tespit edilebilir. Şekil 2.21, küçük, hafif çizgi segmentasyon yöntemini temsil etmektedir.



a) Tiny model için çizgi sonuçları

b) Large model için çizgi sonuçları

Şekil 2.21. Farklı modeller için çizgi sonuçları

Şekil 2.21.a'da küçük hat bölütleme modeli kullanıldığında rayların algılanmadığı görülmektedir. Şekil 2.21.b'deki büyük model kullanıldığında ray konumlarının birden fazla çizgi ile gösterildiği görülmektedir. Takip için sağ ve sol raylar tek çizgi ile gösterilmelidir. Bu nedenle, görüntünün merkezinin sağındaki çizgilerin x eksenine göre ortalaması, sağ rayı göstermek için ve soldakilerin sol rayı göstermek için ortalaması alınır. Aynı durum birbirine yakın traversler için de elde edilmiş ve traversin üst ve alt kenarları gösterilmiştir. Ray çizgilerinin ortalaması alınarak sağ

ve sol ray için tek çizgi oluşturulmuştur. Bu çizgiler kullanılarak rayların orta noktası bulunmuştur. Bu nokta İHA'nın yolu ortalaması için kullanılmıştır.

#### 2.1.4. Yolact Tabanlı Ray Takibi

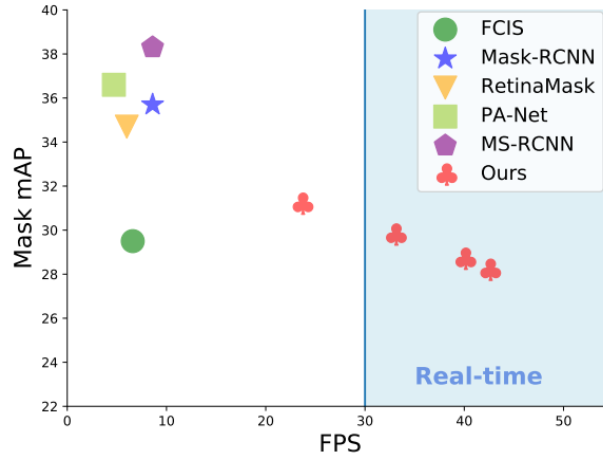
Otonom ray takibi algoritmasında rayların tespiti için derin öğrenme kullanılma amacı, İHA'nın ray çizgilerini algılamasını iyileştirmektir. Önerilen sistem, gerçek zamanlı ray çizgilerini algılar ve raylara ait maskeleri elde eder. Elde edilen maske bilgisi otonom ray takibi algoritmasında kullanılacaktır.

Son yıllarda, bilgisayar görmesi alanında örnek segmentasyonunda büyük ilerlemeler kaydedilmiştir. Mask RCNN [57] ve Unet [58] gibi örnek segmentasyonuna yönelik son teknoloji yaklaşımlar, bu gelişmelerden bazılarıdır. İki aşamalı nesne tahmin algoritmalarını genişleterek ortaya çıkmıştır. Ancak, bu yöntemler hız yerine performansa odaklandığından gerçek zamanlı çalışma için uygun değildir. İki aşamalı tahmin algoritmaları, yüksek doğruluğa ancak düşük performansa sahiptir.

Hâlihazırda önerilen algoritmaların yavaş olmasının sebebi, örnek segmentasyonunun zor bir görev olmasıdır. YOLO [59] gibi tek aşamalı nesne tespit algoritmaları, Faster R-CNN gibi mevcut iki aşamalı dedektörleri, sadece ikinci aşamayı kaldırarak ve kayıp performansı başka yollarla telafi ederek hızlandırabilirken örnek segmentasyonunda böyle bir işlem kolayca uygulanamaz. Ayrıca, iki aşamalı tahmin algoritmaları, nesnelerin maskelerini oluşturmak için özellik yerelleştirmeye bağlıdır.

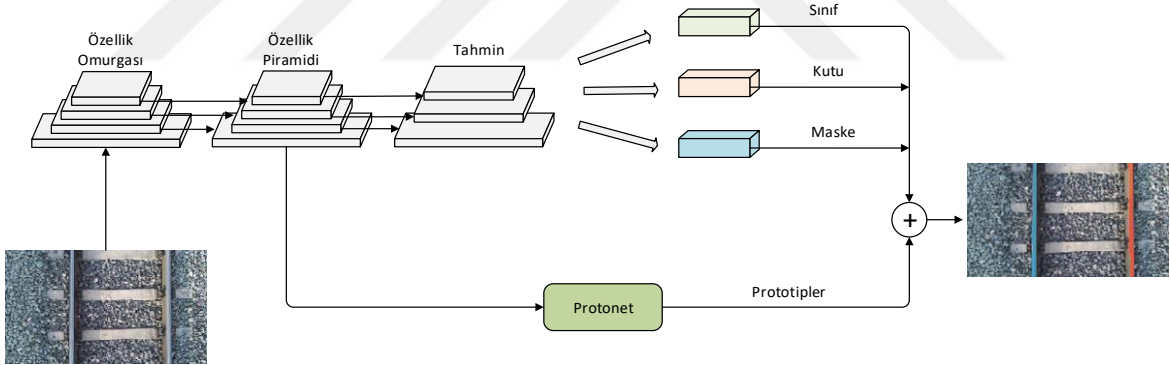
Bu çalışmada, gerçek zamanlı ray takibi yapılacağı için gerçek zamanlı çalışabilen bir segmentasyon yöntemi olan Yolact [60] kullanılmıştır. Yolact, örnek segmentasyonunu iki paralel görev ile yapar. İlk olarak giriş görüntüsü üzerinde prototip maskelerinden oluşan bir sözlük oluşturur. Ardından, örnek başına bir dizi doğrusal kombinasyon katsayısını tahmin eder. Bu iki bileşenden segmentasyon oluştururken her örneğe karşılık gelen tahmini katsayıları kullanarak prototipleri doğrusal olarak birleştirir. Son aşamada ise tahmin bir sınırlayıcı kutu ile kırılır.

Yolact, MS COCO veri kümesinde [61] test edildiğinde gerçek zamanlı kullanıma uygun örnek segmentasyon algoritması olduğu ispatlanmıştır. Diğer örnek segmentasyon algoritmaları ile hız-performans açısın karşılaştırılıp sonuçlar Şekil 2.22'de verilmiştir [60].



**Şekil 2.22.** COCO veri kümesinde çeşitli örnek segmentasyon yöntemleri için hız-performans karşılaştırması

Yolact mimarisi Şekil 2.23'te gösterilmiştir. Özellik çıkarımı oluşturmak için farklı bir omurga kullanan FPN yöntemi ile başlar. Bunlar, ReLU [62] ve maske katsayılarını kullanarak prototip oluşturma dalları oluşturmak ve ardından bunları maske montajı için birleştirmek için kullanılır. Ardından, puan eşliğinde kullanılan birçok parametre görüntüyü filtreler ve bu parametreler çıktı görüntüsü olur ve nesnede bölünmüş bölgeler olarak görüntülenir.



**Şekil 2.23.** Yolact mimarisi

Önerilen Yolact tabanlı ray tespit sistemini tasarımımda ilk olarak İHA, el ile kontrol edilerek demiryolundan görüntü toplanmıştır. Görüntü toplama aşamasından sonra veriler ön işleme adımından geçirilir ve etiketlenir. Bu süreçten elde edilen veri kümesi, sistemi eğitmek için kullanılır. Sistemin yeterli doğrulukta algılama yapması için uzun bir eğitim sürecinden geçmiştir. Eğitim işlemi bittikten sonra, model doğruluk açısından kontrol edilmiştir. Elde edilen sonuç gerçek etiketle eşleştiğinde modelin kullanılabilir olduğuna karar verilmiştir. Elde edilen Sonuç Şekil 2.24'te verilmiştir.



**Şekil 2.24.** Sonuçlar a) Maskelerin elde edilmesi b) İkili maskenin oluşturulması

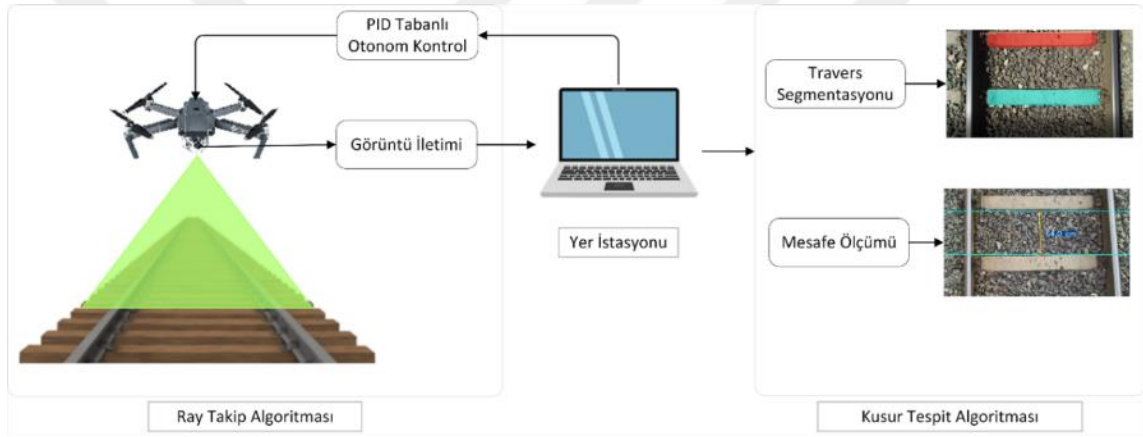
Orta nokta tespiti için ilk olarak Şekil 2.24'te görülen sağ ve sol ray konturlarının merkez noktası hesaplanmıştır. Ardından elde edilen sağ ve sol raya ait merkez noktaların x koordinatlarının orta noktası kırmızı nokta ile işaretlenmiştir. Her görüntü çerçevesi için işaretlenen konum bilgileri, Şekil 2.25'te gösterilmiştir. Burada kırmızı nokta rayların orta merkezini gösterirken yeşil nokta referans noktasını gösterir. Referans noktası kameranın orta noktası olan 240. pikseldir. Tespit edilen iki ray çizgisinin orta konumu ve kameranın konumu kullanılarak, İHA'nın raylara göre konumu hesaplanmıştır. Bu işlem sonucu, İHA'nın merkezinin rayların ortasından ne kadar uzakta olduğunu hesaplanmıştır ve PID denetleyici ile ray takibi yapılmıştır.



**Şekil 2.25.** Orta nokta ve referans noktasının işaretlenmesi

## 2.2. Mask R-CNN Algoritması Tabanlı Travers Boşluklarının Tespiti

Travers eksikliklerinin otonom İHA kullanarak tespiti için tasarlanan sistemin yapısı Şekil 2.26'da gösterilmiştir. Önerilen sistem için rayları 90 derece açıyla yukarıdan gören bir İHA gerekmektedir. Parrot Anafi 4K, kamerasını 180 derece hareket ettirebilmektedir. Bu nedenle çalışmada, Parrot Anafi 4K kullanılmıştır. Önerilen yöntemde Anafi 4K tarafından demiryolundan görüntü toplanacaktır. Ray takibi, alınan görüntü çerçevesinin anlık olarak işlenip anlık olarak İHA'ya pozisyon komutu gönderilmesi ile sağlanmaktadır. Bu işlemler, Parrot Anafi 4K SDK'sı ile yapılabilir. Parrot Anafi 4K ile veriler toplandıktan sonra kusur tespiti aşamasına geçilmiştir. Toplanan görüntüler üzerinde traversler, Mask R-CNN algoritması ile belirlendikten sonra aralarındaki mesafe ölçülmüştür. Elde edilen mesafe bilgileri kullanılarak traverslerdeki eksiklik ve açıklıklar tespit edilmiştir.



Şekil 2.26. Önerilen yöntemin tasarımı

Traversler, trenin geçişi sırasında tekerleklerdeki yükü zemine aktarmak için önemli bir bileşendir. Traverslerin bu görevi nedeniyle, traverslerin arasındaki mesafenin fazla olması veya oluşabilecek eksiklikler önemli sorunlara neden olacaktır. Çalışmanın bu bölümünde, traversler arasındaki mesafenin ölçümü için iki ayrı yöntem önerilmiştir. İki yöntem de Mask R-CNN tabanlıdır. İlk yöntemde mesafe ölçümü için, Mask R-CNN algoritması tarafından elde edilen sınır kutuları kullanılırken diğer yöntemde maske bilgisi kullanılmıştır.

### 2.2.1. Mask R-CNN

Çalışmanın bu bölümünde traverslerin doğru tespiti için Mask R-CNN yapısına dayalı bir algoritma önerilmiştir. Önerilen algoritma, İHA'nın demiryolundan topladığı RGB formatındaki görüntülere uygulanmıştır. En ünlü segmentasyon algoritmalarından biri olan Mask R-CNN, iki aşamalı bir çerçeveye sahip bölgesel bir evrişimli sinir ağıdır [63]. Mask R-CNN algoritması,

traversleri, traversleri sınırlayan kutuları, her maskenin etiketlerini ve her travers için nesnellik değerini çıktı olarak içeren maskeler üretmektedir.

Mask R-CNN çerçevesi, Faster R-CNN altyapısı ile tasarlanmış bölge tabanlı bir evrişimsel sinir ağıdır. Bir bölge teklif ağı aracılığıyla oluşturulan bölge tekliflerine dayanmaktadır [64]. Mask R-CNN, tespit edilecek nesneyi yüksek doğrulukla algılayabilen ve nesneyi doğru bir şekilde segmente edebilen bir algoritmadır. Faster R-CNN'den farklı olarak, ROI-Pooling işlemi Mask-RCNN tarafından değiştirilir. Bu işlem, doğru numune segmentasyon maskelerinin oluşturulmasını sağlar. Diğer fark, Mask-RCNN'nin, oluşturulacak örnek segmentasyonları oluşturmak için küçük, tamamen evrişimli bir sinir ağı eklemesidir. Segmentasyon, örnek segmentasyonu olarak yapılır. Örnek bölütlemede sınıftan bağımsız bölütleme yapılır. Bu çalışmada hedef, karmaşık bir demiryolu görüntüsündeki traversleri tespit etmek ve segmentlere ayırmak olduğundan, sonuçlar yalnızca travers sınıfına sahiptir.

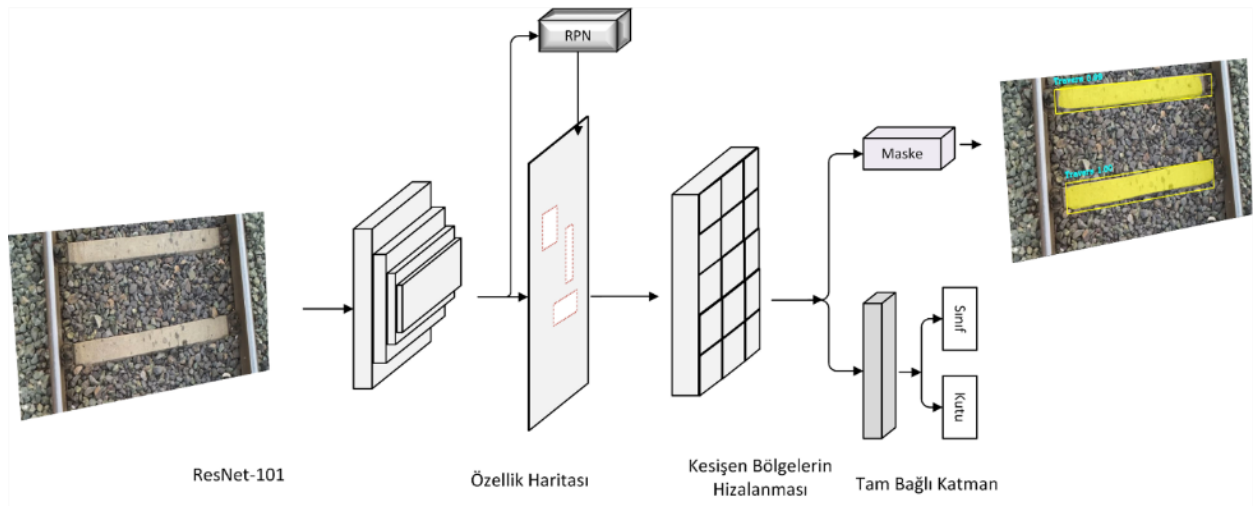
Mask-RCNN, iki aşamalı bir çerçeveye sahip bölgesel evrişimli sinir ağıdır. İlk aşama, görüntüyü işlemek için bölge teklif ağını (RPN) kullanır ve aday nesne sınırlayıcıları oluşturur. Ardından nesne sınırlayıcı kutularını ikinci aşamaya teslim eder. İkinci aşamada ise aday nesne sınırlayıcı kutuları sınıflandırılır, rafine sınırlayıcı kutular üretilir ve maskeler oluşturulur. Sınıf bilgisi, nesnenin ait olduğu sınıfı gösterir. Sınırlayıcı kutu bilgisi, nesneyi sınırlayan kutunun boyutu ve konumudur. Maske bilgisi ise nesnenin segmentasyonudur. Mask R-CNN'in mimarisi Şekil 2.27'deki gibidir.

Uygulamada, giriş görüntüsünün uyuyan özellikleri çıkarılır. Bu işlem için ResNet-101 omurgası kullanılmaktadır. Ardından, RPN aracılığıyla özellik haritasında İlgi Bölgesi (RoI) oluşturulur. Ayrıca, Konumları tam olarak korumak ve özellik haritasını sabit boyutta tutmak için İlgi Bölgesi Hizalaması (RoI Hizalaması) kullanılır. Ağın sonunda, sınırlayıcı kutu algılama bölümünde bulunur ve sınıflandırılır. Travers maskesi, görüntü üzerinde segmentasyon olarak Full Convolution Network (FCN) tarafından oluşturulur. Son olarak maske bilgisi ve sınıf bilgisi ayrıştırılır. Maske ağ kafası, sınıf tahmini ağ kafasından bağımsız olarak maskeyi tahmin eder. Bu, çoklu görev kaybı fonksiyonunun [64] kullanılmasını gerektirir.

Mask R-CNN algoritmasında kullanılan kayıp fonksiyonu denklem 2.14'te verilmiştir.

$$L = L_{cls} + L_{reg} + L_{mask} \quad (2.14)$$

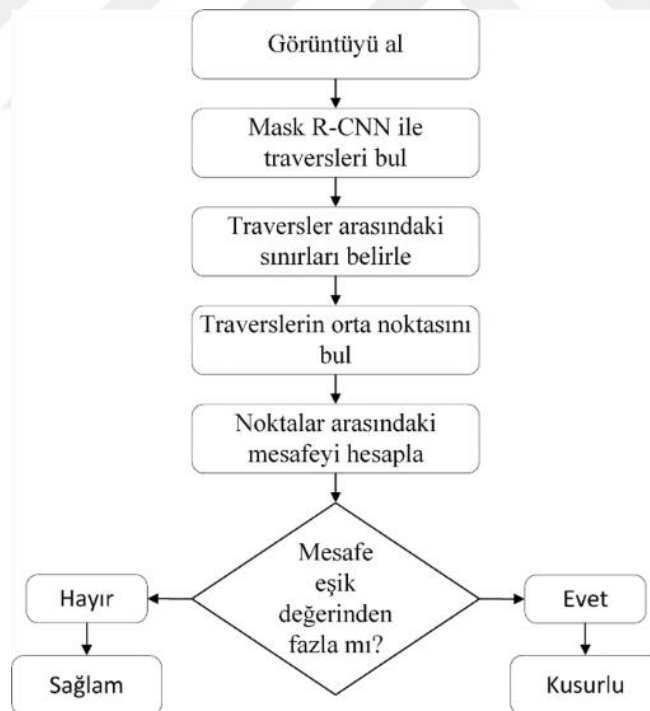
Denklem 2.14'te,  $L_{cls}$ , sınıflandırma sonucunun kayıp fonksiyonudur,  $L_{reg}$ , sınıflandırma fonksiyonunun kayıp fonksiyonudur ve  $L_{mask}$ , segmentasyon sonucunun kayıp fonksiyonudur.



**Şekil 2.27.** Mask R-CNN mimarisi

### 2.2.2. Maske Sınırlarına Dayalı Ölçüm Yöntemi

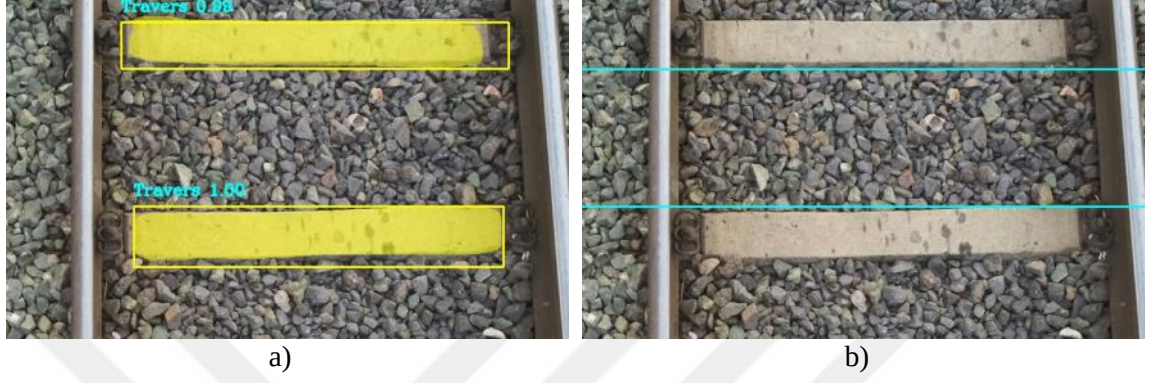
Tezin bu bölümünde, Mask R-CNN tarafından üretilen maske sınırlarına dayalı ölçüm yöntemi önerilmiştir. Önerilen yöntemle ait akış diyagramı, Şekil 2.28’de verilmiştir.



**Şekil 2.28.** Travers eksikliklerinin tespiti için önerilen algoritma

Eğitim için görüntüler demiryolundan Parrot Anafi 4K ile toplanmıştır. Çeşitli açılarda 800 adet görüntü toplanmıştır. Demiryolu görüntüleri 416x219 çözünürlükte JPEG formatına dönüştürülmüştür. Etiketlenen görüntüler 25 adım eğitilmiştir. Uygulama, test için ayrılan rastgele

bir görüntü için çalıştırıldığında Şekil 2.29'da gösterildiği gibi sonuç elde edilmiştir. Burada traversler arası mesafeyi ölçebilmek için üstte kalan traversin alt sınırı ile altta kalan traversin üst sınırı belirlenmiştir. Bu sınırlar Mask R-CNN algoritmasının çizdiği sınırlardan faydalanılarak çizilmiştir. Sınırlar, Şekil 2.29.b'de gösterildiği gibi belirlenmiştir.

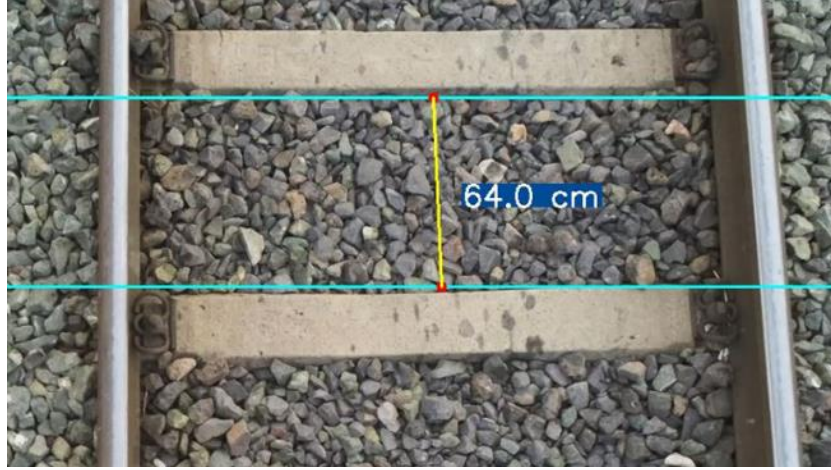


Şekil 2.29. a) Mask R-CNN algoritmasının çıktısı b) Ölçüm için sınırların belirlenmesi

Görüntüdeki iki travers arasındaki mesafeyi ölçebilmek için traversler arasındaki mesafe standartlara uyan bir referans görüntü kullanarak bir kalibrasyon yapılmalıdır. Traversler, belirli açıklıklarda dizilirler. Konvansiyonel hatlarda traversler arasındaki mesafesi 63-64 cm olarak uygulanmaktadır. Bu nedenle, referans olarak alınan görüntüdeki mesafe 64 cm olarak belirlenmiştir. Referans görüntüden Denklem 2.15'teki oran elde edilmiştir.

$$\text{Oran} = \frac{\text{düşey\_piksel\_sayısı}}{\text{bilinen\_genislik}} \quad (2.15)$$

Burada bilinen genişlik 64 cm'dir. Aradaki piksel ise Şekil 2.29.b'de çizilen sınırların arasında dikeyde bulunan piksel sayısıdır. Oran değeri sadece bir kez hesaplanır ve diğer test görüntülerine bu oran uygulanarak ölçüm yapılır. Ölçüm için üst ve alt sınırlarda birer nokta belirlenmelidir. Noktalar, Şekil 2.29.a'daki sınırların tam orta noktası olarak belirlenmiştir. Belirlenen noktalar kırmızı daire ile işaretlenmiştir. Bu noktalar arasındaki mesafenin ölçümü Şekil 2.30'da gösterildiği gibi ölçülmüştür.



Şekil 2.30. Referans görüntüde iki nokta arası ölçüm sonucu

### 2.2.3. İkili Maskeye Dayalı Travers Eksikliklerinin Tespiti

Bu bölümde, traversler arasındaki mesafeyi ölçmek için kullanılan yaklaşımımızı detaylandırıyoruz. Bu bölümün amacı, giriş görüntüsü üzerinde traverslerin konumunu belirlemek ve traversler arasındaki mesafeyi hesaplamaktır. Bu amaçla, kullanılan algoritmanın ilk adımı Mask R-CNN ile traverslerin ikili maskelerini elde etmektir. Ortaya çıkan görüntüde, birler ve sıfırlar sırasıyla beyaz ve siyah alanları temsil eder. İkinci adımda, uyuyanların tepe noktalarını tespit etmek için görüntü 90 derece döndürülüp tüm piksellerin toplamı alınmıştır. Ortaya çıkan tepe noktaları, traverslerin konumunu temsil eder. Son aşamada ise pikler arası mesafe hesaplanarak kayıp travers olup olmadığı kontrol edilmiştir.

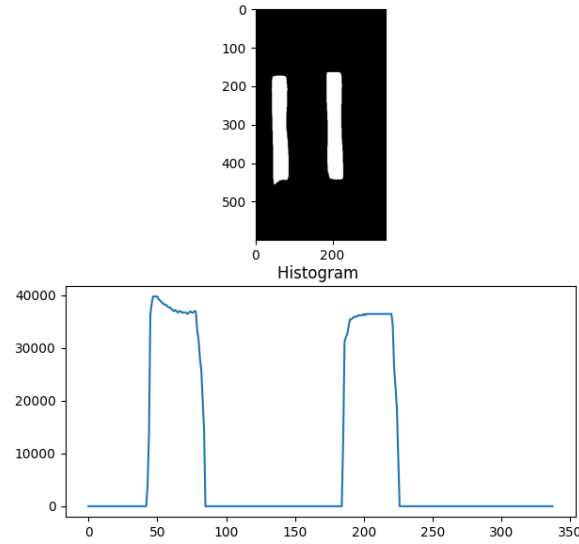
Bu çalışmanın ilk aşamasında, Parrot Anafi 4K ile demiryolundan görüntüler toplanmıştır. Toplanan görüntüler etiketlendi ve json formatında kaydedildi. Daha sonra Mask R-CNN modeli eğitilir ve ikili maske elde edilir. İkili maskeden elde edilen histogram tepe noktaları kullanılarak traverslerin konumu belirlenmiştir. Önerilen yöntem, üst traversin alt sınırı ile alt traversin üst sınırı arasındaki mesafenin bulunmasına dayanmaktadır. Bu yöntemle toplanan görüntüler üzerinde travers eksiklikleri tespit edilmiştir. Traversler, Otsu yöntemi ve Mask R-CNN'ye dayalı iki farklı yöntemle belirlenmiş ve karşılaştırılmıştır.

Traversleri tespit etmek için Mask R-CNN tabanlı ölçüm yöntemi önerilmiştir. İlk olarak RGB renk uzayında elde edilen görüntü Mask R-CNN algoritmasına girdi olarak verilmiştir. Mask R-CNN algoritması ile maske, kutu ve etiketli çıktı görüntüsü elde edilir. Sonuç, Şekil 2.31.a'da gösterilmektedir. Maske R-CNN sonucunda elde edilen Şekil 2.31.b'deki ikili maske görüntüsü, travers sınırlarının daha iyi tespiti için kullanılmıştır.



**Şekil 2.31.** Mask R-CNN sonucu elde edilen görüntü a) Maske, kutu, etiket bilgisi b) İkili maske

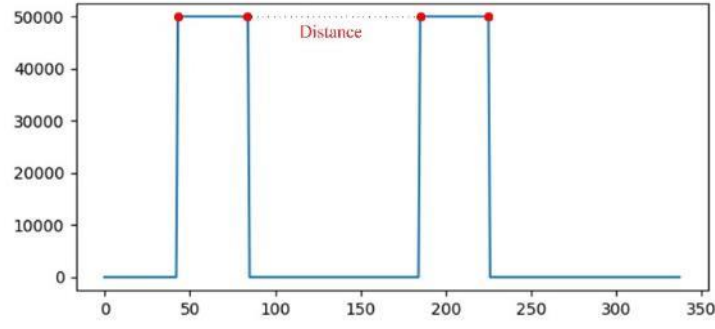
Görüntüdeki bir traversin noktalarını belirlemek için önce görüntünün bir histogramı elde edilir. Histogram, görüntünün X eksenı yönü boyunca en büyük yoğunluğa sahip piksel konumunda bir tepe gösterir. Şekil 2.32, traverslere karşılık gelen maksimum değerlerle birlikte bir görüntü histogramı örneğini göstermektedir.



**Şekil 2.32.** x ekseninde piksel yoğunluğu toplamının gösterimi

Görüntüdeki traversin konumlarını belirlemek için, dikey yöndeki piksel yoğunluğunun toplamı, Şekil 2.32'de çizildiği gibi x eksenine yansıtılır. 1 ve diğer pikseller 0 ile gösterilir. Başka bir deyişle, traversli ve traverssiz bölge Şekil 2.31.b'de gösterildiği gibi net ve kolay bir şekilde ayırt edilebilir.

Bu nedenle, traversler Şekil 2.32'deki histogram grafiğinde belirgin bir şekilde gösterilmektedir. Ancak traverslerin tepe kenar çizgilerini tam olarak elde edebilmek için histogram grafiği tepe noktası tek bir değer olacak şekilde yeniden çizilmiştir. Sonuç, Şekil 2.33'te verilmiştir.

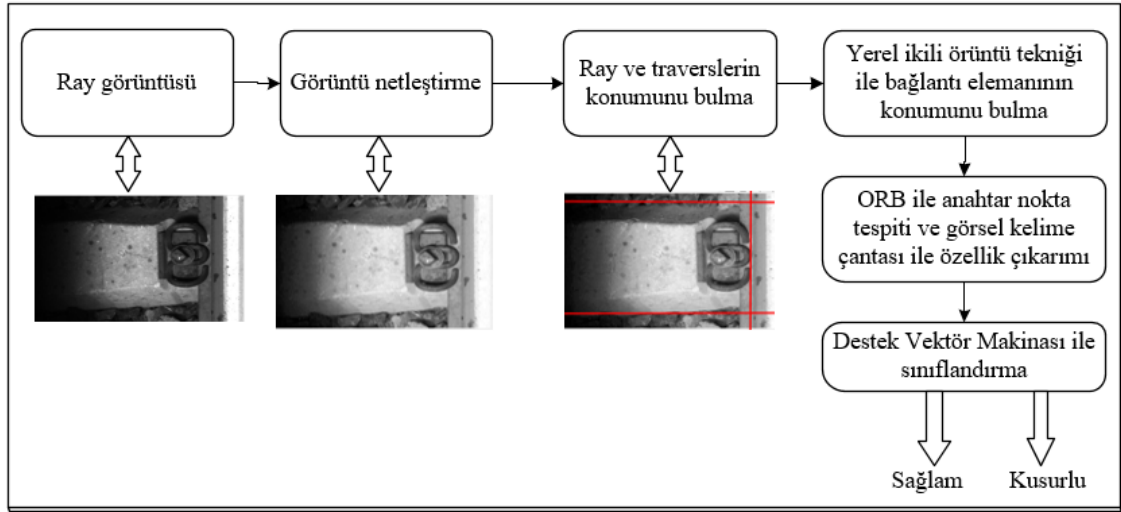


Şekil 2.33. Histogramın yeniden çizilmesi

Histogram, tepe noktaları sabit tutularak yeniden çizilmiştir. Köşelerde gösterilen kırmızı noktalar, traverslerin başlangıç ve bitiş noktalarını temsil etmektedir. İlk traversin bitiş noktası ile ikinci traversin başlangıç noktası arasındaki piksel farkı hesaplanarak kayıp travers olup olmadığı tespit edilmiştir.

### 2.3. Görsel Kelime Çantası'na Dayalı Kusur Tespit Yöntemi

Demiryolu bileşenlerinden biri olan bağlantı elemanları, diğer bileşenleri bir arada tutmak için kullanılmaktadır. Bu kullanımıyla, rayın ve traversin uygun pozisyonlarda sabitlenmesini sağlamaktadır. Kırık veya eksik bağlantı elemanları diğer bileşenlerin pozisyonunun bozulmasına sebep olarak kazalara sebebiyet verebilir. Bu nedenle, demiryolu bakımında bağlantı elemanlarının önemli bir yeri vardır. Bu çalışmada, bilgisayarlı görmeye dayalı bağlantı elemanı kusur tespit yöntemi önerilmiştir. Önerilen yöntemde ilk aşama görüntü ön işleme işlemidir. Görüntü ön işleme ile görüntü netliği iyileştirilmektedir. Bu işlemden sonra, ray ve traversler için konum tespiti yapılmıştır. Bağlantı elemanının görüntüdeki yerinin tespiti önemli bir aşamadır. Yer tespiti için çizgi yerel ikili örüntü (ÇYİÖ) tekniği kullanılmıştır. Bağlantı elemanının görüntüdeki tespitinden sonra, ORB algoritması ile bağlantı elemanına ait özellikler çıkarılıp destek vektör makinası ile sınıflandırılmıştır. Destek vektör makinası, bağlantı elemanlarını sağlam ve kusurlu olmak üzere sınıflandırmaktadır. Önerilen yöntemin akış diyagramı Şekil 2.34'te verilmiştir.



Şekil 2.34. Önerilen yöntemin akış diyagramı

### 2.3.1. Görüntü Önileme Adımı ve İyileştirme

Demiryolundan toplanan görüntüler düşük kontrast değerine sahiptir. Bu nedenle, görüntü önileme ile görüntü netleştirme yapılmıştır. Giriş görüntüsü RGB formatındadır. Görüntü gri seviyeye dönüştürülürken kullanılan denklem, R,G ve B değerleri için sırasıyla  $k_r$ ,  $k_b$  ve  $k_g$  katsayıları değerleri için Denklem 2.16’da verilmiştir.

$$I(x,y) = k_r R(x,y) + k_b B(x,y) + k_g G(x,y) \quad (2.16)$$

İyileştirilmiş bir görüntü elde edebilmek için katsayıların uygun olarak belirlenmesi gerekmektedir. Görüntü, gri seviyeye çevrildikten sonra varyans Denklem 2.17’deki gibi hesaplanır.

$$\sigma_f^2 = \text{varyans}((I - I_{\min})/(I_{\max} - I_{\min})) \quad (2.17)$$

Uygulamada,  $k_b$  değeri 1 olarak belirlenmiştir.  $k_r$  ve  $k_g$  değerleri ise Denklem 2.18’deki denklemle hesaplanır.

$$(\hat{k}_r, \hat{k}_g) = \arg \max_{k_r, k_g} (\sigma_f^2) \quad (2.18)$$

Elde edilen yeni katsayılarla yeni görüntü, Denklem 2.19’deki gibi hesaplanır.

$$\hat{I}(x,y) = \hat{k}_r R(x,y) + B(x,y) + \hat{k}_g G(x,y) \quad (2.19)$$

Son olarak yüksek kontrastlı görüntü Denklem 2.20’deki gibi hesaplanır.

$$J = (\hat{I} - \hat{I}_{\min}) / (\hat{I}_{\max} - \hat{I}_{\min}) \quad (2.20)$$

Görüntü önışleme adımlarından sonra görüntüdeki ray ve traverslerin konumları tespit edilmiştir. Ray ve travers tespiti için görüntünün ikili görüntüye dönüştürülmesi gerekmektedir. Bu işlem için Otsu metodu kullanılmıştır [65]. Dikey projeksiyonun oluşturulabilmesi için ikili görüntüye Denklem 2.21'deki işlem uygulanır.

$$P(j) = \sum_{i=1}^w S(i, j) \quad i=1, \dots, w, j=1 \dots h \quad (2.21)$$

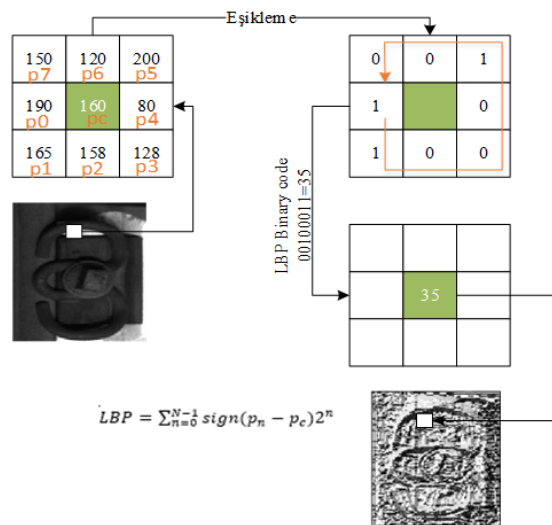
Burada  $S(i, j)$ , segmente edilen görüntünün piksel değerini ifade etmektedir. Bu işlem sonucunda ikili görüntü oluşmaktadır. Elde edilen  $P(j)$ 'nin ardışık farkları Denklem 2.22 ile alınmaktadır.

$$DP(j) = P(i + 1) - P(i), i = 1, 2, \dots, w \quad (2.22)$$

Denklem 2.22'de ardışık farklar elde edilmektedir. Bu denklem ile elde edilen en büyük iki değer ray sınırlarını temsil etmektedir. Aynı şekilde görüntüye yatay uygulanırsa elde traversler de tespit edilebilir.

### 2.3.2. Bağlantı Elemanı Konumlarının Çizgi Yerel İkili Örüntü(ÇYİÖ) Tabanlı Tespiti

Yerel ikili örüntü yöntemi, merkezdeki piksel ile çevre piksellerin karşılaştırılarak yeni değer üretildiği bir yöntemdir. Bu yöntemde merkez piksel çevre pikselden küçükse ilgili piksel 1, büyükse 0 alınır [66]. Merkez pikselin değeri, her çevre bitin ağırlıklı toplamıdır. Yerel ikili örüntü yöntemine ait görsel Şekil 2.35'te verilmiştir.

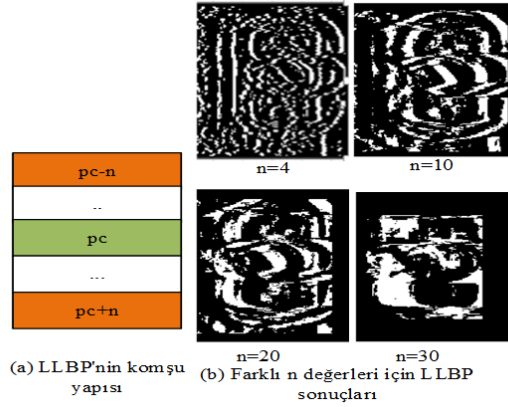


Şekil 2.35. Bağlantı elemanı görüntüsünde yerel ikili desen

Yerel ikili örüntü yöntemi kullanılarak bağlantı elemanı tespiti oldukça zor bir işlemdir. Bu nedenle bu çalışmada çizgi yerel ikili örüntü yöntemi kullanılarak bağlantı elemanı tespiti yapılmıştır. ÇYİÖ yönteminin farkı karşılaştırma işleminin çizgi yönünde yapılmasıdır [67]. Denklem 2.23'te, yöntem verilmiştir.

$$LLBP = \text{sign}(p_{c-n} - p_c) \times \text{sign}(p_{c+n} - p_c) 2^0 \quad (2.23)$$

Denklem 2.23'te  $p_c$  merkez pikseli temsil ederken  $P_{c \pm n}$  iki komşusuna karşılık gelmektedir. Çizgi yerel ikili örüntü,  $n$  adında bir parametreye sahiptir. Burada,  $n$  parametresinin değeri büyüdükçe görüntüdeki bağlantı elemanı geniş görünür. Eğer  $n$  değeri küçükse oluşan gürültü fazladır. Bu durumda, bağlantı elemanının tespiti zor olur. Şekil 2.36'da,  $n$  için farklı değerler verilerek ÇYİÖ yönteminin sonuçları gösterilmiştir.



Şekil 2.36. Farklı  $n$  değerleri için ÇYİÖ uygulaması

Şekil 2.36'da gösterildiği gibi  $n$  değeri küçük seçilirse tamamen gürültülü bir görüntü elde edilmektedir. Bu nedenle, bağlantı elemanının tespiti zorlaşmaktadır. Görüntü üzerinde kenar noktaları çizilerek bağlantı elemanı tespit edilmiştir.

### 2.3.3. Özellik Çıkarımı ve Kusur Sınıflandırma

Özellik çıkarımı, ORB algoritması kullanılarak yapılmıştır. ORB algoritması, SIFT ve SURF'a etkili bir alternatif olarak Ethan Rublee vd. tarafından 2011 yılında geliştirilmiştir [68]. ORB algoritması, görüntü üzerine uygulandığında görüntüdeki kilit noktalar bulunmaktadır. Görüntü içerisindeki kilit noktalar, piksellerin ani değiştiği noktalardır.

ORB algoritması, FAST anahtar noktası algılayıcısı ve BRIEF tanımlayıcısını temel alır. Bu tekniklerin her ikisi de iyi performans gösterir ve düşük maliyetlidir. Bu nedenle ORB, gerçek zamanlı kullanıma uygundur. ORB'de ilk olarak FAST anahtar noktası algılayıcısı kullanılarak

anahtar nokta tespiti yapılır. Daha sonra bu anahtar noktalara ait yönelimler bulunur. Son aşamada ise BRIEF tanımlayıcısı kullanılarak tanımlayıcılar bulunur.

FAST algoritmasında yapılan ilk işlem, I yoğunluğa sahip bir p pikselinin belirlenmesidir. Bu işlem p pikselinin köşe noktası olup olmadığı belirlenir.

$$N = \sum_{x \in \text{circle}(p)} f(I(p), I(x)) \quad (2.24)$$

Denklem 2.24'teki f fonksiyonu Denklem 2.25'teki gibi hesaplanır.

$$f = \begin{cases} 1, & |I(x) - I(p)| > \epsilon_d \\ 0, & \text{diğer} \end{cases} \quad (2.25)$$

Burada I(p), çemberdeki merkez pikselidir. I(x) ise çember içinde bulunan herhangi bir noktaya karşılık gelir. Noktalar arasındaki mutlak fark  $\epsilon_d$  değerinden büyük ise 1, değilse 0 değeri verilir. Yukarıdaki denklem ile verilen bir eşik değerinden daha büyük olan piksel sayısı olan N değeri hesaplanır. Giriş görüntüsünün I(x,y) parçası için, moment değerleri Denklem 2.26'daki gibi hesaplanır.

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y) \quad (2.26)$$

Hesaplanan moment değerleri kullanılarak C merkezi Denklem 2.27'deki gibi hesaplanır.

$$C = \left( \frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (2.27)$$

$\theta$ , parçanın yönünü ifade etmektedir. Bu değer, Denklem 2.28'deki gibi bulunur.

$$\theta = \arctan(m_{01}, m_{10}) \quad (2.28)$$

ORB algoritmasında ikinci aşama ise, FAST ile bulunan kilit noktalar alınıp özellik vektörlerine dönüştürülür. Özellik vektörlerinin oluşturulmasında BRIEF algoritması kullanılır. BRIEF, ikili dizeleri doğrudan bulmak için kullanılan ikili özellik tanımlayıcısıdır. İkili

tanımlayıcılar oluştururken algılanan noktaların çevresindeki iki pikselin konumu arasındaki yoğunluk karşılaştırılır.

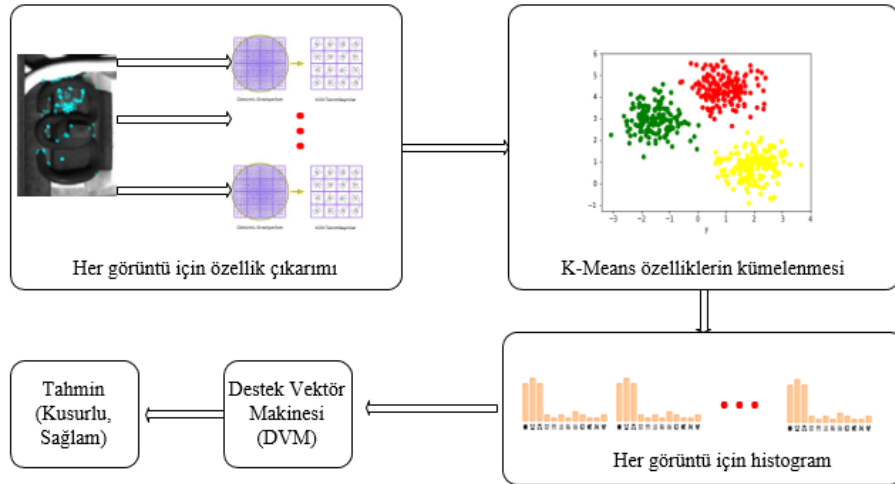
BRIEF algoritması, benzersiz olarak bir dizi (x, y) konumu seçer. Ardından bu noktalarda piksel yoğunluğu karşılaştırmaları yapılır. Bu işlem,  $P(x)$ 'in, x noktasındaki P'nin yoğunluğu ve  $P(y)$ 'nin, y noktasındaki P'nin yoğunluğunu temsil ettiği varsayılırsa Denklem 2.29'da gösterilmiştir.

$$\tau(p; x, y) = \begin{cases} 1, & p(x) < p(y) \\ 0, & p(x) \geq p(y) \end{cases} \quad (2.29)$$

Özellik vektörü Denklem 2.30'daki gibi oluşturulur.

$$f_n(p) = \sum_{1 \leq i \leq n} 2^{i-1} \tau(p; x, y) \quad (2.30)$$

ORB kullanılarak elde edilen özellikler, Bag of visual words (BOVW) tarafından kullanılarak kusur sınıflandırma yapılmıştır. Bag of visual words kavramı, görsel kelime çantası anlamına gelmektedir. Bu yöntem, metin analizi için önerilen Bag of words (BOW) yönteminden türetilen ve bilgisayarla görme uygulamaları için kullanılan bir yöntemdir. BOVW'de görüntü analizi için görsel özellikler kümelenir. Görsellerden kelime çantası elde edebilmek için önce özellikler elde edilmelidir. Bir görüntüden SIFT, SURF ve ORB gibi algoritmalar vasıtasıyla özellik çıkarımı yapılabilmektedir. Bu tez kapsamında, özellik çıkarımı için yukarıda bahsedildiği gibi ORB algoritması kullanılmıştır. Özellikler, destek vektör makinası ile sınıflandırılarak bağlantı elemanı görüntüsü için sınıflandırma yapılmıştır. Yöntemi aşamaları, Şekil 2.37'de verilmiştir.



Şekil 2.37. Kelime çantası modeli akış diyagramı.

Giriş görüntülerinden çeşitli özellikler elde edilir. Elde edilen özellikler kümelenmelidir. Her görüntü için frekans histogramı oluşturulur. Görsel kelime çantası, bu histogramlardan oluşmaktadır. Destek Vektör Makinesi, bu histogramları kullanarak bağlantı elemanlarını sınıflandıracaktır.



## 2.4. YOLOv4 ve Bulanık Mantık Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti

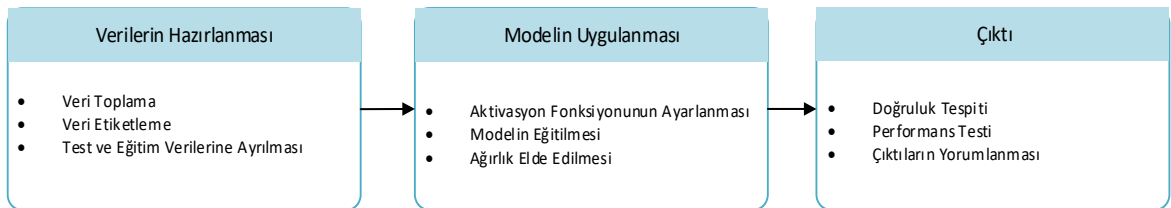
Tezin bu bölümünde, deforme bağlantı elemanlarının tespiti için YOLOv4 ve bulanık mantık kullanılmıştır. Çalışmada, bağlantı elemanı görüntüsü kırılma potansiyeli olan bölgelerden 6 farklı sınır belirlenerek etiketlenmiştir. Bu işlemin amacı, bağlantı elemanında daha detaylı inceleme yapabilmek ve bağlantı elemanına ait farklı bölgelerin detaylı incelenmesini sağlamaktır, çünkü demiryolu üzerindeki bağlantı elemanları treninin oluşturduğu titreşim gibi etkiler ile zarar görebilir. Hasarlı bağlantı elemanlarının bazı bölgeleri etkinliğini yitirmiş olabilir. Şekil 2.38’de bazı bağlantı elemanı görüntüleri gösterilmiştir. Görüntüyü farklı parçalara bölmek diğer avantajı ise tanıma doğruluğunu artırmaktır. Bağlantı elemanını oluşturan 6 farklı parçaya YOLOv4 algoritması uygulanarak 6 parçanın güven değerleri elde edilmiştir. Elde edilen 6 farklı güven değeri, 6 girişli tek çıkışlı bulanık mantık yapısına verilerek yüzde cinsinden sonuç değeri üretilmiştir. Bu değer bağlantı elemanının sağlık durumu hakkında bilgi vermektedir. Bulanık mantık tasarımı için Python üzerinde Scikit-Fuzzy kullanılmıştır.



Şekil 2.38. Kusurlu bağlantı elemanı örnekleri

### 2.4.1. Bağlantı Elemanlarının Tespiti İçin YOLOv4 Algoritmasının Uygulanması

YOLOv4 ile bağlantı elemanı tespiti için yapılan işlemler Şekil 2.39’da gösterilmiştir. Model uygulanmadan önce verilerin hazırlanması ve ön işlemeden geçirilmesi gerekmektedir.



Şekil 2.39. Yapılan işlemlerin akış diyagramı

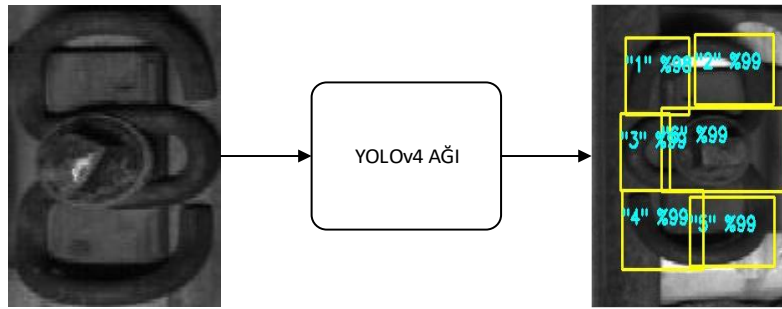
Çalışmada kullanılan veri kümesi demiryolundan toplanan görüntülerden kırılarak elde edilmiştir. Farklı aydınlatma koşullarından toplanan görüntüler veri kümesindeki çeşitliliği sağlamaktadır. 2100 bağlantı elemanı görüntüsü eğitim için kullanılmak üzere etiketlenip eğitimde kullanılmıştır. Görüntüler üzerinde bağlantı elemanı üzerindeki hasar oluşabilecek noktalar belirlenerek 6 sınır belirlenmiştir. Seçilen bölgelerin sınırları Şekil 2.40.b’de gösterilmiştir.



Şekil 2. 40. a) Orijinal görüntü b) Etiketli görüntü

YOLOv4, darknet ağı üzerinde geliştirilmiş bir nesne tespit algoritmasıdır. YOLOv4 ağı, hızlı çalışması nedeniyle birçok uygulamada kullanılır. YOLOv4 algoritmasının hızlı olma sebebi tek aşamalı nesne dedektörü olmasıdır. Nesne tespit algoritmaları, bir ve iki aşamalı olmak üzere iki türden oluşur. İki aşamalı nesne tespit algoritmalarında önce önemli bölgeler tespit edilip bölgeler sınıflandırılarak nesnenin o bölgede olup olmadığı belirlenir. YOLOv4, tek aşamalı bir nesne tespit algoritmasıdır. R-CNN, Fast R-CNN gibi iki aşamalı nesne tespit algoritmalarından daha doğru ve daha hızlı çalışmaktadır. Bu nedenle bu tezde nesne tespiti için YOLOv4 tercih edilmiştir.

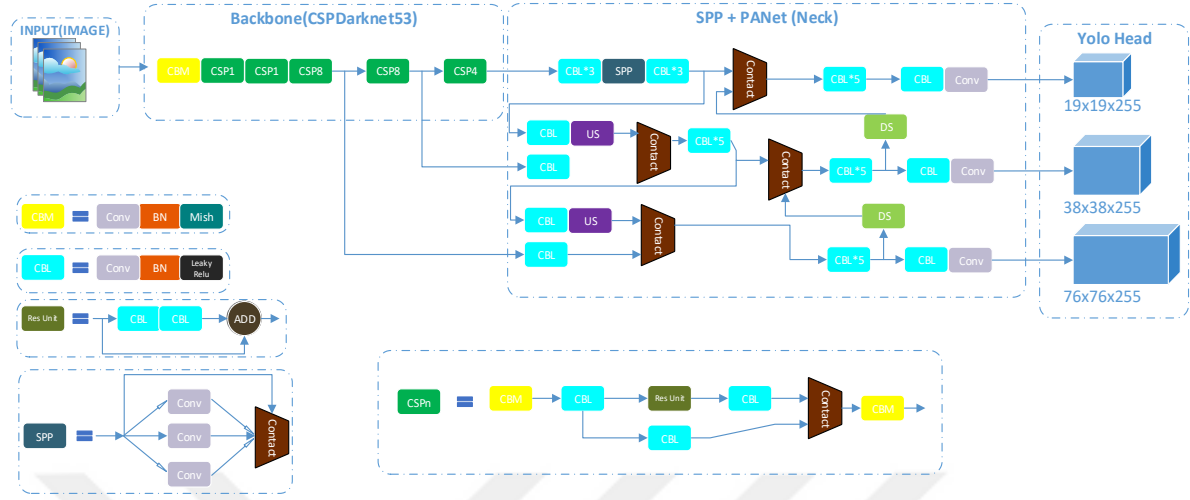
Eğitim kümesindeki veriler kullanılarak eğitim yapıldıktan sonra, YOLOv4 ağına bir bağlantı elemanı görüntüsü verildiğinde çıkış olarak görüntü üzerinde bulunan parçaların tespiti Şekil 2.41'deki gibi yapılmıştır.



Şekil 2. 41. YOLOv4 ağı giriş çıkış örneği

YOLOv4 ağı, üç bölümden oluşmaktadır. Bunlar; omurga, boyun ve kafadır. Omurga, farklı görüntü tanecikleri üzerinden özellik toplayan ve oluşturan evrimsel sinir ağıdır. Boyun, omurgada elden edilen özelliklerin toplandığı ve bu özelliklerin nesne tespiti için başa beslendiği ikinci aşamadır. Kafa, sınırlayıcı kutuların sınıflandıran tahmini içermektedir. Kafa bölümünde,

sınırlayıcı kutuların koordinatları hakkında bilgi sağlanır. Kafa, Etiket, genişlik, yükseklik, merkez ve tahmin puanlarını içermektedir. YOLOv4 mimarisi Şekil 2.42’de gösterilmiştir.



Şekil 2. 42. YOLOv4 mimarisi

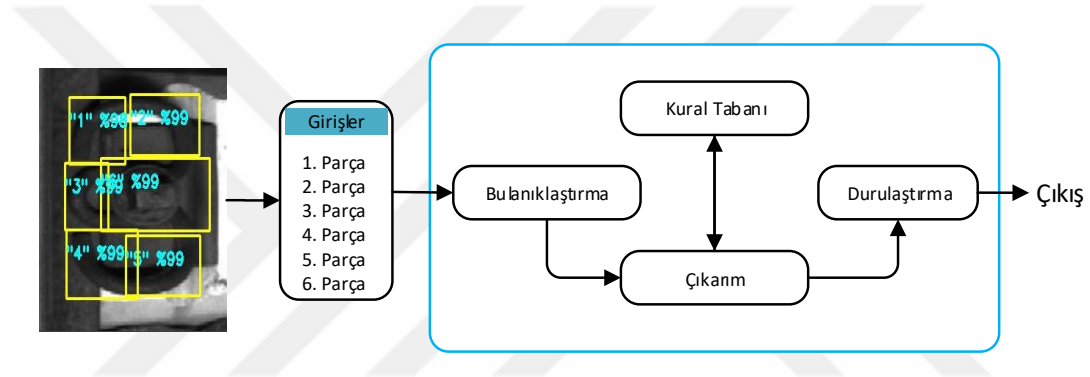
#### 2.4.2. Kusur Tespiti İçin Tasarlanan Bulanık Mantık Yapısı

Giriş olarak verilen bağlantı elemanı görüntüsüne YOLOv4 algoritması uygulanıp 6 farklı güven değeri oluşturulmuştur. Elde edilen güven değerlerini kullanarak bağlantı elemanının sağlık durumu hakkında yorum yapabilmek zor bir görevdir. Ayrıca, eğer bağlantı elemanı sayısı fazlaysa tüm bağlantı elemanı görüntülerinin incelenmesi zaman alacaktır. Bu nedenle elde edilen güven değerleri kullanılarak bulanık mantık tabanlı yöntem geliştirilmiştir. Bulanık mantık, bu tür çözülmesi zor problemlerin, insan beynine benzer tahminler yaparak çözümü için kullanılmaktadır [74]. Bu çalışmada, bağlantı elemanının sağlık durumu hakkında yüzde cinsinden değer üretmek için kullanılmıştır.

Bulanık mantık, Lotfi Zadeh tarafından 1965 tarihinde modern bilgisayarın dayandığı olağan "doğru veya yanlış" (1 veya 0) mantığından ziyade "doğruluk derecelerine" dayalı bir hesaplama yaklaşımıdır [75]. Bulanık Mantık, insanın düşünme yapısına benzeyen bir akıl yürütme yöntemidir. Bulanık mantık yaklaşımında, dijital değerler "0" ve "1" arasındaki tüm ara olasılıkları içeren karar verme şeklini taklit eder. Yapay zekâ sistemlerinde, insan muhakeme yeteneğini taklit etmek için bulanık mantık kullanılır. Bulanık mantık, açık kesinlikler ve belirsizlikler olmadan yani, kesin olmayan verilerle kararlar için kullanılmaktadır. Aynı zamanda, birden çok girişe göre makine çıkışlarının düzenlenmesi ve kontrol edilmesinde kullanılmaktadır. Bağlantı elemanlarının sağlık durumu için düşünülürse, giriş olarak güven değerleri verilip çıktı durumu olarak "eksik", "orta hasarlı", "az hasarlı", "sağlam" gibi sonuçlar üretilecektir.

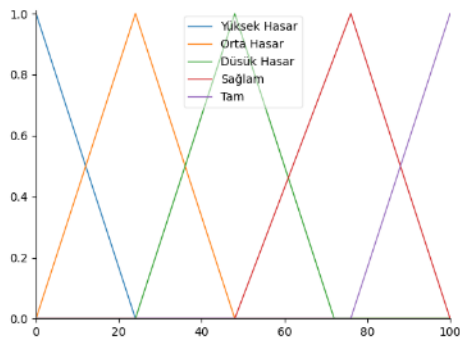
Bulanık mantık mimarisi dört bölümden oluşur. Bunlar; bulanıklaştırma birimi, çıkarım birimi, kural tabanı ve durulaştırma birimidir. Bulanıklaştırma biriminde, giriş bilgileri dilsel

niteleyicilere dönüştürülür. Bulanıklaştırma biriminde genellikle net girdiler, sensörler tarafından ölçülen ve sıcaklık, basınç, gibi işlemler için kontrol sistemine geçirilen tam girdilerdir. Çıkarım birimi, bulanık girdinin kurallara göre eşleşme derecesini belirler ve girdiye göre hangi kuralların çalıştırılacağını belirler. Kural birimi, dil bilgisine dayalıdır. Karar verme sistemini yönetmek için uzmanlar tarafından oluşturulan kurallar kümesini ve “eğer-değilse” koşullarını içermektedir. Bulanık mantık üzerine son gelişmeler ile bulanık denetleyicilerin tasarımı ve ayarlanması için bazı yöntemler sunulmaktadır. Bu gelişmeler, oluşturulan kuralların sayısını azaltmaktadır. Son bölüm ise durulaştırma birimidir. Durulaştırma biriminde, çıkarım birimi tarafından elde edilen bulanık kümeleri kesin bir değere dönüştürülür. Kullanılan bazı durulaştırma yöntemleri vardır ve en uygun olanı tercih edilmelidir. Durulaştırma işleminde bulanık değerlerin insanlar tarafından anlaşılabilir hale getirilir. Kullanılan bulanık sistemin blok diyagramı Şekil 2.43’teki gibidir.

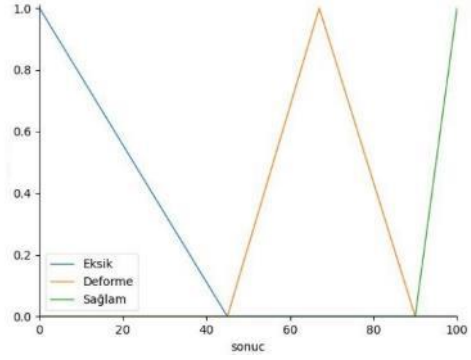


Şekil 2. 43. Bulanık mantık yapısı

Giriş görüntüsüne YOLOv4 algoritması uygulandıktan sonra elde edilen güven değerleri kullanarak bağlantı elemanının durumu hakkında çıkış üretilcektir. Bu amaçla, giriş ve çıkış için isimler, sayılar ve değerler belirlenmiştir. Giriş için Şekil 2.44.a’da gösterildiği gibi 6 adet giriş değeri için üyelik fonksiyonları belirlenmiştir. Şekil 2.44.b’de görüldüğü gibi %0 ile %45 aralığında sonuç alınırsa “eksik”, %45 ile %90 aralığında sonuç alınırsa “deforme”, %90 üzerinde sonuç alınırsa “sağlam” olarak nitelendirilecektir. Oluşturulan kurallar ise Tablo 2.4’te verilmiştir.



a)



b)

**Şekil 2. 44.** Giriş ve çıkış için oluşturulan üyelik fonksiyonları a) Girişler için üyelik fonksiyonları b) Çıkış için üyelik fonksiyonları

**Tablo 2.4.** Kurallar

| Girişler |          |          |          |          |          | Çıkış |
|----------|----------|----------|----------|----------|----------|-------|
| 1. Parça | 2. Parça | 3. Parça | 4. Parça | 5. Parça | 6. Parça | Sonuç |
| YH       | YH       | YH       | YH       | YH       | YH       | E     |
| OH       | OH       | OH       | OH       | OH       | OH       | D     |
| DH       | DH       | DH       | DH       | DH       | DH       | D     |
| S        | S        | S        | S        | S        | S        | S     |
| T        | T        | T        | T        | T        | T        | S     |

Tabloda YH; yüksek hasar, OH; orta hasar, DH; düşük hasar, S; sağlam, T; tam, E; eksik ve D; deforme olarak kodlanmıştır.

## 2.5. DCGAN ve Siyam Sinir Ağı Kullanarak Bağlantı Elemanı Kusur Tespiti

Tezin bu bölümünde, bağlantı elemanlarındaki kusurların tespiti için derin öğrenme tabanlı yöntem önerilmiştir. Ancak derin öğrenme benzeri yöntemlerde modeli eğitmek için çok fazla veri setine ihtiyaç duyulur. Evrişimli sinir ağları, küçük veri kümesi ile özellikleri öğrenememektedir. Bu nedenle fazla sayıda bağlantı elemanı görüntüsü içeren veri kümesi oluşturulmalıdır. Ancak, sağlam bağlantı elemanı görüntülerini toplamak kolayken deforme bağlantı elemanı görüntülerini bulmak zor bir işlemdir. Deforme bağlantı elemanlarından oluşan veri kümesini oluşturmak için yüzlerce kilometre demiryolunu gezinmek gerekebilir. Bu nedenle önerilen yöntemde, DCGAN kullanarak demiryolu bağlantı elemanı görüntüleri, yapay olarak üretilmiştir. Buradaki amaç, veri kümesindeki deforme bağlantı elemanı görüntülerinin fazla olmasını sağlamaktır. Çoğaltılan veri kümesi kullanılarak siyam sinir ağı ile kusur durumu incelenmiştir. Sağlam ve deforme olmak üzere iki sınıf bulunmaktadır. Her sınıf için benzerlik puanları hesaplanmıştır. Amaç, görüntüler arasındaki benzerlikleri kullanarak karşılaştırma yapmaktır.

Veri kümesinde, 800 bağlantı elemanı görüntüsü bulunmaktadır. Görüntülerin 400'ü sağlam 400'ü deforme bağlantı elemanı görüntülerinden oluşmaktadır. Eğitim kümesindeki deforme bağlantı elemanı görüntülerine DCGAN ile üretilen deforme bağlantı elemanı görüntüleri dâhildir. Test veri kümesinde ise yapay görüntü bulunmamaktadır. Test için 100 sağlam 100 deforme bağlantı elemanı görüntüsü bulunmaktadır. Veri kümesi 1000 görüntüden oluşmaktadır ve her görüntünün çözünürlüğü,  $227 \times 227$  pikseldir.

Demiryolundan toplanan görüntüler, parlak olmadığından görüntü önileme ile parlaklığın artırılması gerekmektedir. Görüntülere uygulanan algoritma, [76]'ya dayanmaktadır. Algoritmada giriş olarak düşük düşük seviye görüntü alınır. Giriş görüntüsünün tersi alınır ve ardından ters çevrilen görüntüye optimize görüntü bulanıklığı giderme algoritması uygulanır. Böylece daha net görüntüler elde edilmiş olur. İlk aşamada, giriş görüntüsü  $I$  için Denklem 2.31 kullanarak görüntü ters çevrilir.

$$R^c(x) = 255 - I^c(x) \quad (2.31)$$

Denklem 2.31'de  $c$  renk kanalı,  $I^c(x)$ , düşük aydınlatmalı görüntüyü temsil eden  $I$ 'nin piksel  $x$ 'in renk kanalının yoğunluğu ve  $R^c(x)$ , ters çevrilmiş görüntü olan  $R$  ile aynı yoğunluğu ifade etmektedir. Elde edilen görüntüye Denklem 2.32'de verilen denklem kullanılarak bulanıklık kaldırma uygulanmaktadır [77].

$$R(x) = J(x)t(x) + A(1 - t(x)) \quad (2.32)$$

Burada  $A$ , küresel atmosferik ışıktır.  $R(x)$ , kameranın yakaladığı piksel  $x$ 'in yoğunluğudur.  $J(x)$ , orijinal nesnelerin veya sahnenin yoğunluğudur.  $t(x)$ , nesnelerden veya sahneden yayılan

ışığın yüzde kaçının kameraya ulaştığıdır. Atmosfer homojen olduğunda,  $t(x)$  şu şekilde ifade edilebilir:

$$t(x) = e^{-\beta d(x)} \quad (2.33)$$

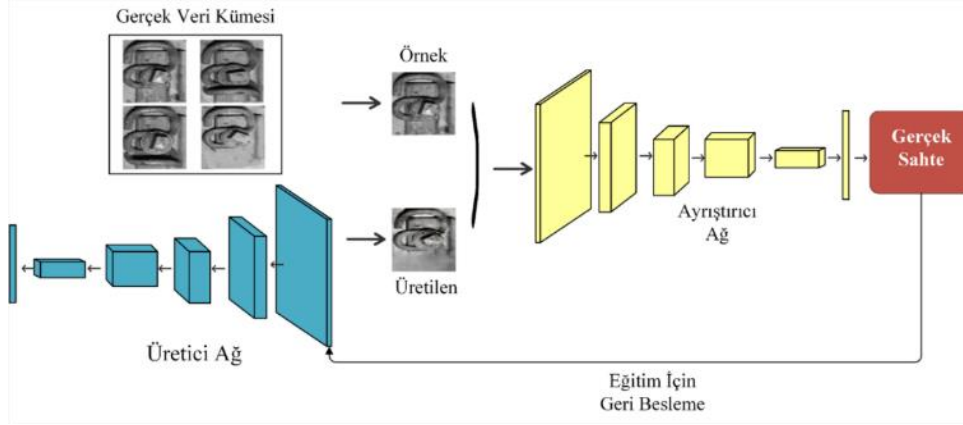
Burada  $\beta$  atmosferin saçılma katsayısı ve  $d(x)$  piksel  $x$ 'in sahne derinliğidir.  $\beta$ 'nın sabit olduğu aynı görüntüde,  $t(x)$ , nesne ile kamera arasındaki mesafe olan  $d(x)$  tarafından belirlenir. [76]'ya göre  $J(x)$  Denklem 4'teki gibi hesaplanmıştır:

$$J(x) = \frac{R(x) - A}{P(x)t(x)} + A \quad (2.34)$$

Denklem 2.34'e göre eğer  $t(x)$  0,5'ten küçük ise, pikselin güçlendirilmesi gerektiği anlaşılır. Bu pikselin RGB yoğunluğu  $P(x)t(x)$ 'i daha da küçültmek için  $P(x)$ 'e daha küçük bir değer atanarak artırılır. Ters durumda piksel yoğunluğu aşırı derecede artırılmaz.

### 2.5.1. DCGAN ile eğitim görüntülerinin çoğaltılması

Aktif kullanılan demiryolu sahasında deforme olmuş bağlantı elemanı bulmak oldukça zor bir görevdir. Kusur tespitinde kullanılacak ağı eğitmek için yüzlerce deforme bağlantı elemanı görüntüsünü içeren veri kümesi oluşturulmalıdır. Bu nedenle tezin bu bölümünde deforme bağlantı elemanı görüntülerinin çoğaltılması için derin öğrenme tabanlı yöntem önerilmiştir. Derin öğrenme tabanlı yöntemler, görüntülerin yapay olarak üretilmesi için kullanılmaktadır [78]. Görüntü üretiminde derin öğrenme tabanlı yöntemin ilk kullanımı Goodfellow tarafından yapılmıştır [79]. Görüntü üretmek için çekişmeli üretici ağlar (GAN) kullanılır. GAN'lar bir adet üretici (generative) ve bir adet ayırıştırıcı (discriminator) olmak üzere iki farklı derin ağı sahiptir. Bu tezde, görüntü üretmek için çekişmeli üretici ağların alt dallarından biri olan Derin Evrişimli Çekişmeli Üretici Ağlar (DCGAN) kullanılmıştır. Derin Evrişimli Çekişmeli Üretici Ağlar, bir CNN ile bir GAN'ı birleştirir [80]. DCGAN mimarisi Şekil 2.45'te verilmiştir. DCGAN yapısında üretici ağ ve ayırıştırıcı ağ, çekişmeli olarak aynı anda eğitilir.

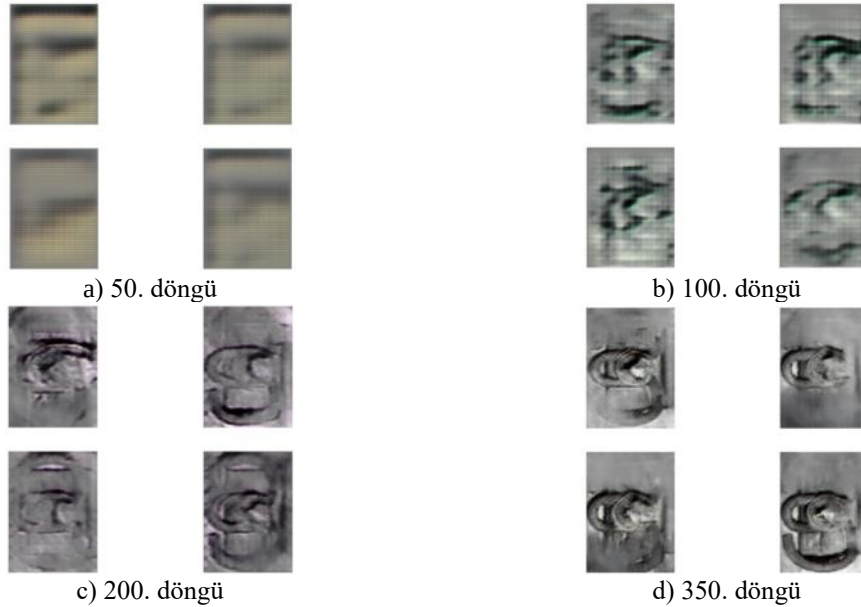


Şekil 2. 45. DCGAN Yapısı

Ayrıştırıcı ağ, bir görüntünün gerçek olup olmadığını belirlemeye yöneliktir. Ayrıştırıcı ağ, üretici modelin ürettiği verileri gerçekçi ve sahte olarak sınıflandıran ikili bir sınıflandırıcıdır. DCGAN modelinin çekişmeli eğitim süreci Denklem 2.35 kullanılarak hesaplanır.

$$\minmax(D, G) = E_{x \sim p_{data}(x)}[\log D(x)] + E_{z \sim p(z)}[\log(1 - D(G(z)))] \quad (2. 35)$$

Üretici ağ, gerçekçi görüntüler üretmeyi öğrenirken, ayrıştırıcı ağ ise gerçek görüntüler ile sahte görüntüleri ayırt etmeyi öğrenir. Eğitim sırasında sürekli geri besleme yapılarak üretici ağın daha gerçekçi görüntüler üretmesi sağlanır. Ayrıştırıcı ağ ise üretilen görüntüler ile sahte görüntüleri ayırt etmede daha iyi hale gelir. Bu süreç dengeye ulaştığında ayrıştırıcı ağ tarafından artık gerçek görüntüler sahtelerden ayırt edilemez ve en gerçekçi yapay görüntüler elde edilmiş olur. Üretici ağın ürettiği sonuçlar devir sayısına göre Şekil 2.46’da verilmiştir.



Şekil 2. 46. Eğitim aşamasında devir sayısına göre DCGAN ile üretilen bazı deforme bağlantı elemanı görüntüleri

DCGAN tarafından üretilen kusurlu bağlantı elemanı görüntüleri eğitim için kullanılmıştır. Böylece, doğruluğun artırılması amaçlanmıştır. Hâlihazırda 100 adet kusurlu bağlantı elemanı görüntüsü bulunan veri kümesine 300 adet yapay görüntü eklenerek toplam kusurlu bağlantı elemanı sayısı 400'e tamamlanmıştır. DCGAN ile görüntü üretmek için kullanılan veri kümesinden bazı görüntüler Şekil 2.47.a'da verilmiştir. DCGAN ile oluşturulan bazı yapay görüntü örnekleri ise Şekil 2.47.b'de verilmiştir.



**Şekil 2. 47.** Oluşturulan deforme veri kümesi a) Görüntü üretimi için kullanılan veri kümesinden bazı deforme bağlantı elemanı görüntüleri b) Eğitim için kullanılmak üzere DCGAN ile üretilen görüntüler

### 2.5.2. Siyam Ağı Modeli

Siyam ağları, Bromley vd. tarafından imza doğrulaması problemini gerçekleştirmek için önerilmiştir [81]. Az miktardaki veriden öğrenme yeteneği, Siyam ağlarını son yıllarda daha popüler hale getirmiştir. Siyam ağları, daha iyi tahminler elde etmek için yalnızca birkaç görüntü kullanır ve diğer ağlara göre daha iyi sonuçlar üretir. Yüz tanıma ve imza doğrulama gibi alanlarda kullanılmaktadır. Siyam ağları, özellik vektörlerini karşılaştırarak girişlerin benzerliğini bulmak için kullanılır. Siyam ağlarında, giriş görüntülerinin karşılaştırılabilmesi için uygun benzerlik fonksiyonu öğrenilir. Model, iki girdi arasında ayrım yapmayı öğrenir. Siyam ağının sahip olduğu girişlerden farklı görüntüler kullanıldığında hesaplanan fark fazla olacaktır. Aksi durumda ise fark az olacaktır. Böylece, iki görüntünün farklı olup olmadığı belirlenecektir. Siyam sinir ağının mimarisi Şekil 2.48'de verilmiştir.

Şekil 2.48'de gösterildiği gibi ağlardan birindeki ağırlıklar güncellenirse diğer alt ağdaki ağırlıklar da güncellenmektedir. İlk olarak, siyam ağının iki alt ağına, sağlam ve deforme bağlantı elemanı görüntüleri verilmektedir. İki görüntü ağdan geçtikten sonra özellik vektörleri elde edilir. Elde edilen özellik vektörleri karşılaştırılarak iki görüntünün ne kadar benzer olduğu belirlenir. Karşılaştırma işlemi için Öklid mesafesi ile vektörler arasındaki mesafe ölçülür. Görüntüler, elde

edilen mesafe değeri küçükse benzer, büyükse benzer değildir. Öklid formülü Denklem 6'da verilmiştir. Denklemde, x ve y vektörleri ifade etmektedir.

$$D = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2. 36)$$

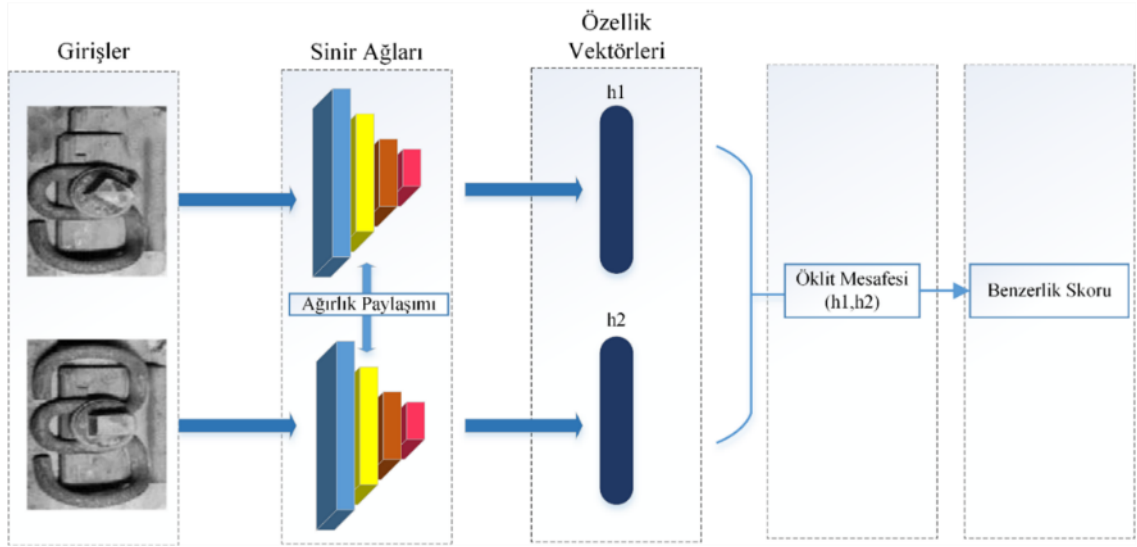
Eğitimde kullanılabilecek farklı hata fonksiyonları kullanılabilir. Hata fonksiyonu kullanılmasının amacı, görüntü çiftlerinin ne kadar iyi ayırt edildiğini belirlemektir. Genellikle siyam ağları için hata fonksiyonu olarak karşılaştırmalı hata fonksiyonu ve ikili çapraz entropi hata fonksiyonları kullanılır. Bu çalışmada, tercih edilen hata fonksiyonu ikili çapraz entropidir. Burada, ikili sınıflandırma gerçekleştirmek için girdilerin aynı sınıfta olup olmadığını sınıflandırmak amaçlanmıştır. İkili çapraz entropi'de kayıp hesabı şu şekilde yapılır.

$$L = -y \log p + (1 - y) \log(1 - p) \quad (2. 37)$$

Denklemde L kayıp fonksiyonunu ifade ederken y sınıf etiketini (0 veya 1) ve p ise tahmini ifade etmektedir. Siyam ağını eğitmek için, her seferinde bir benzer ve bir benzer olmayan örnek beslenebilir ve kayıplar hesaplanabilir. Bu kayıp fonksiyonu, pozitif ve negatif kategorilerden bir eğitim örneği sağlanıp her iki kayıp Denklem 2.38'deki gibi toplanırsa, pozitif ve negatif görüntüler arasında ayırım yapabilmek için ağı eğitmek için kullanılır.

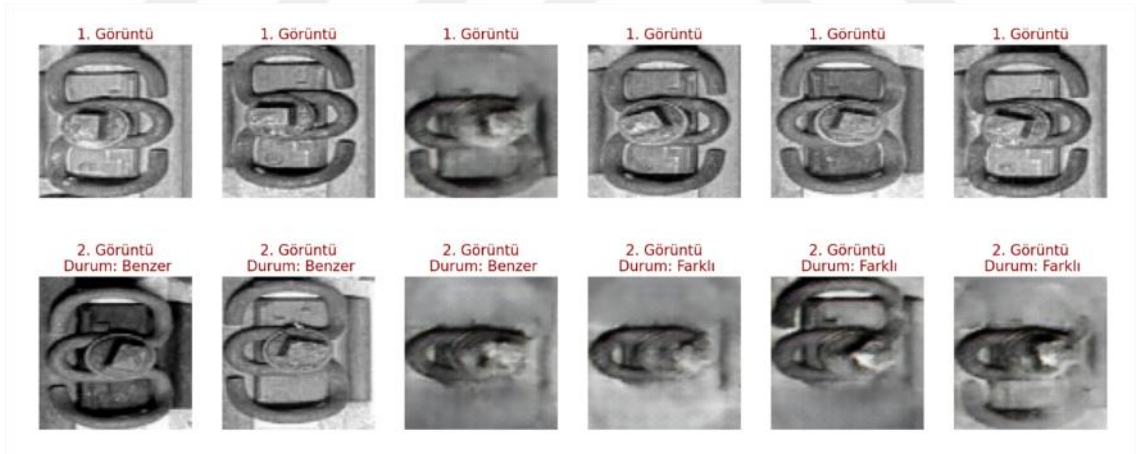
$$L = L_{pozitif} + L_{negatif} \quad (2. 38)$$

Şekil 5'te görüldüğü gibi siyam ağında iki aynı alt ağ olduğundan, siyam ağına iki giriş olmalıdır. Siyam ağı eğitiminde kullanılacak görüntü çiftlerine ihtiyaç vardır. Bu nedenle aynı sınıfa ait ve farklı sınıfa ait görüntü çiftleri oluşturulmuştur. Görüntü çiftleri eğitim veri kümesinden rastgele seçilmiştir ve siyam ağının benzerlikleri öğrenebilmesi için kullanılacaktır.



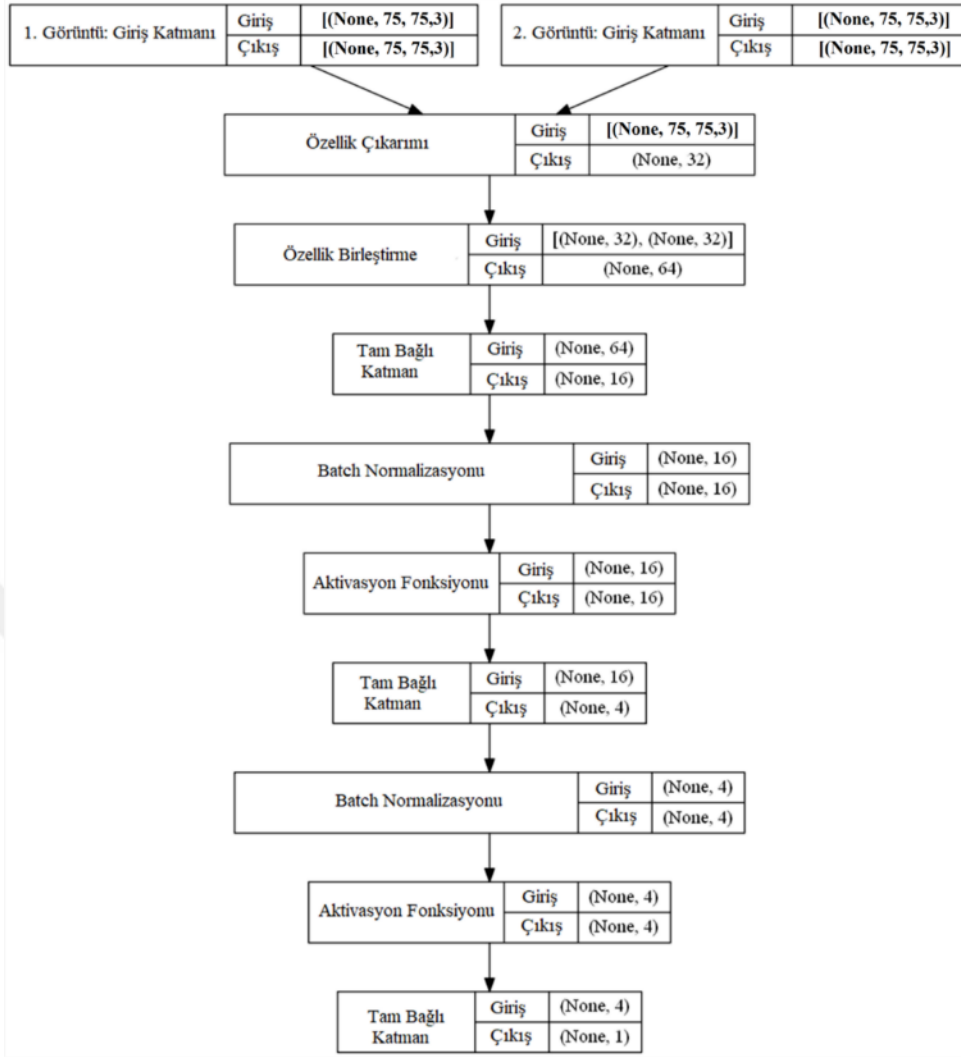
Şekil 2. 48. Siyam sinir ağı yapısı

Siyam ağıının eğitim aşamasında, aynı ve farklı görüntü çift örnekleri eğitim veri kümesinden rastgele seçilerek oluşturulmuştur. Şekil 2.49’da rastgele seçilerek oluşturulan bazı görüntü çiftleri verilmiştir. Seçilen görüntüler, seçilen görüntüler aynı sınıftaysa benzer, farklı sınıftaysa farklı olarak nitelendirilmiştir.



Şekil 2. 49. Rastgele seçilen bazı görüntü çiftleri

Alt ağlara verilecek giriş görüntüleri için uzamsal boyular genişlik, yükseklik ve sayı kanalı şeklindedir. Kullanacağımız veri kümesi için giriş görüntüleri 75x75x3 biçimindedir. Çalışmada, aktivasyon fonksiyonu olarak Relu [82] kullanılmıştır. Siyam ağıında kullanılan ağıın yapısı Şekil 2.50’de verilmiştir.



Şekil 2. 50. Siyam sinir ağı modelinde kullanılan katmanlar ve parametreler

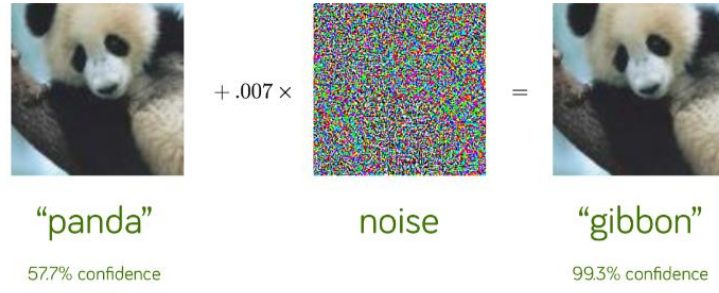
## 2.6. Gürbüz Derin Öğrenme Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti

Son yıllarda yapay zekâ uygulamaları, birçok alanda yaygın olarak kullanılmaktadır. Bununla birlikte bu uygulamalar genellikle çeşitli gürültülere karşı savunmasızdır ve en doğru derin öğrenme modelleri bile çeşitli bozulmalarla kolayca aldatılabilir. Sinir ağları da dâhil olmak üzere makine öğrenimi algoritmalarının çoğu, rakip örnekleri sürekli olarak yanlış sınıflandırır. Bu nedenle, yapay zekâ sistemlerinin gürültülere karşı sağlamlığının artırılması, yapay zekânın daha da geliştirilmesinde giderek daha önemli bir rol oynamıştır.

Tezin bu bölümünde, derin ağların çekışmeli eğitimi araştırılmıştır. Çalışmanın odak noktası, gürültülü bağlantı elemanı görüntülerinin nasıl doğru sınıflandırılacağıdır. Düşmanca eğitim, sınıflandırıcıları düşman saldırılarına karşı sağlamlaştırmak için kullanılan bir yöntemdir. Burada, eğitim örneklerinden rakip örnekler oluşturulur ve ardından model, orijinal örnekler yani temiz örnekler ile karıştırılarak yeniden eğitilir. Bu yaklaşım, modelin sağlamlığını artırmak için en çok kullanılan yöntemdir.

Derin sinir ağları, görüntü sınıflandırma, nesne algılama ve doğal dil işleme gibi birçok uygulama için kullanılmaktadır. Makine öğrenimi sistemlerinde çoğu zaman modelin çalışıp çalışmadığına dikkat edilse de modelin aynı zamanda sağlam ve güvenilir olup olmadığının incelenmesi kritik bir görevdir. Modellerin rakip saldırılara karşı savunmasız olması, derin öğrenmeye dayalı yöntemler başta olmak üzere modern makine öğrenimi sistemlerinin karşı karşıya olduğu büyük bir sorundur.

Modelleri yanıltıcı verilerle kandırmaya çalışan bir teknik olan rakip saldırılar, yapay zekâ ve makine öğrenimi alanında büyüyen bir tehdittir. Şekil 2.51’de görüldüğü gibi, girdi verisine gürültü eklenmesi derin öğrenme modelinde hataya sebep olabilmektedir [83]. Rakip örnek saldırısı eğitim sırasında modelin yanlış verilerle eğitilmesini veya eğitilmiş modeli aldatmak için yanlış verilerin sunulmasını gerektirebilir. Hâlihazırda üretimde kullanılan modellerin çoğu saldırıya açıktır. Daha önceki yapılan araştırmalarda bir otonom arabanın geçtiği yola birkaç küçük çıkartma yerleştirilerek arabanın şerit değiştirmesine neden olabileceği gösterilmiştir.



Şekil 2. 51. Rakip örneklerin kullanılması

Yapay zekâ uygulamalarında veri kümesi boyutunun önemi büyüktür. Veri kümelerinin boyutu artırılarak derin sinir ağları için görüntü sınıflandırma ve nesne algılama gibi görevlerde daha yüksek doğruluk oranları elde edilebilir. Bununla birlikte, veri setinin gürültülü veriler içermesi modelin performansına ciddi derecede zarar verecektir. Girdideki genellikle insanlar tarafından görülemeyecek kadar küçük bozulmalar, tahminde çok sayıda hata yaratabilmektedir. Bu tür girdilere, karşıt örnekler denmektedir. Derin öğrenme modellerinin bu tür bozulmalara karşı dayanıksız olduğu keşfedildiğinden beri derin öğrenme modellerinin sağlamlığının nasıl geliştirileceği konusunda araştırmalar yapılmaktadır. Sağlamlık üzerine birçok çalışma yapılsa da bu alanda son yıllarda çekişmeli sağlamlık üzerine büyük bir ilgi vardır. Çekişmeli öğrenmede amaç, giriş görüntülerinin bozulmalarına karşı dayanıklı sınıflandırıcılar geliştirilmesidir.

Düşmanca derin sinir ağları eğitmek için bazı savunma yöntemleri vardır. Bunların arasında iyi sonuçlar çekişmeli eğitim ile elde edilmektedir. Düşmanca eğitim, sinir ağlarını orijinal veriler yerine rakip örnekler üzerinde eğiten bir tür veri büyütme yöntemidir. Rakip görüntü, derin sinir ağları gibi bir makine öğrenimi tekniğini kandırmak için kasıtlı olarak gürültü eklenerek bozulan temiz bir görüntünün değiştirilmiş bir versiyonudur. Rakip örnekler, modelin hata yapmasına neden olmak için bilerek tasarlanan makine öğrenimi modellerine girdilerdir. Rakip bir örnek, orijinal bir verinin bozuk bir versiyonudur. Bozulma, görüntüye küçük bir gürültü eklenerek yapılır. Rakip veriler, insanlara "normal" görünür ancak hedeflenen makine öğrenimi modeli tarafından yanlış sınıflandırmaya neden olmaktadır.

### 2.6.1. Popüler Düşman Saldırı Yöntemleri

#### Sınırlı bellekli BFGS (L-BFGS)

Sınırlı bellekli Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) yöntemi, görüntülere eklenen bozulmaların sayısını en aza indirmek için doğrusal olmayan gradyan tabanlı bir sayısal optimizasyon algoritmasıdır. Rakip örnekler oluşturmada etkili bir yöntemdir. Hesaplama işlemlerinin zaman alıcı olması nedeniyle pratik değildir.

### **Hızlı Gradyan İniş Yöntemi (FGSM)**

FGSM yöntemi, görüntünün herhangi bir pikseline eklenen maksimum bozulma miktarını en aza indirmek ve çekişmeli örnekler oluşturmak için basit ve hızlı gradyan tabanlı bir yöntem kullanır. FGSM yöntemi verileri, yanlış sınıflandırmaya neden olacak şekilde bozar. Hesaplama işlemleri diğer birçok yöntemle göre daha hızlıdır.

### **Jacobian Tabanlı Belirginlik Haritası Yöntemi (JSMA)**

FGSM'den farklı olarak bu yöntemde, değiştirilen özelliklerin sayısı en aza indirilmek için özellik seçimi kullanılır. Düz pertürbasyonlar, azalan düzende belirginlik değerine göre yinelemeli olarak özelliklere eklenmektedir. Gürültü eklenirken çok az özellik bozulmaktadır. Hesaplama yoğunluğu, FGSM'den fazladır.

### **Carlini ve Wagner Yöntemi (C&W)**

C&W yöntemi, L-BFGS saldırısına dayanmaktadır. Bu yöntemde, kutu kısıtlamaları ve farklı amaç fonksiyonları bulunmamaktadır. Böylece, muhalif örnekler oluşturmada daha verimlidir. Rakip veriler üretmede etkili bir algoritmadır. Hesaplama yoğunluğu, FGSM, JSMA ve Deepfool'dan fazladır.

### **Üretken Düşman Ağları (GAN)**

GAN'lar, iki sinir ağının birbiriyle rekabet ettiği düşmanca saldırılar oluşturmak için kullanılmıştır. Bir üretici ve bir ayırıştırıcı ağı sahiptir. Eğitimde kullanılan örneklerden farklı örneklerin üretilmesi için kullanılır. Hesaplama açısından çok yoğundur. Üretilen veriler bazen çok istikrarsız olabilir.

### **Sıfırıncı Dereceden Optimizasyon Yöntemi (ZOO)**

ZOO tekniği, sınıflandırıcıların gradyanının tahminini sınıflandırıcıya erişim olmadan sağlar. Yöntem, değiştirilmiş bireysel özelliklerle hedef modeli sorgulayarak gradyan ve kendi tahmin eder ve bozulmaları optimize etmek için Adam veya Newton'un yöntemini kullanır. Performansı C&W yöntemine benzemektedir. Hedef sınıflandırıcı için sorgu sayısı fazladır.

Bu çalışmada rakip görüntüler oluşturmak için Hızlı Gradyan İniş Yöntemi (FGSM) kullanılmıştır. Rakip görüntü oluşturmak için FGSM yönteminin kullanılma sebebi, bu yöntemin rakip görüntüler oluşturmak için basit ama etkili bir yöntem olmasıdır. FGSM şu şekilde çalışmaktadır:

1. Giriş görüntüsünü al
2. Eğitimli bir CNN kullanarak görüntü üzerinde tahminde bulun
3. Gerçek sınıf etiketine dayalı tahmin kaybı hesapla
4. Giriş görüntüsüne göre kaybın gradyanlarını hesapla

5. Gradyanın işaretini hesapla
6. Düşman görüntü oluşturmak için işaretli gradyanı kullan

FGSM, girdi görüntüsüne göre bir kayıp fonksiyonunun gradyanlarını hesaplayıp gradyanların işaretini kullanarak yeni bir görüntü oluşturur. Oluşturulan görüntü rakip görüntüdür. Elde edilen görüntü sinir ağının yanlış bir tahmin yapmasına neden olmaktadır. Hızlı Gradyan İşaret Yöntemi aşağıdaki denklem ile ifade edilebilir:

$$Rakip_x = x + \epsilon * sign(\nabla_x J(\theta, x, y)) \quad (2. 39)$$

Burada,  $Rakip_x$  üretilen rakip görüntü,  $x$  orijinal giriş görüntüsü,  $\epsilon$  bozulmayı sağlayan değer,  $\theta$  sinir ağı modeli ve  $J$  kayıp fonksiyonudur.



### 3. BULGULAR VE TARTIŞMA

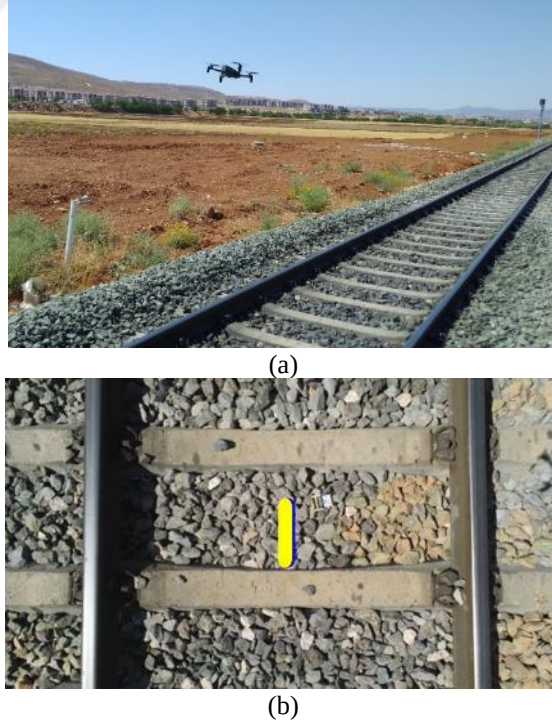
#### 3.1. Otonom Ray Takip Algoritmaları İçin Deneysel Sonuçlar

##### 3.1.1. HSV'ye Dayalı Ray Takip Algoritmasının Deneysel Sonuçları

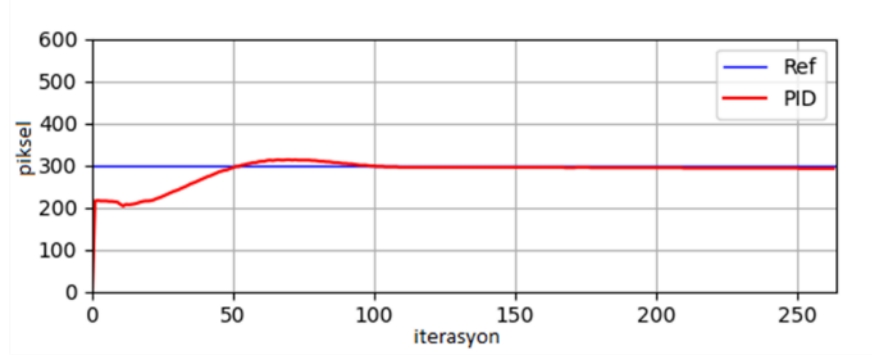
HSV'ye dayalı ray takip algoritması gerçek demiryolu sahasında test edilmeden son kontrolleri videolar üzerinde yapılmıştır. Algoritma, gerçek demiryolu ortamında denenmeden önce demiryolu ortamından alınan videolar üzerinde en iyi eşik değerleri belirlenmiştir. Bu videolar, İHA ile farklı koşullarda manuel kontrolle toplanmıştır. Buradaki amaç, algoritmanın farklı koşullarda performansını doğrulamaktır. Yapılan deneyde, farklı ray koşullarında uygulamanın iyi çalıştığı görülmüştür.

Son aşamada algoritma, aktif olarak kullanılan demiryolu üzerinde test edilmiştir. Şekil 3.1'de demiryolu sahasında yapılan test anlarına ait görüntüler görülmektedir. Takip için PID denetleyici kullanılmıştır. Uygulamada,  $K_p$  değeri 0.09,  $K_i$  değeri 0.05 ve  $K_d$  değeri 3 olarak belirlenmiştir. Şekil 3.1.b'de İHA'nın rayların ortasını yakalama anı gösterilmiştir.

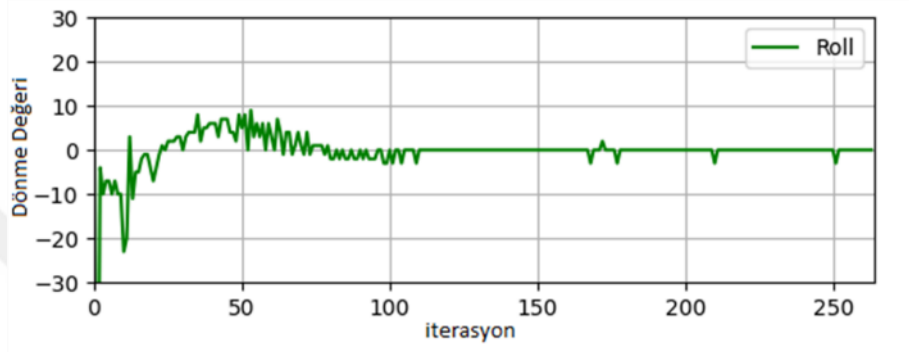
Şekil 3.2.a'da PID denetleyici kullanılarak ray takibi yapılan İHA'nın, rayların ortasını yakalamasını göstermektedir. Şekil 3.2.b'de ise PID denetleyici çıktısının İHA'ya dönüş değeri olarak gönderilmesi sırasında oluşan değerleri göstermektedir.



Şekil 3.1. Ray takibi algoritmasının sahada test edilmesi (b) İHA'nın istenen değeri yakalaması



a)



b)

Şekil 3.2. a) Demiryolu sahasında ray hattını takip eden İHA'nın orta noktayı yakalaması b) İHA'ya gönderilen dönme değerleri

### 3.1.2. Gabor Filtresi Kullanılarak Ray Takibi İçin DeneySEL Sonuçlar

Önerilen yöntemde ilk olarak alınan görüntü, önışleme adımlarından geçirilmelidir. Görüntü önışleme, kameradan gelen görüntülerle ilgilenen ve yol çizgilerini algılamak için yararlı bilgiler üreten düşük seviyeli görüntü işlemedir. Ardından yukarıda anlatılan HSV'ye dayalı ray tespiti yöntemi ile ray çizgileri tespit edilmiştir. Son olarak, hesaplama süresinden tasarruf etmek için görüntü daha küçük bir ilgi alanına (ROI) indirgenmiştir. İHA tarafından yakalanan demiryolu görüntülerinde bulunan rayların şekli sabittir ve önemli ölçüde değişmez. Yine de hesaplama maliyetlerinin azaltılması için belirli bölge anlamına gelen ROI (Region of Interest) belirleme işlemi yapılmıştır.

ROI işlemi sonrasında ise ana işlem başlatılmıştır. Bu işlemde, elde edilen kenarlara Hough dönüşümü uygulanmıştır. Hough dönüşümü kullanılarak tespit edilen çizgilerin eğimlerinin ortalaması hesaplanmıştır. Sol ray çizgisi negatif eğime sahiptir. Bu nedenle, görüntüde elde edilen ray çizgilerinden negatif eğimli tüm çizgiler sol şerit noktaları olarak kabul edilir. Sol çizgilerin eğim ortalaması hesaplanarak tek bir sol çizgi oluşturulmuştur. Aynı şekilde sağ şerit çizgisi ise pozitif eğim değerine sahiptir. Pozitif eğimli tüm çizgiler ise sağ şerit çizgileri olarak kabul edilir. Sağ çizgilerin ortalamasından ise tek bir sağ çizgi oluşturulmuştur. Görüntüde bulunan dikey çizgiler ise atlanmıştır. Bu işlemin amacı sadece iki farklı çizginin kesişiminden tek ufuk noktası

elde etmektir. Önerilen yöntemin amacı sağ ve sol çizgilerin kesişiminden ufuk noktası elde etmektir. Sonuçlar Şekil 3.3'te gösterilmiştir.



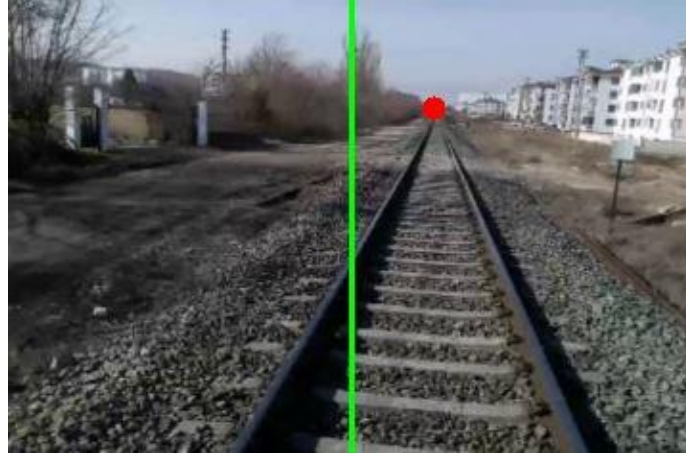
**Şekil 3.3.** Sağ ve sol çizgilerin oluşturulması

Şekil 3.3'te yeşil renk ile gösterilen iki çizginin kesişiminden oluşturulan ufuk noktası Şekil 3.4'te gösterilmiştir. Ufuk noktası kırmızı nokta ile işaretlenmiştir.



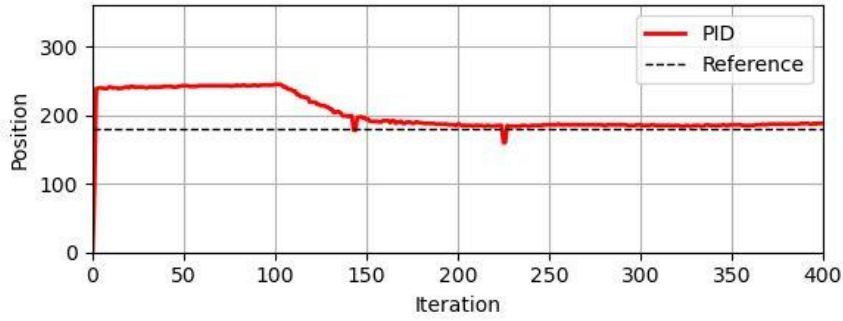
**Şekil 3.4.** Ufuk noktasının kırmızı nokta ile işaretlenmesi

PID denetleyici kullanarak takip işlemini gerçekleştirebilmek için ilk olarak, İHA'dan alınan görüntünün genişliği 360 px ve yüksekliği 240 px olarak alınmıştır. Bu değerlerin düşük çözünürlüklü seçilme sebebi görüntü işlemede hızı yükseltmektir. İHA'nın demiryolunu ortalaması için alınan görüntünün x ekseninde tam orta noktası olan 180. pikseli istenen değer olarak seçilmektedir. Böylece İHA, ufuk noktası ile orta noktayı birbirine yaklaştıracak ve yolun ortalanması işlemi gerçekleşecektir. Görüntünün orta noktası Şekil 3.5'te gösterilen yeşil çizgi, x ekseninde bulunan 180. pikselin y eksenini boyunca çizilmesinden oluşmuştur.



**Şekil 3.5.** Ufuk noktası ile merkez konumun gösterimi

PID kontrol sistemi için  $K_p$  değeri 0.4,  $K_i$  değeri 0.2 ve  $K_d$  değeri 0.4 olarak belirlenmiştir. Bu değerler kullanılarak hatanın sıfıra inme süresinin makul seviyede olduğu gözlenmiştir. Önerilen algoritma gerçek demiryolunda test edilmiştir. Sistemin ufuk noktasını yakalaması Şekil 3.6’da gösterilmiştir.

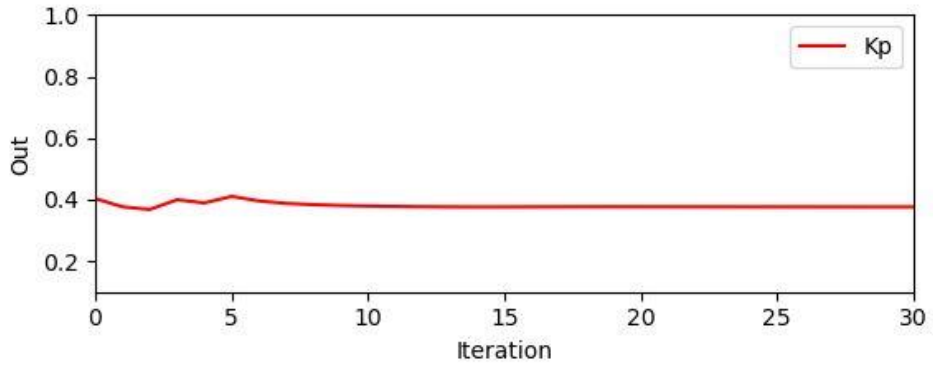


**Şekil 3.6.** Referans noktasına göre PID çıkışı

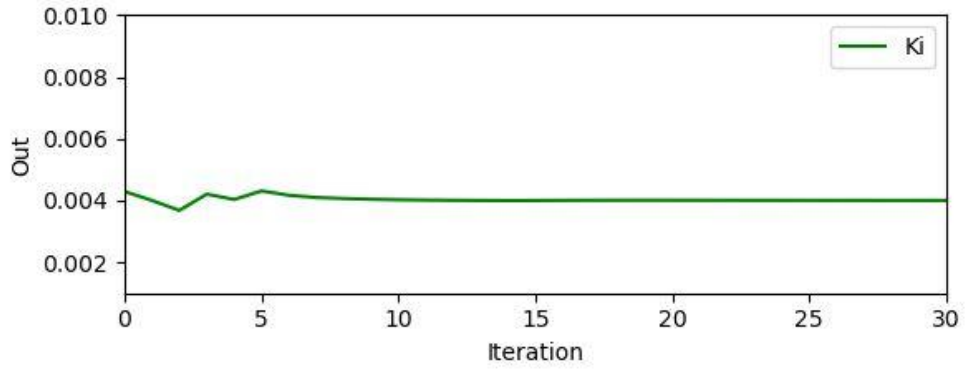
PID kontrolörünün çıkışı, İHA'nın sapma kanalına iletilmiştir. Bu sayede İHA'nın yolun ortasında hareket etmesi sağlandı. Bu süreçte İHA sabit bir hızla ileri yönlü hareket yapmaktadır. Yüksekliği ise sabittir. Testler, PID kontrolörünün dört rotorlu bir İHA'yı kontrol etmede etkili olduğunu göstermektedir.

### 3.1.3. Gabor Filtresi ve Bulanık PID Tabanlı Ray Takip Algoritmasının Deneysel Sonuçları

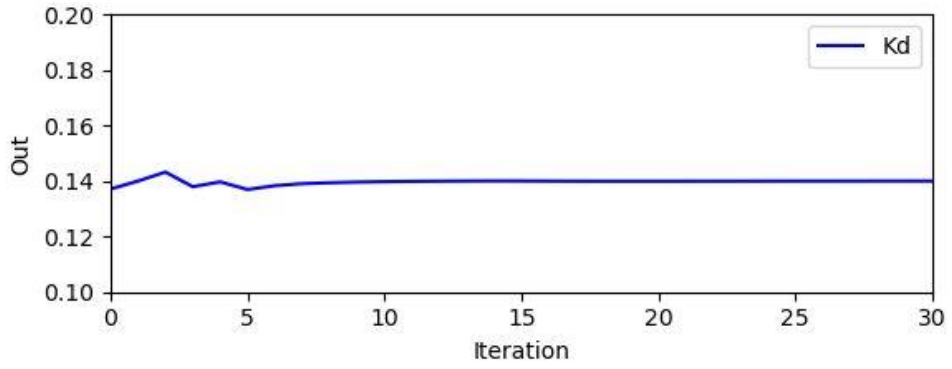
Kullanılan Bulanık-PID yapısındaki PID kazanımları, her iterasyon sonunda hataya ve hatanın türevine göre değişecektir. Bu durumun bir sonucu olarak klasik PID denetleyiciye göre daha dinamik kontrol sağlanacaktır.  $K_p$ ,  $K_i$  ve  $K_d$  değerlerinin değişimi Şekil 3.7, Şekil 3.8, Şekil 3.9’da gösterilmiştir.



Şekil 3.7.  $K_p$ 'nin değişimi.

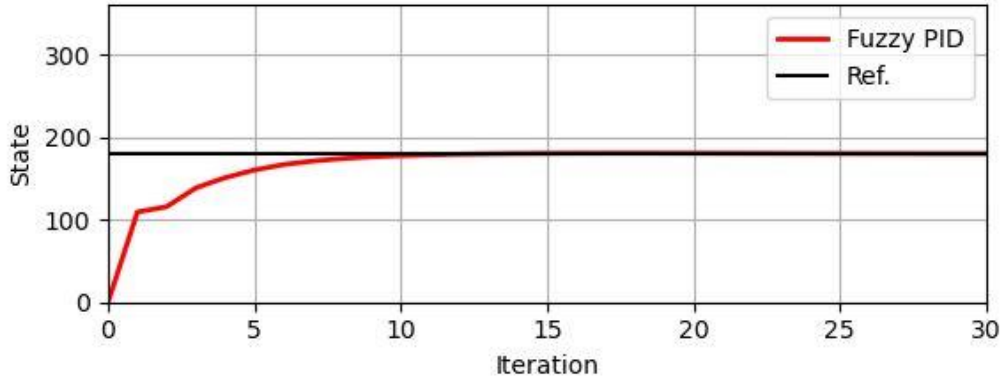


Şekil 3.8.  $K_i$ 'nin değişimi

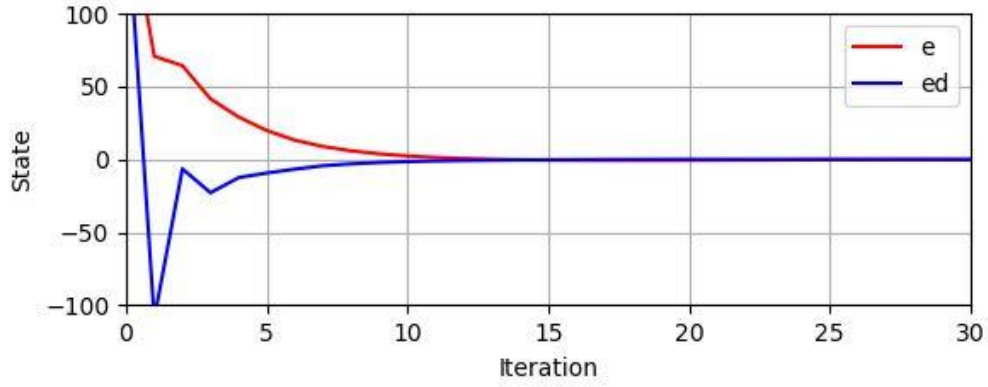


Şekil 3.9.  $K_d$ 'nin değişimi

Şekil 3.10'da, referans değerini yakalayan Bulanık PID kontrolörü gösterilmektedir. Şekil 3.11, hatanın (e) ve hatanın (ed) türevinin sıfıra yakınsadığını göstermektedir.



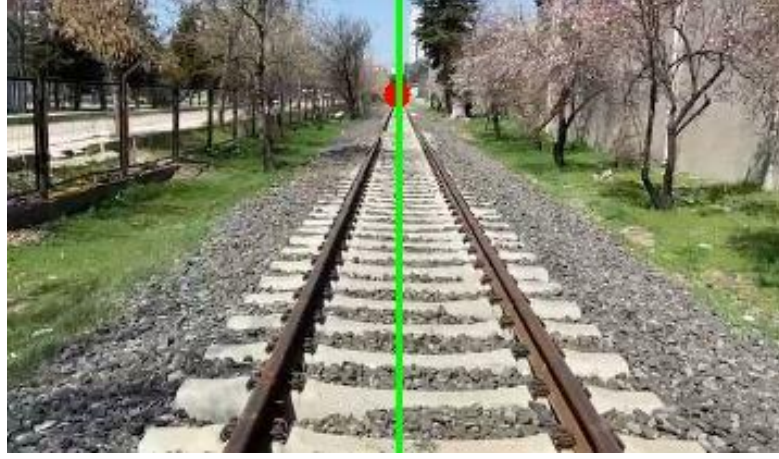
Şekil 3.10. Referans değerinin bulanık PID



Şekil 3.11. Hata(e) ve hatanın türevinin(ed) sıfıra yakınsaması

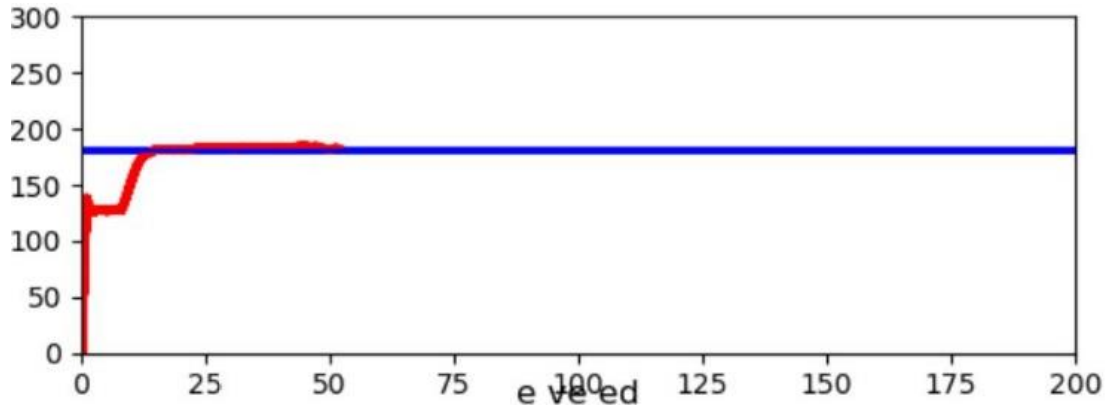
Tasarlanan demiryolu takip algoritması da sahada test edilmiştir. Algoritmayı test etmek için 720p çözünürlüklü ve 30 fps video kaydına sahip DJI Ryze Tello kullanılmıştır. İHA, demiryolu üzerinde konumlandırılmıştır. İHA'dan alınan görüntülerden ufuk noktasının bulunmasının ardından takip sürecine geçilmiştir.

İHA'dan alınan görüntü Şekil 3.12'de gösterilmektedir. Burada kırmızı işaret ufuk noktasını temsil etmektedir. Görüntünün orta noktası olan 180. pikselden dikey eksende yeşil çizgi çizildi.

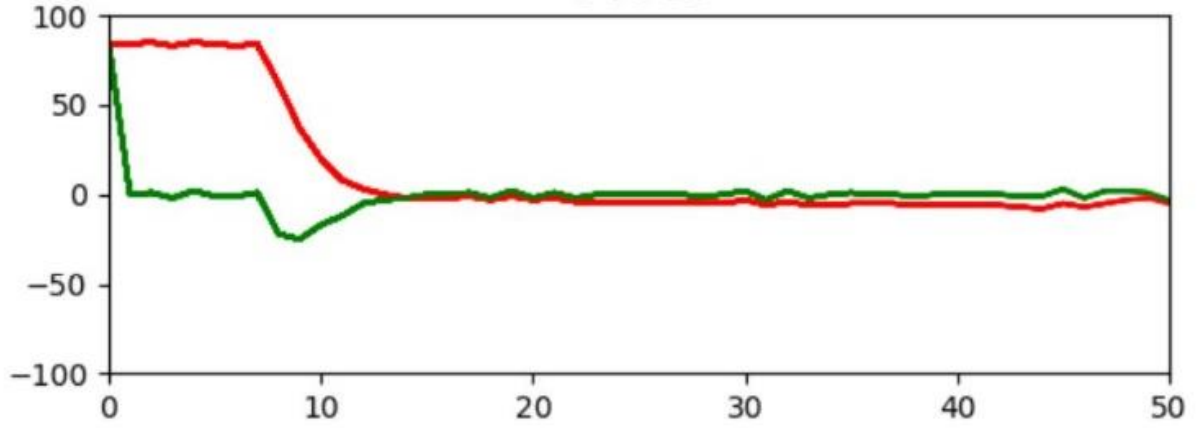


**Şekil 3.12.** Ufuk noktasının bulanık-PID ile yakalanması

İzleme komutu verildiğinde, PID kontrol parametreleri, mevcut hata durumuna ve hatanın türevine göre bulanık sistem tarafından ayarlanır. Bu işlem, çekilen her kare için devam etmektedir. Böylece klasik PID'ye göre daha dinamik bir sistem tasarlanmış ve bu sistem hedefi klasik PID'ye göre daha sorunsuz yakalamıştır. Bunun nedeni, bulanık sistem sayesinde PID kontrol parametrelerinin hedefe yaklaştıkça azalmasıdır. Bu işlem sonucunda hedefe ne kadar yakınsa sapma kanalına o kadar düşük değer atanır ve yumuşak bir dönüş sağlanır. Şekil 3.13, İHA tarafından sahada elde edilen ufuk noktasının yakalanmasını göstermektedir. Şekil 3.14, hatanın (e) ve hatanın türevinin (ed) sıfıra yakınsamasını gösterir.



**Şekil 3.13.** Ufuk noktasının İHA tarafından yakalanması



Şekil 3.14. Deney esnasında hatanın (e) ve hatanın türevinin (ed) sıfıra yakınsaması

Bulanık PID sistemini tercih etmemizin nedeni klasik PID'den daha verimli çalışmasıdır. Tasarlanan sistemin sahada test edildiği ve verimli çalıştığı, İHA'nın demiryolunu sabit bir hızda otonom olarak takip edebildiği görülmüştür.

#### 3.1.4. Light-Weight Çizgi Tespiti Yaklaşımına Dayalı Ray Takip Algoritmasının Deneysel Sonuçları

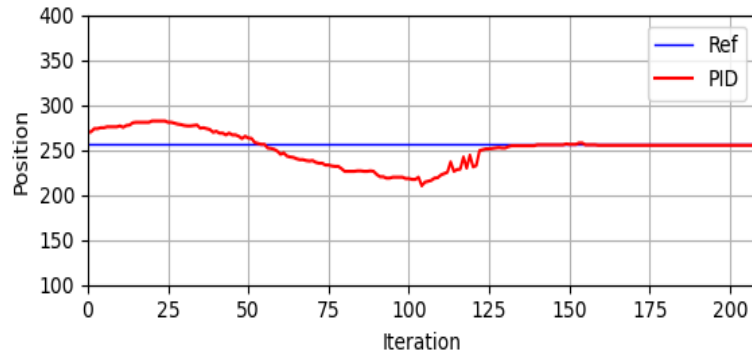
Şekil 3.15, sol ve sağ ray bileşenlerini ve travers bileşenlerini göstermektedir. Travers bilgileri sonraki aşamada travers eksikliklerinin tespiti için kullanılacaktır.



Şekil 3.15. Ray ve traverslerin tespiti

Şekil 3.15'teki sağ ve sol raylar için elde edilen çizgilerin x eksenı boyunca orta noktası bulunarak İHA'nın konumu ile iki rayın merkezi arasındaki fark en aza indirilecektir. Bu amaçla İHA'nın PID sistemi ile rayı takip etmesi sağlanacaktır. PID kontrol sistemi sonucu Şekil 3.16'da verilmiştir.

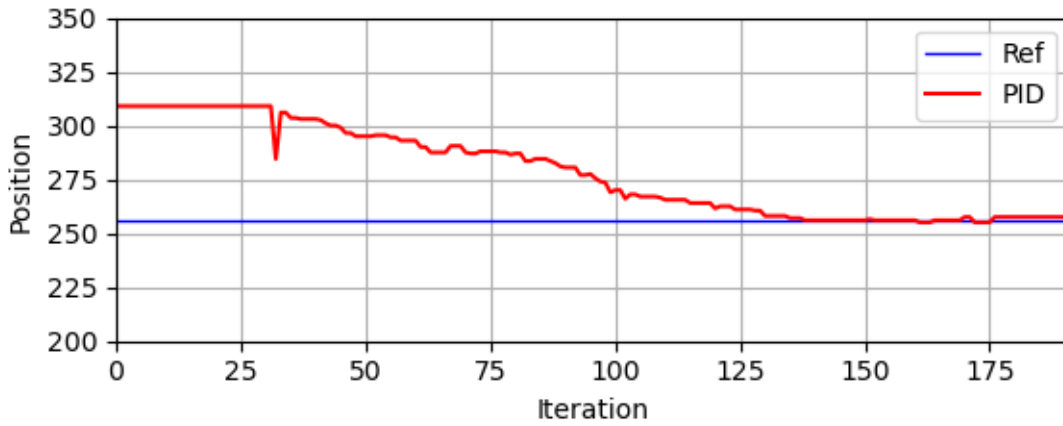
Şekil 3.16'da görüntünün merkezi ile sol ve sağ rayın orta noktası arasındaki mesafe PID kontrolü kullanılarak en aza indirilmeye çalışılmıştır.



**Şekil 3.16.** İHA'nın istenen pozisyonu yakalaması

### 3.1.5. Yolact'a Dayalı Ray Takip Algoritmasının Deneysel Sonuçları

İHA'dan alınan görüntünün genişliği 480 px ve yüksekliği 240 px olarak alınmıştır. İHA'nın demiryolunu ortalaması için alınan görüntünün x ekseninde tam orta noktası olan 240. pikseli istenen değer olarak seçilmektedir. Böylece İHA, ufuk noktası ile orta noktayı birbirine yaklaştıracak ve yolun ortalanması işlemi gerçekleşecektir. Önerilen algoritma test edilmiştir. Sistemin rayların orta noktasını yakalaması Şekil 3.17'de gösterilmiştir.



**Şekil 3.17.** PID denetleyici sonucu

Şekil 3.17'de görüldüğü gibi ray takibi, rayların orta noktasının alınan görüntünün merkezine yakınsaması ile sağlanmıştır. İki nokta bir arada tutulurken ileri yönlü hız ve irtifa sabittir. Böylece İHA, demiryolunu otonom olarak gezinmektedir.

### 3.2. Mask R-CNN Algoritması Tabanlı Travers Boşluklarının Tespiti için Deneysel Sonuçlar

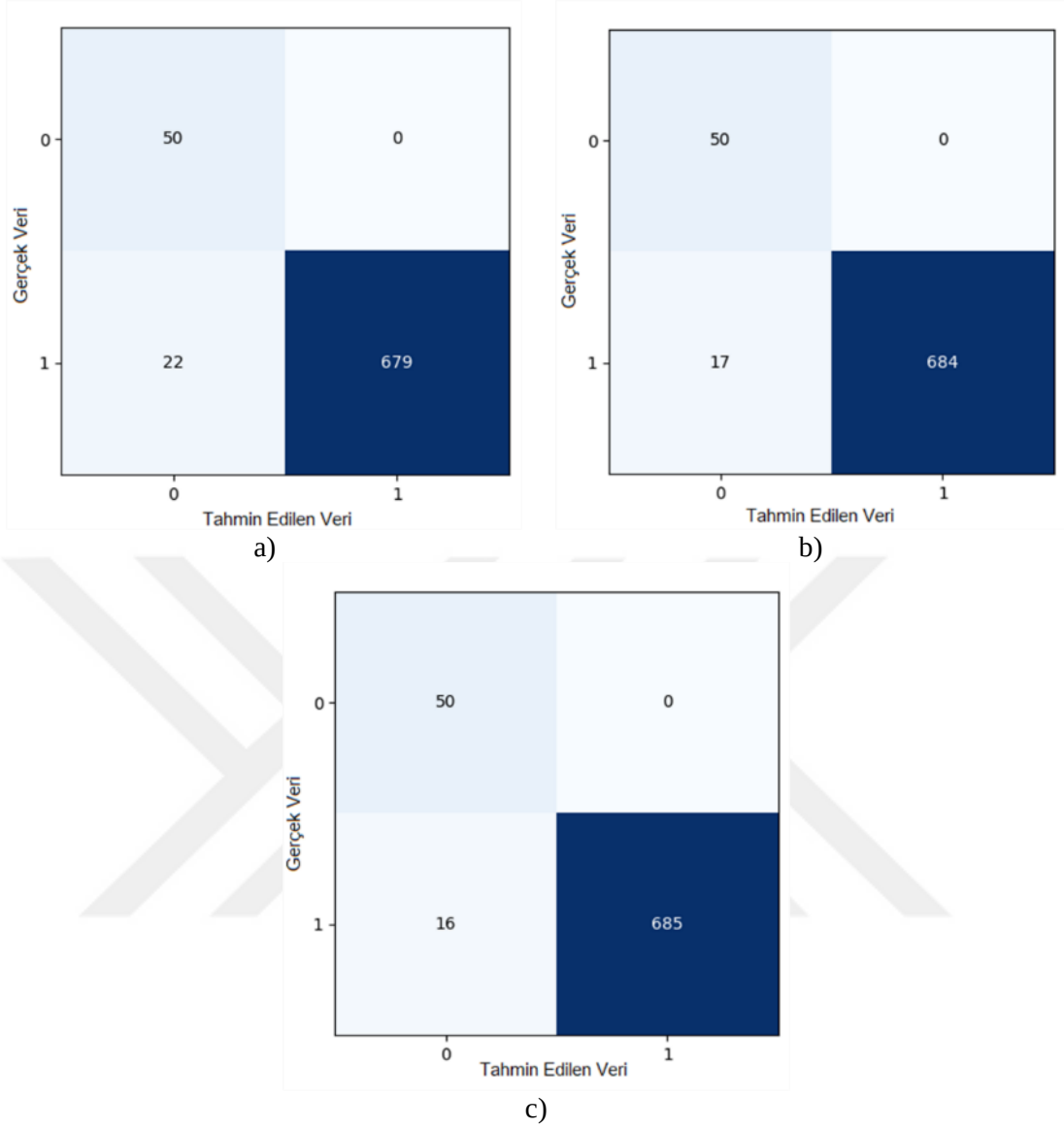
#### 3.2.1. Maske Sınırlarına Dayalı Travers Eksikliklerinin Tespiti

Önerilen algoritma toplanan görüntüler üzerinde test edilmiştir. Algoritmanın karmaşık çalışma koşullarında ve dinamik ortamlarda tanıma performansını doğrulamak için bu çalışmada daha önce İHA tarafından manuel olarak toplanan demiryolu görüntüleri kullanılmıştır.

Test verilerinin sınıflandırılmasındaki doğruluk oranları 80 cm, 75 cm ve 70 cm eşik değeri için sırasıyla %97,8, %97,7 ve %97,0 olarak bulunmuştur. Algoritma sonucunun diğer metriklerine ait sonuçlar Tablo 3.1’de verilmiştir. Elde edilen karmaşıklık matrisleri ise Şekil 3.18’de verilmiştir.

**Tablo 3.1.** Diğer ölçütler için performans sonuçları

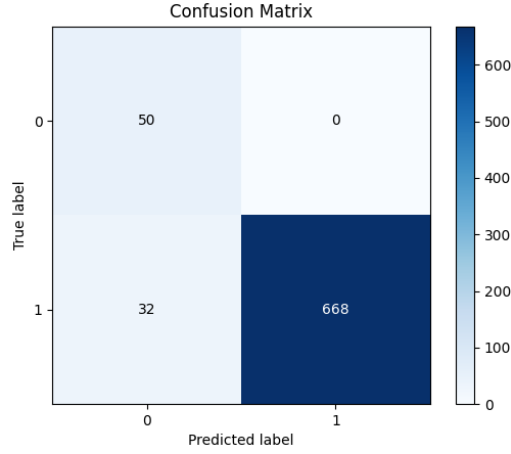
| Eşik değeri | Doğruluk | Duyarlılık | Özgünlük | F1    |
|-------------|----------|------------|----------|-------|
| 80 cm       | %97,8    | %100       | %97,7    | %98,8 |
| 75 cm       | %97,7    | %100       | %97,5    | %98,7 |
| 70 cm       | %97,0    | %100       | %96,8    | %98,4 |



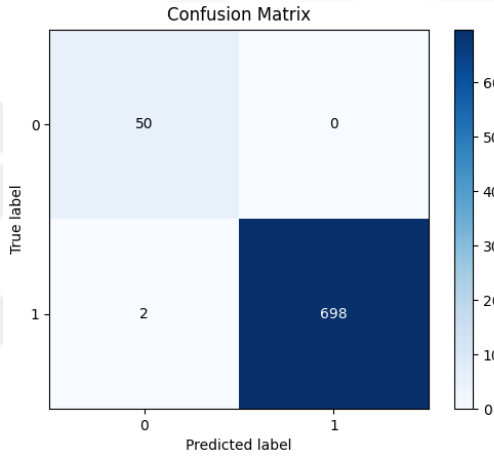
**Şekil 3.18.** Test verilerinin sınıflandırılmasından elde edilen karmaşıklık matrisi a) Eşik değeri 70 cm için sonuç b) Eşik değeri 75 cm için sonuç c) Eşik değeri 80 cm için sonuç

### 3.2.2. İkili Maskeye Dayalı Travers Eksikliklerinin Tespiti

Önerilen algoritma 700 tanesi hatasız ve 50 tanesi hatalı olmak üzere 750 görüntü üzerinde test edilmiştir. Şekil 3.19, Otsu tabanlı ikili maske ile elde edilen sınıflandırma sonucunu göstermektedir. Şekil 3.20, Mask R-CNN yöntemiyle elde edilen ikili maskeyi kullanan sınıflandırma sonucunu içerir.



Şekil 3.19. Otsu Metodu için sınıflandırma sonucu



Şekil 3.20. Mask R-CNN için sınıflandırma sonucu

Tablo 3.2, Otsu yöntemiyle elde edilen ikili maske ile Mask R-CNN ile elde edilen ikili maskenin performans sonuçlarını karşılaştırır. Tablo 3.2'de görülebileceği gibi, Mask R-CNN daha yüksek doğruluk elde etmiştir.

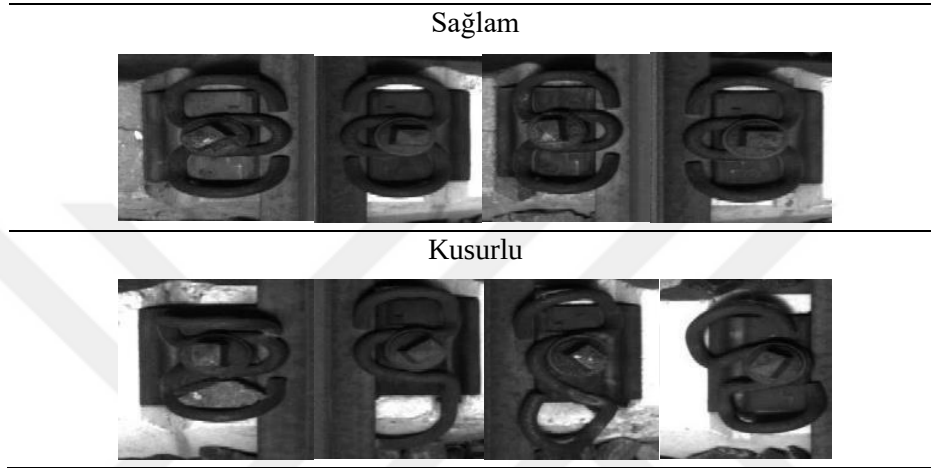
Tablo 3.2. İki farklı travers segmentasyon yöntemi için sonuçların karşılaştırılması

| Bölütleme Tipi | Accuracy | Recall | Precision | F1    |
|----------------|----------|--------|-----------|-------|
| Otsu Metodu    | 95,73    | 100    | 95,43     | 97,66 |
| Mask R-CNN     | 99,73    | 100    | 99,71     | 99,86 |

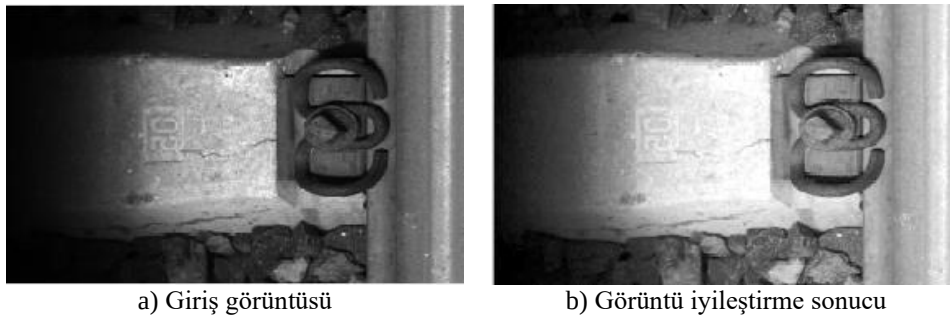
### 3.3. Görsel Kelime Çantası'na Dayalı Kusur Tespit Yöntemi İçin Deneysel Sonuçlar

Tezin bu bölümünde kullanılan veri kümesi 144 sağlam, 55 deforme olmak üzere toplam 199 bağlantı elemanı görüntüsünden oluşmaktadır. Veri kümesini oluşturan bazı sağlam ve deforme bağlantı elemanı örnekleri Tablo 3.3'te gösterilmiştir.

**Tablo 3.3.** Veri kümesinden rastgele seçilen bağlantı elemanı örnekleri

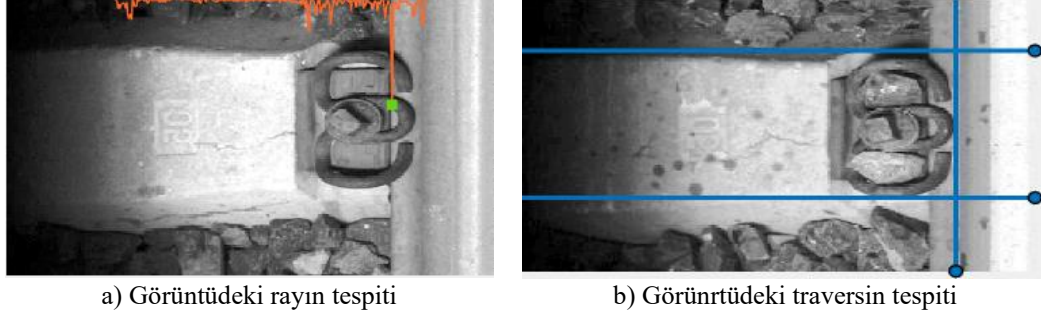


Tablo 3.3'te görüldüğü gibi bazı bağlantı elemanları eğilme ve kırılma gibi kusurlara sahiptir. Bağlantı elemanlarının konumlarını tespit etmeden önce görüntü ön işleme yapılmıştır. Görüntü ön işleme sonucunda görüntü kontrastı iyileştirilmiştir. Şekil 3.21'de görüntü iyileştirme işleminin sonucu gösterilmiştir.



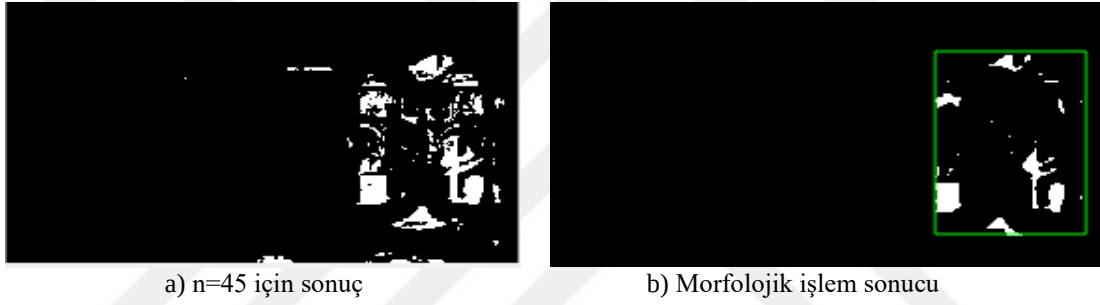
**Şekil 3.21.** Görüntü ön işleme sonucu

Görüntü ön işleme adımından sonra elde edilen görüntüde ray ve traverslerin konumu belirlenmiştir. Ray konumu için gereken hesaplar yapıldıktan sonra elde edilen en büyük değer rayın sınırıdır. Şekil 3.22.a'da yatay projeksiyon algoritmasının uygulanması gösterilmektedir. Travers sınırlarının tespit için yatay projeksiyon algoritması, Şekil 3.22.b'de gösterildiği gibi dikeyde çalıştırılmıştır.



**Şekil 3.22.** Görüntü üzerinde ray ve traverslerin tespit edilmesi

Travers ve rayın tespitinden sonra ilgili alan kırılmıştır. Kırılan görüntüye ÇYİÖ algoritması uygulanarak kontur çizgileri ile vidanın tam konumu belirlenmiştir. İşlemin sonucu Şekil 3.23'te verilmiştir.



**Şekil 3.23.** ÇYİÖ uygulaması ve bağlantı elemanı tespiti.

3.23.a'da görüldüğü gibi ÇYİÖ algoritması sonucunda bağlantı elemanının olduğu görüntü elde edilmiştir. Ancak elde edilen görüntüde gürültüler bulunmaktadır. Bu nedenle gürültüyü kaldırmak için morfolojik işlemler uygulanmıştır. Bağlantı elemanının konumu Şekil 3.23.b'de gösterildiği gibi bulunmuştur. Konum tespitinden sonra bağlantı elemanının kusur durumu incelenmiştir.

Görüntüler ilk olarak ORB algoritması ile test edilmiştir ve eğitim görüntülerine ait tanımlayıcılar listeye eklenmiştir. Ardından tanımlayıcılar, K-means algoritması kullanılarak kümelendirilmiştir. Kümeleme işleminde  $k$  değeri 200 olarak belirlenmiştir. Daha sonra görüntülerin histogramları oluşturulup standardizasyon işlemi uygulanmıştır. Son aşamada ise destek vektör makinesi ile sınıflandırma yapılmıştır. Önerilen yöntemin, karmaşıklık matrisi sonucu Şekil 3.24.a'da verilmiştir. Önerilen yöntem HOG ve SURF anahtar nokta çıkarım teknikleri için de test edilmiştir. Bu yöntemler için karmaşıklık matrisleri Şekil 3.24.b ve Şekil 3.24.c'de verilmiştir.

|             |                     |         |    |
|-------------|---------------------|---------|----|
| Doğru sınıf | Tahmin edilen sınıf |         |    |
|             | Sağlam              | Kusurlu |    |
|             | Sağlam              | 140     | 3  |
| Doğru sınıf | Kusurlu             | 4       | 52 |

a) ORB tabanlı özellik çıkarımı

|             |                     |         |    |
|-------------|---------------------|---------|----|
| Doğru sınıf | Tahmin edilen sınıf |         |    |
|             | Sağlam              | Kusurlu |    |
|             | Sağlam              | 139     | 12 |
| Doğru sınıf | Kusurlu             | 4       | 43 |

b) SURF tabanlı özellik çıkarımı

|             |                     |         |    |
|-------------|---------------------|---------|----|
| Doğru sınıf | Tahmin edilen sınıf |         |    |
|             | Sağlam              | Kusurlu |    |
|             | Sağlam              | 125     | 18 |
| Doğru sınıf | Kusurlu             | 9       | 46 |

c) HOG tabanlı özellik çıkarımı

**Şekil 3.24.** Farklı özellik çıkarım yöntemlerine göre başarımlar.

Tezin bu bölümünde, amaç bağlantı elemanlarını yüksek doğruluk oranı ile sınıflandırmaktır. Önerilen yöntemde yüksek doğruluk oranı elde edilmiştir. Test edilen üç yöntemin karşılaştırma sonuçları Tablo 3.4’te verilmiştir.

**Tablo 3.4.** Beş ayrı ölçüt için ORB, SURF ve HOG performans değerleri.

| Ölçüt (%)      | ORB          | SURF   | HOG   |
|----------------|--------------|--------|-------|
| Doğruluk       | <b>96.48</b> | 91.92  | 86.36 |
| Geri çağırma   | <b>97.22</b> | 97.20  | 93.28 |
| Kesinlik       | <b>97.90</b> | 92.05  | 87.41 |
| F-1            | <b>97.56</b> | 94.56  | 90.25 |
| Özellik sayısı | <b>1000</b>  | 129792 | 71604 |

Tablo 3.4 incelendiğinde ORB’ye dayalı yöntemde daha iyi performans elde edilmiştir. Literatürde farklı metotlar kullanılarak çalışmalar yapılmıştır. Bu tezde önerilen yöntem literatürdeki farklı yöntemlerle karşılaştırılmıştır. Sonuçlar, Tablo 3.5’te verilmiştir.

**Tablo 3.5.** Literatürde bulunan diğer yaklaşımlar ile karşılaştırma.

| Referans               | Özellik Çıkarımı                              | Sınıf Sayısı | Veri Boyutu | Sınıflandırma Metodu                           | Doğruluk (%) |
|------------------------|-----------------------------------------------|--------------|-------------|------------------------------------------------|--------------|
| [69]                   | Histogram tepe noktası kontrolü               | 2            | 745         | Temel bileşen analizi ve derinlik tepe yöntemi | 92.91        |
| [70]                   | Yönlendirilmiş gradyanların histogramı (HOG)  | 3            | 6445        | Bulanık kümeleme ve eşikleme                   | 95.50        |
| [71]                   | Histogram Eşitleme                            | 2            | 584         | Evrişimsel sinir ağı                           | 95.20        |
| [72]                   | Local ikili örüntü (LBP)                      | 3            | 6000        | Destek vektör makinası                         | 93.38        |
| [73]                   | Normalize edilmiş çapraz korelasyon           | 2            | 195         | Geometrik Düzenleme indeksi                    | 89.23        |
| <b>Önerilen Yöntem</b> | <b>Yüksek kontrastlı görüntü, ORB ve ÇYİÖ</b> | <b>2</b>     | <b>199</b>  | <b>Destek vektör makinası</b>                  | <b>96.48</b> |

Tablo 3.5 incelendiğinde önerilen yöntemin performansı farklı yöntemlere göre yüksektir. [69]’da histogram tepe noktası benzerlik algoritması kullanılmıştır. [70]’te yönlendirilmiş

gradyanlara ait histogram ve bulanık mantık tabanlı yöntem kullanılmıştır. [71]'de histogram eşitleme kullanılarak kusurlar belirlenmiştir. [72]'de, bağlantı elemanı tespiti için lokal ikili örüntü kullanılmıştır. [73]'te, Normaliz edilmiş çapraz korelasyon kullanılarak bağlantı elemanları tespit edilip kusurlar tespit edilmiştir. Önerilen yöntemde ise bağlantı elemanları başarıyla tespit edildikten sonra yüksek doğruluk oranıyla kusur tespiti yapılmıştır. Ayrıca, uygulanan görüntü iyileştirme algoritması ile bağlantı elemanının konumunun daha kolay tespiti sağlanmıştır. Bu nedenle önerilen yöntem literatüre önemli bir katkı sağlamaktadır.

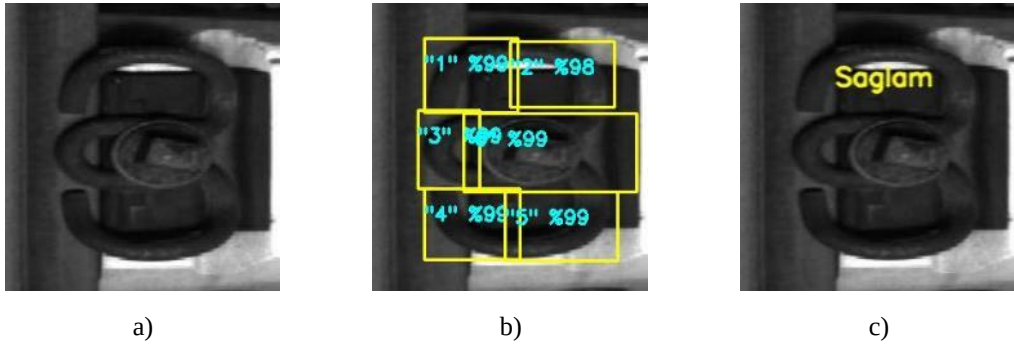


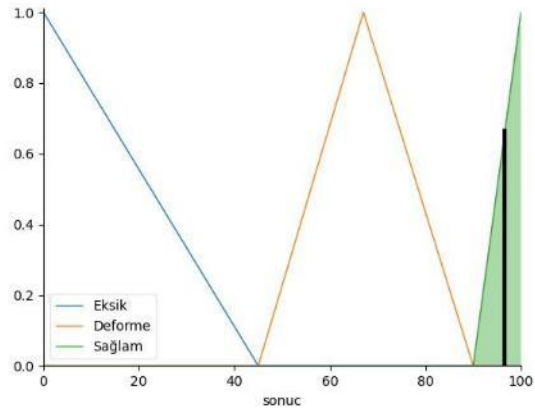
### 3.4. YOLOv4 ve Bulanık Mantık Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti İçin Deneysel Sonuçlar

Demiryolundan elde edilen görüntülerden, bağlantı elemanlarında farklı çeşitlerde kusurlar olduğu gözlenmiştir. Elde edilen bağlantı elemanı görüntüleri sağlam, deforme ve eksik olmak üzere üç sınıfa ayrılmıştır. Önerilen algoritma, üç ayrı sınıf için de test edilmiştir. Şekil 3.25'te sağlam, Şekil 3.26'da deforme ve Şekil 3.27'de eksik bağlantı elemanlarının bulunduğu senaryolar denenip sonuçlar gösterilmiştir. Bulanık mantık çıkışı %0 ile %45 arasında ise “eksik”, %45 ile %90 arasında ise “deforme”, %90 üzeri ise “sağlam” olarak sınıflandırılmıştır.

Şekil 3.25.a'da verilen bağlantı elemanı sağlam olduğu için Şekil 3.25.b'de görüldüğü gibi bağlantı elemanının parçalarının güven değerler yüksek değerli çıkmıştır. Bulanık mantık sonucu da yüksek çıktığından görüntü sağlam olarak nitelendirilmiştir (Şekil 3.25.c). Şekil 3.25.d'de verilen sonuç grafiğinde görüldüğü gibi göre bağlantı elemanı %90 üzeri sağlamlık oranı elde edilmiştir. Şekil 3.26.a'da verilen bağlantı elemanının alt bölümü eğildiği için Şekil 3.25.b'de görüldüğü gibi üstte kalan dört parça yüksek oranlı güven değerleri oluştururken alttaki iki parçanın düşük güven değerleri düşüktür. Bu da sonucun daha düşük değerlerde çıkmasını sağlamaktadır. Bu nedenle, Şekil 3.25.c'de görüldüğü gibi görüntü, deforme olarak sınıflandırılmıştır. Şekil 3.25.d'de verilen sonuç grafiğine göre bağlantı elemanın sağlamlık yüzdesi %45 ile %90 aralığındadır.

Şekil 3.26.a'da verilen görüntüde ise bağlantı elemanı kopmuştur. Bu nedenle, sadece ortadaki vida yüksek güven değeri oluştururken diğer parçalar düşük değerli ya da sıfırdır. Yerinde olmayan parçaların güven değerlerinin %0 olması sonuç yüzdesini düşürmektedir. Şekil 3.26.d'de görüldüğü gibi sonuç değeri %45'ten düşük olduğu için eksik olarak sınıflandırılmıştır. Sınıflandırma sonucu Şekil 3.26.c'de gösterilmiştir.



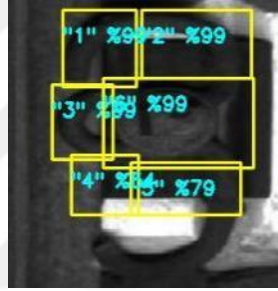


d)

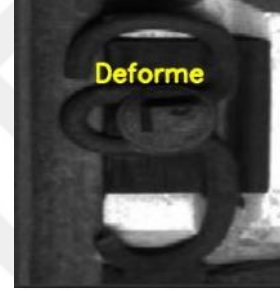
**Şekil 3. 25.** Sağlam bağlantı elemanı için elde edilen sonuçlar a) Giriş görüntüsü, b) YOLOv4 çıktısı, c) Sonuç d) Sonuç grafiği



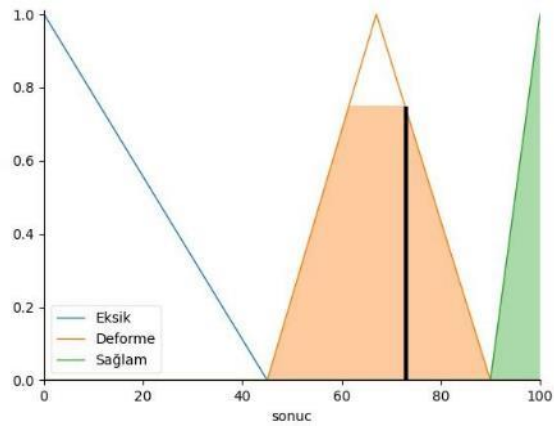
a)



b)

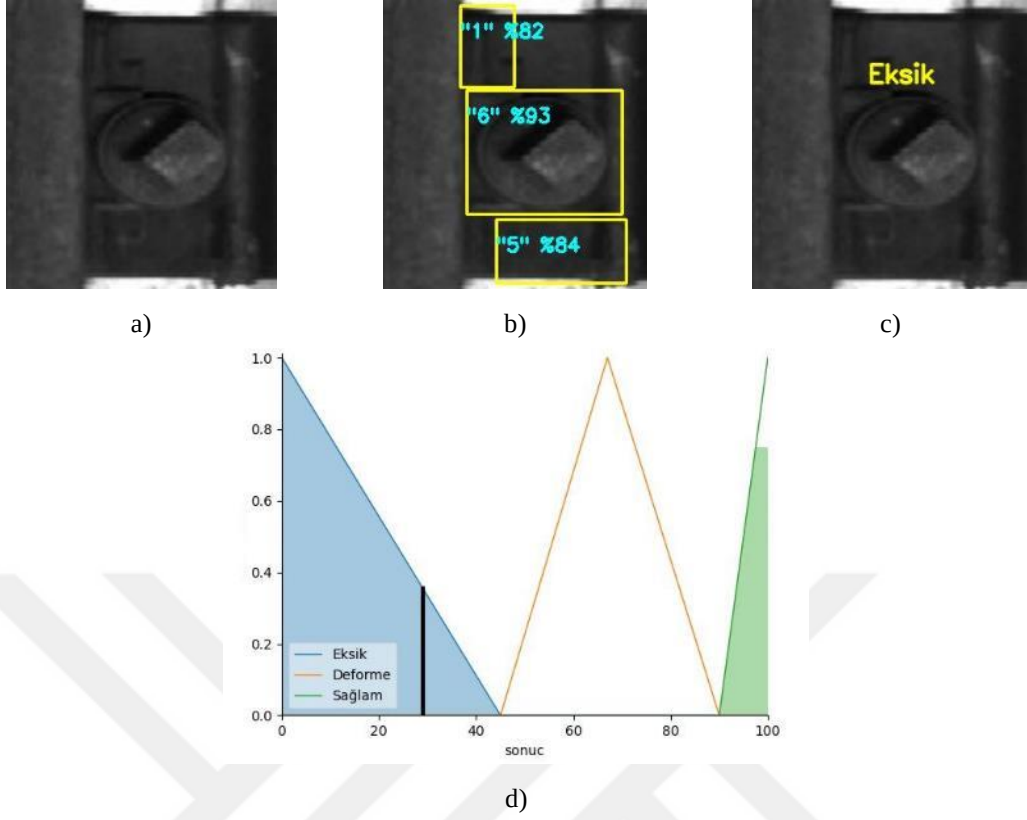


c)



d)

**Şekil 3. 26.** Deforme bağlantı elemanı için elde edilen sonuçlar a) Giriş görüntüsü, b) YOLOv4 çıktısı, c) Sonuç d) Sonuç grafiği



**Şekil 3. 27.** Eksik bağlantı elemanı için elde edilen sonuçlar a) Giriş görüntüsü, b) YOLOv4 çıktısı, c) Sonuç d) Sonuç grafiği

Yapılan çalışmada, tahmin başarı ölçütü için karmaşıklık matrisi kullanılmıştır. Karmaşıklık matrisi Tablo 3.6’da verilen parametrelerden oluşmaktadır.

**Tablo 3.6.** Karmaşıklık matrisi ölçüm değerleri için parametreler

| Parametre           | Açıklama                                                  |
|---------------------|-----------------------------------------------------------|
| Doğru Pozitif (DP)  | Gerçek sonuç sağlam olup sağlam olarak tahmin edilenler   |
| Doğru Negatif (DN)  | Gerçek sonuç deforme olup deforme olarak tahmin edilenler |
| Yanlış Pozitif (YP) | Gerçek sonuç sağlam olup deforme olarak tahmin edilenler  |
| Yanlış Negatif (YN) | Gerçek sonuç deforme olup sağlam olarak tahmin edilenler  |

Sınıflandırma sonuçlarını değerlendirmek için üç metrik ve kullanılmıştır. Bu metrikler doğruluk(accuracy), hassasiyet(precision), geri çağırma(recall) ve F1 değerleridir. Uygulamanın, karmaşıklık matrisi sonuçları Tablo 3.7’de verilmiştir.

**Tablo 3.7.** Bağlantı elemanı kusur tespiti için karmaşıklık matrisi

|             |         | Tahmin edilen sınıf |         |
|-------------|---------|---------------------|---------|
|             |         | Sağlam              | Kusurlu |
| Doğru sınıf | Sağlam  | 199                 | 1       |
|             | Kusurlu | 2                   | 198     |

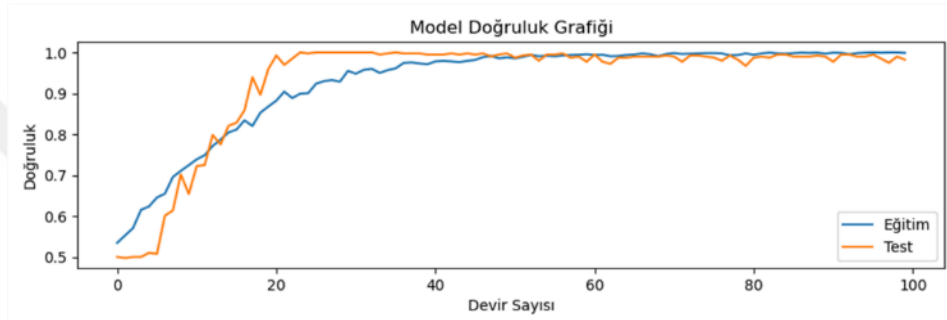
**Tablo 3.8.** Dört ayrı ölçüt için performans değerleri

| <b>Doğruluk(%)</b> | <b>Hassasiyet(%)</b> | <b>Geri Çağırma(%)</b> | <b>F1 (%)</b> |
|--------------------|----------------------|------------------------|---------------|
| 99.25              | 99                   | 99.5                   | 99.24         |

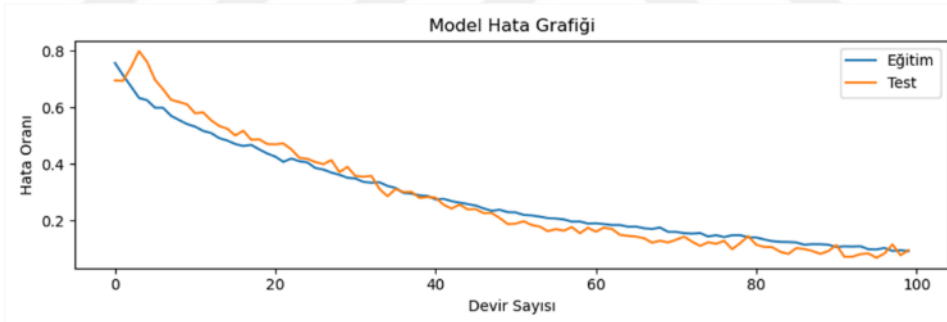
Önerilen algoritma, 400 bağlantı elemanı görüntüsü üzerinde test edilmiştir. 400 görüntü içerisinde 200 adet sağlam ve 200 adet kusurlu görüntü bulunmaktadır. Sonuçlar Tablo 3.8’de gösterilmiştir. Önerilen algoritma 400 adet görüntüden 397’sini doğru, 3’ünü ise yanlış değerlendirmiştir. Tablo 3.8’de farklı ölçütler için doğruluk değerleri verilmiştir. Elde edilen sonuçlar, önerilen YOLOv4 ve bulanık mantık tabanlı algoritmanın bağlantı elemanı kusur tespiti için yüksek performans ile çalıştığını göstermektedir.

### 3.5. DCGAN ve Siyam Sinir Ağı Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti İçin Deneysel Sonuçlar

Siyam sinir ağlarının kullanılma amacı, bağlantı elemanı görüntülerinin deforme ve sağlam olarak sınıflandırılmasıdır. Yapılan uygulamalar, Python 3.8 de derlenmiştir. Windows işletim sistemi kullanılmıştır. Derin öğrenme uygulamaları, TensorFlow çerçevesinde uygulanmıştır. Siyam sinir ağı modelini eğitmek için NVIDIA GeForce GTX 1060 kullanılmıştır. Optimizasyon yöntemi olarak Adam Optimizer kullanılmıştır. Batch değeri 32 olarak seçilip öğrenme hızı 0,0001 olarak belirlenmiştir. Devir sayısı 100 olarak ayarlanmıştır. Eğitim işlemi sonucunda oluşan eğitim ve test görüntülerinin doğruluk grafikleri Şekil 3.28 ve Şekil 3.29’da verilmiştir.



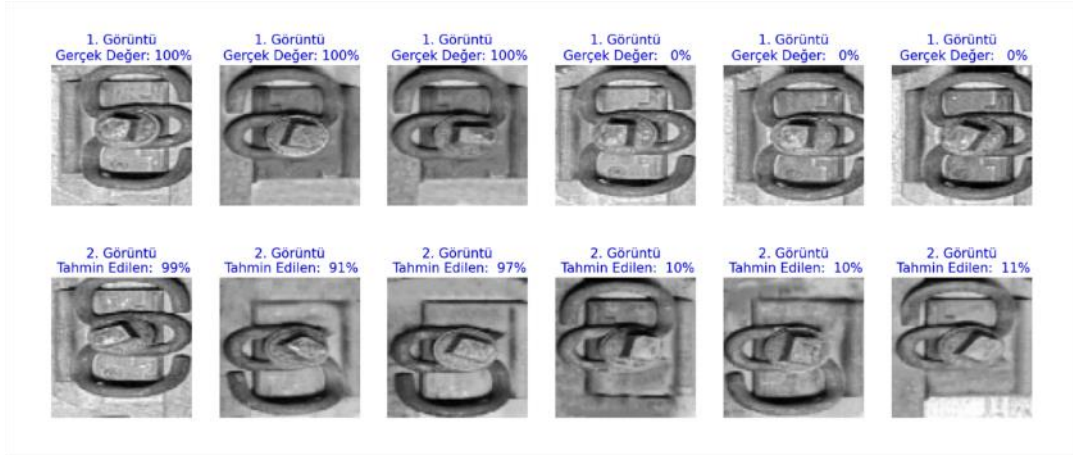
Şekil 3. 28. Deney sonucunda oluşan model doğruluk grafiği



Şekil 3. 29. Deney sonucunda oluşan model hata grafiği

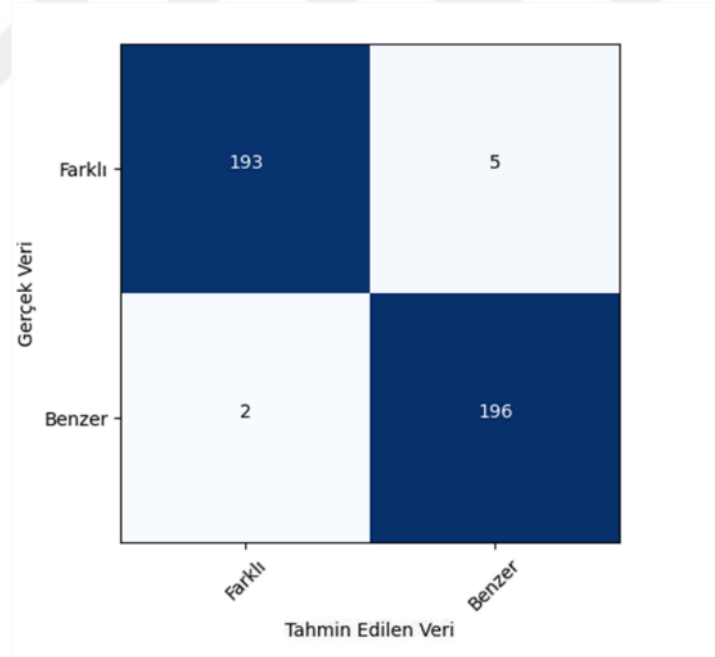
Eğitim işleminden sonra elde edilen model, iki görüntü arasındaki benzerliği tahmin etmek için kullanılmıştır. Ağı son katmanında Sigmoid aktivasyon fonksiyonu kullanıldığından 0 ile 1 aralığında bir çıktısı oluşacaktır. Sonucun 0 olması, iki görüntünün birbirinden tamamen farklı olduğu anlamına gelirken sonucun 1 olması ise iki görüntünün tamamen benzer olduğu anlamına gelmektedir. Örnek olarak, deforme ile sağlam görüntüler karşılaştırılırsa elde edilen sonuç 0'a yakınken iki adet sağlam bağlantı elemanını görüntüsü karşılaştırıldığında sonuç 1'e yakın olacaktır. Elde edilen sonuç 100 ile çarpılarak yüzde cinsinden değer elde edilmiştir. Veri kümesinden rastgele örnekler seçilip karşılaştırma yapılmıştır. Sonuçlar, Şekil 3.30'da verilmiştir. Karşılaştırmada elde edilen sonuç %50'nin altında ise görüntü farklı, %50'nin üzerinde ise

görüntüler aynı olarak belirlenmiştir. Rastgele yapılan eşleştirme sayısı 396'dır. Karşılaştırma sonuçlarından elde edilen karmaşıklık matrisi Şekil 3.31'de verilmiştir.



Şekil 3. 30. Eğitilen siyam ağına göre iki görüntünün benzerlik oranına göre karşılaştırılması

Karmaşıklık matrisi kullanılarak doğruluk, duyarlılık, özgünlük ve F1 değerlerini ölçülmüştür.



Şekil 3. 31. Test görüntülerinin eşleştirilmesi sonucu oluşan karmaşıklık matrisi

Tablo 3.9. Önerilen yöntemin performans sonuçları

| Doğruluk (%) | Duyarlılık (%) | Kesinlik (%) | F1 (%) |
|--------------|----------------|--------------|--------|
| 98,23        | 98,98          | 97,51        | 98,25  |

Önerilen yöntem ile elde edilen ölçüm sonuçları Tablo 3.9’da verilmiştir. Elde edilen deneysel sonuçlar, önerilen yöntemin yüksek doğruluk elde ettiğini göstermektedir. Ayrıca, kusur tespit hassasiyetinin yüksek olması önerilen yöntemin, bağlantı elemanlarında bulunan küçük kusurların da tespiti için kullanılabileceğini göstermektedir. Veri kümesi 2’den fazla sınıf olacak şekilde genişletilirse farklı kusur türlerinin tespitinde de kullanılabilir. Bu çalışmanın ana hedefi demiryolunun yüksek doğruluk oranlı bir denetim yöntemiyle denetlenmesi ve işgücü maliyetinin azaltılmasıdır.



### 3.6. Gürbüz Derin Öğrenme Kullanarak Bağlantı Elemanlarındaki Kusurların Tespiti İçin Deneysel Sonuçlar

Hızlı gradyan iniş yönteminin (FGSM) testi için ImageNet veri kümesi üzerinde önceden eğitilmiş ResNet mimarisi kullanılmıştır. Burada ağa girdi olarak bir yolcu otobüsü görüntüsü verildiğinde Şekil 3.32.a’de görüldüğü gibi %93.17 güven değeri ile “passenger\_car” olarak sınıflandırılmıştır. Yine aynı şekilde uçak görüntüsü verildiğinde Şekil 3.32.b’de görüldüğü gibi %99.12 güven değeri ile “airliner” olarak sınıflandırmıştır.



a)



b)

Şekil 3. 32. Gürültü eklenmeden önce sınıflandırma sonuçları

Giriş görüntüsüne FGSM yöntemi ile gürültü eklendiğinde ise modelin sınıflandırma başarımı oldukça düşmektedir. Şekil 3.33’te görüldüğü gibi yolcu otobüsünü %100 güven değeriyle “minibus” olarak uçağı ise %99.99 olarak “wing” olarak sınıflandırmaktadır.



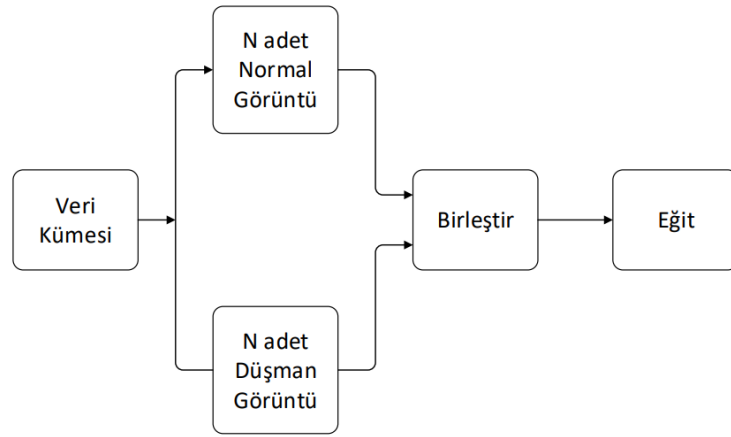
a)



b)

Şekil 3. 33. Gürültü eklendikten sonra sınıflandırma sonuçları

Bu sorunun üstesinden gelebilmek için rakip görüntüler ile orijinal görüntülerin karıştırılmasına dayalı bir yöntem önerilmektedir. Önerilen yöntemde, veri kümesi ikiye bölünecektir. Veri kümesinin yarısı orijinal olarak bırakılıp diğer yarısına FGSM metodu ile gürültü eklenerek rakip görüntüler oluşturulacaktır. Önerilen yöntemin akış diyagramı Şekil 3.34’te verilmiştir.



**Şekil 3. 34.** Önerilen yöntemin akış diyagramı

Çalışmada kullanılan veri kümesi 1000 sağlam, 1000 deforme ve 1000 eksik olmak üzere 3000 bağlantı elemanı görüntüsünden oluşmaktadır. Şekil 3.35’te veri kümesinden bazı örnek görüntüler gösterilmiştir.



**Şekil 3. 35.** Kullanılan veri kümesinden bazı görüntüler

Gürültü eklenmeden 100 Döngüde eğitim ve doğrulama için sonuçlar Tablo 3.10’da verilmiştir.

**Tablo 3.10.** Normal veri kümesi için sonuçlar

| Kayıp  | Doğruluk |
|--------|----------|
| 0.7511 | %83.95   |

Eğitilen ağın performansı, üretilen rakip görüntüler ile denendiğinde Tablo 3.11’de görüldüğü gibi ciddi oranda düşmüştür.

**Tablo 3.11.** Rakip görüntülerden oluşan veri kümesi için sonuçlar

| Kayıp    | Doğruluk |
|----------|----------|
| 850.5140 | %0.13    |

Veri kümesi, modelin sağlamlığını artırmak için Şekil 3'te gösterilen yöntemle karıştırılıp 100 döngü boyunca tekrar eğitilmiştir. Normal veri kümesi ile rakip görüntülerden oluşan veri kümesi ile tekrar test edildiğinde doğruluk oranında yüksek oranda artışlar görülmüştür. Normal veri kümesi eğitim ve doğrulama için sonuçlar Tablo 3.12'de verilmiştir.

**Tablo 3.12.** Tekrar eğitildikten sonra normal veri kümesi için sonuçlar

| Kayıp  | Doğruluk |
|--------|----------|
| 0.4972 | %91.45   |

Rakip görüntüler için test sonuçları Tablo 3.13'te verilmiştir. Görüldüğü gibi rakip örnekler için doğruluk oranı % 0.13 iken önerilen yöntem kullanılarak %91.45'e çıkarılmıştır.

**Tablo 3.13.** Tekrar eğitildikten sonra rakip görüntülerden oluşan veri kümesi için sonuçlar

| Kayıp  | Doğruluk |
|--------|----------|
| 0.0977 | %98.95   |

## 4. SONUÇLAR

Demiryolları, birçok ülkede başlıca ulaşım yollarından biridir ve yolcu ve yük taşımacılığı için sıklıkla kullanılır. Demiryolu altyapısı büyük ve maliyetli bir yatırımdır. Bu yatırımın faydalarını uzun yıllar korumak için etkin bakım gereklidir. Gerekli bakım yapılmazsa raydan çıkma gibi riskler olacaktır. Doğru bakım stratejisini belirlemek için demiryolu altyapısında ortaya çıkan kusurların ayrıntılı bir incelemesi gereklidir. Etkili ve zamanında yapılan bakım, potansiyel ray kırılması ve raydan çıkma riskini azaltabilir. Oluşabilecek potansiyel riskler, demiryolunda uzun vadede birikimli olarak kusur oluşturacaktır. Demiryollarında kusur tespiti için kullanılan çeşitli teknikler vardır. Bu çalışmada, kusur tespiti için İnsansız hava araçlarının kullanımı önerilmiştir.

İHA kullanılarak demiryollarındaki kusurların incelenmesi İHA'lar için nispeten yeni bir uygulamadır. İHA'nın demiryolunu otonom olarak takip etmesini sağlayarak, soruşturma maliyeti düşürür ve izleme sürecini hızlandırır. Ancak böyle bir uygulamada yol tespit ve takip algoritmasının iyi çalışması önemli bir konudur. Bu tezin ilk bölümünde, demiryollarındaki kusurları tespit etmek için verimli bir otonom izleme algoritması tasarlanmıştır. Farklı ray tespit algoritmaları ile ray tespiti yapılmıştır. Geliştirilen yöntemler sahada test edilip algoritmaların verimli çalıştığı ve İHA'nın otonom olarak demiryolunu sabit bir hızda takip edebildiği gözlemlenmiştir. Tezin ikinci bölümünde travers eksikliklerinin tespitine odaklanılmıştır. İki ayrı yöntem önerilerek traverslerin tespit edilmesi sağlanmıştır. Tezin diğer bölümlerinde ise bağlantı elemanlarını yüksek doğruluk oranıyla sınıflandırmaya odaklanılmıştır. Geliştirilen yöntemler, yüksek doğruluk oranlarıyla çalışmaktadır.

## ÖNERİLER

Bu tez kapsamında, İHA ile otonom ray takibi yapıp temassız inceleme yöntemleriyle bağlantı elemanlarında ve traverslerde bulunan kusurların tespiti için incelemeler yapılmıştır. Görüntüler İHA ile toplanmıştır. Gelecekteki çalışmalarda, görüntü toplamaya uygun iki kameralı İHA tasarımı yapılacaktır. Otonom İHA tarafından toplanan görüntülerden demiryolu hatalarının tespiti için çalışılacaktır. Derin öğrenme ve makine öğrenmesi gibi teknikler kullanılarak yeni yöntemlerin geliştirilmesi sağlanacaktır.



## KAYNAKLAR

- [1] Liu, S., Wang, Q., & Luo, Y. (2019). A review of applications of visual inspection technology based on image processing in the railway industry. *Transportation Safety and Environment*, 1(3), 185-204.
- [2] Janeliukstis, R., Clark, A., Papaalias, M., & Kaewunruen, S. (2019). Flexural cracking-induced acoustic emission peak frequency shift in railway prestressed concrete sleepers. *Engineering Structures*, 178, 493-505.
- [3] Teidj, S. (2020). Defect Indicators in a Rail Based on Ultrasound Generated by Laser Radiation. *Procedia Manufacturing*, 46, 863-870..
- [4] Zhang, J., Ma, H., Yan, W., & Li, Z. (2016). Defect detection and location in switch rails by acoustic emission and Lamb wave analysis: A feasibility study. *Applied Acoustics*, 105, 67-74.
- [5] Yaman, O., Karaköse, M., Akın, E., & Aydın, İ. (2015, May). Image processing based fault detection approach for rail surface. In *2015 23rd Signal Processing and Communications Applications Conference (SIU)* (pp. 1118-1121). IEEE.
- [6] GC/RT 5021 issue 5 (2011). Track System Requirements . (Cited on pages 16, 17, 20, 21, 26 and 166.)
- [7] T.C. Milli Eğitim Bakanlığı, 2011. Raylı Sistemler Teknolojisi, Demiryolu İnşaatı 582YIM282, Ankara.
- [8] P. K. N, R. K. A, S. Anand, Dr. E. N. Ganesh, and Dr. V. Prithiviraj, “Retracted: Quad Band Signal Strength Monitoring System Using Quadcopter and Quad Phone,” *Journal of Green Engineering*, vol. 5, no. 1, pp. 1–22, 2015, doi: 10.13052/jge1904-4720.511.
- [9] V. Kadamatt, “Tricopter vs Quadcopter vs Hexacopter: A Comparison,” *Droney Bee*, 2017. [Online].
- [10] Abd Rahman, Y. A., Hajibeigy, M. T., Al-Obaidi, A. S. M., & Cheah, K. H. (2018). Design and fabrication of small vertical-take-off-landing unmanned aerial vehicle. In *MATEC Web of Conferences* (Vol. 152, p. 02023). EDP Sciences.
- [11] Aytekin, Ç., Rezaeitabar, Y., Dogru, S., & Ulusoy, I. (2015). Railway fastener inspection by real-time machine vision. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(7), 1101-1107.
- [12] Liu, J., Huang, Y., Zou, Q., Tian, M., Wang, S., Zhao, X., ... & Ren, S. (2019). Learning visual similarity for inspecting defective railway fasteners. *IEEE Sensors Journal*, 19(16), 6844-6857.
- [13] Zhang, H., Song, Y., Chen, Y., Zhong, H., Liu, L., Wang, Y., ... & Wu, Q. J. (2021). MRSDI-CNN: Multi-Model Rail Surface Defect Inspection System Based on Convolutional Neural Networks. *IEEE Transactions on Intelligent Transportation Systems*.
- [14] Yang, H., Wang, Y., Hu, J., He, J., Yao, Z., & Bi, Q. (2021). Deep Learning and Machine Vision-Based Inspection of Rail Surface Defects. *IEEE Transactions on Instrumentation and Measurement*, 71, 1-14.
- [15] Wei, X., Yang, Z., Liu, Y., Wei, D., Jia, L., & Li, Y. (2019). Railway track fastener defect detection based on image processing and deep learning techniques: A comparative study. *Engineering Applications of Artificial Intelligence*, 80, 66-81.

- [16] Chen, J., Liu, Z., Wang, H., Nunez, A., & Han, Z. (2017). Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network. *IEEE Transactions on Instrumentation and Measurement*, 67(2), 257-269.
- [17] Dubey, A. K., & Jaffery, Z. A. (2016). Maximally stable extremal region marking-based railway track surface defect sensing. *IEEE Sensors Journal*, 16(24), 9047-9052.
- [18] Singh, M., Singh, S., Jaiswal, J., & Hemphshall, J. (2006, October). Autonomous rail track inspection using vision based system. In *2006 IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety* (pp. 56-59). IEEE.
- [19] Li, Y., Trinh, H., Haas, N., Otto, C., & Pankanti, S. (2013). Rail component detection, optimization, and assessment for automatic rail track inspection. *IEEE Transactions on Intelligent Transportation Systems*, 15(2), 760-770.
- [20] Liu, J., Li, B., Xiong, Y., He, B., & Li, L. (2015). Integrating the symmetry image and improved sparse representation for railway fastener classification and defect recognition. *Mathematical Problems in Engineering*, 2015.
- [21] Gibert, X., Patel, V. M., & Chellappa, R. (2016). Deep multitask learning for railway track inspection. *IEEE transactions on intelligent transportation systems*, 18(1), 153-164.
- [22] Surinta, O., & Khruahong, S. (2019). Tracking people and objects with an autonomous unmanned aerial vehicle using face and color detection. In *2019 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT-NCON)* (pp. 206-210). IEEE.
- [23] Li, Y., Fu, C., Huang, Z., Zhang, Y., & Pan, J. (2020, May). Keyfilter-aware real-time uav object tracking. In *2020 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 193-199). IEEE.
- [24] Guan, L., Li, X., Yang, H., & Jia, L. (2020, August). A Visual Saliency Based Railway Intrusion Detection Method by UAV Remote Sensing Image. In *2020 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)* (pp. 291-295). IEEE.
- [25] Wu, Y., Qin, Y., Wang, Z., & Jia, L. (2018). A UAV-based visual inspection method for rail surface defects. *Applied sciences*, 8(7), 1028.
- [26] Bojarczak, P., & Lesiak, P. (2021). UAVs in rail damage image diagnostics supported by deep-learning networks. *Open Engineering*, 11(1), 339-348.
- [27] Jarrett, C., Perry, K., & Stol, K. A. (2015, February). Controller comparisons for autonomous railway following with a fixed-wing UAV. In *2015 6th International Conference on Automation, Robotics and Applications (ICARA)* (pp. 104-109). IEEE.
- [28] Banić, M., Miltenović, A., Pavlović, M., & Ćirić, I. (2019). Intelligent machine vision based railway infrastructure inspection and monitoring using UAV. *Facta Universitatis, Series: Mechanical Engineering*, 17(3), 357-364.
- [29] Sahebdivani, S., Arefi, H., & Maboudi, M. (2020). Rail Track Detection and Projection-Based 3D Modeling from UAV Point Cloud. *Sensors*, 20(18), 5220.
- [30] Shukla, A., Xiaoqian, H., & Karki, H. (2016, October). Autonomous tracking and navigation controller for an unmanned aerial vehicle based on visual data for inspection of oil and gas pipelines. In *2016 16th international conference on control, automation and systems (ICCAS)* (pp. 194-200). IEEE.
- [31] Zouaoui, S., Mohamed, E., & Kouider, B. (2019). Easy tracking of UAV using PID controller. *Periodica Polytechnica Transportation Engineering*, 47(3), 171-177.

- [32] Rosales, C., Soria, C. M., & Rossomando, F. G. (2019). Identification and adaptive PID Control of a hexacopter UAV based on neural networks. *International journal of Adaptive control and signal processing*, 33(1), 74-91.
- [33] Al-Qadi, I., Xie, W., & Roberts, R. (2010). Optimization of antenna configuration in multiple-frequency ground penetrating radar system for railroad substructure assessment. *NDT & E International*, 43(1), 20-28.
- [34] A. Uğur, "Investigation of the world railways sector development prospects and turkey's status," *Alphanumeric j.*, vol. 7, no. 2, pp. 369–398, 2019.
- [35] Yuan, Z., Zhu, S., Yuan, X., & Zhai, W. (2021). Vibration-based damage detection of rail fastener clip using convolutional neural network: Experiment and simulation. *Engineering Failure Analysis*, 119, 104906.
- [36] Kanis, J., Zitrický, V., Hebelka, V., Lukáč, P., & Kubín, M. (2021). Innovative diagnostics of the railway track superstructure. *Transportation Research Procedia*, 53, 138-145.
- [37] de la Calle Herrero, F. J., García, D. F., & Usamentiaga, R. (2018). Inspection system for rail surfaces using differential images. *IEEE Transactions on Industry Applications*, 54(5), 4948-4957.
- [38] Farahani, B. V., Barros, F., Sousa, P. J., Tavares, P. J., & Moreira, P. M. (2020). A railway tunnel structural monitoring methodology proposal for predictive maintenance. *Structural Control and Health Monitoring*, 27(8), e2587.
- [39] Mariani, S., & di Scalea, F. L. (2018). Predictions of defect detection performance of air-coupled ultrasonic rail inspection system. *Structural Health Monitoring*, 17(3), 684-705.
- [40] Ramatlo, D. A., Long, C. S., Loveday, P. W., & Wilke, D. N. (2022). Physics-based modelling and simulation of reverberating reflections in ultrasonic guided wave inspections applied to welded rail tracks. *Journal of Sound and Vibration*, 530, 116914.
- [41] Montinaro, N., Epasto, G., Cerniglia, D., & Guglielmino, E. (2020). Laser ultrasonics for defect evaluation on coated railway axles. *NDT & E International*, 116, 102321.
- [42] Sánchez-Rodríguez, A., Soilán, M., Cabaleiro, M., & Arias, P. (2019). Automated inspection of railway tunnels' power line using LiDAR point clouds. *Remote Sensing*, 11(21), 2567.
- [43] Geng, Y., Pan, F., Jia, L., Wang, Z., Qin, Y., Tong, L., & Li, S. (2021). UAV-LiDAR-based measuring framework for height and stagger of high-speed railway contact wire. *IEEE Transactions on Intelligent Transportation Systems*.
- [44] Zhan, Z., Sun, H., Yu, X., Yu, J., Zhao, Y., Sha, X., ... & Li, W. J. (2019). Wireless rail fastener looseness detection based on MEMS accelerometer and vibration entropy. *IEEE Sensors Journal*, 20(6), 3226-3234.
- [45] Stypułkowski, K., Gołda, P., Lewczuk, K., & Tomaszewska, J. (2021). Monitoring System for Railway Infrastructure Elements Based on Thermal Imaging Analysis. *Sensors*, 21(11), 3819.
- [46] Chen, Q., Niu, X., Zuo, L., Zhang, T., Xiao, F., Liu, Y., & Liu, J. (2018). A railway track geometry measuring trolley system based on aided INS. *Sensors*, 18(2), 538.
- [47] Trinh, H., Haas, N., Li, Y., Otto, C., & Pankanti, S. (2012, January). Enhanced rail component detection and consolidation for rail track inspection. In *2012 IEEE Workshop on the Applications of Computer Vision (WACV)* (pp. 289-295). IEEE.
- [48] Li, Z. W., Liu, X. Z., & He, Y. L. (2019). Identification of temperature-induced deformation for HSR slab track using track geometry measurement data. *Sensors*, 19(24), 5446.

- [49] Sural, S., Qian, G., & Pramanik, S. (2002, September). Segmentation and histogram generation using the HSV color space for image retrieval. In *Proceedings. International Conference on Image Processing* (Vol. 2, pp. II-II). IEEE.
- [50] Shaik, K. B., Ganesan, P., Kalist, V., Sathish, B. S., & Jenitha, J. M. M. (2015). Comparative study of skin color detection and segmentation in HSV and YCbCr color space. *Procedia Computer Science*, 57, 41-48.
- [51] Rong, W., Li, Z., Zhang, W., & Sun, L. (2014, August). An improved CANNY edge detection algorithm. In *2014 IEEE international conference on mechatronics and automation* (pp. 577-582). IEEE.
- [52] Vincent, O. R., & Folorunso, O. (2009, June). A descriptive algorithm for sobel image edge detection. In *Proceedings of informing science & IT education conference (InSITE)* (Vol. 40, pp. 97-107). Informing Science Institute California.
- [53] Yang, L., Wu, X., Zhao, D., Li, H., & Zhai, J. (2011, October). An improved Prewitt algorithm for edge detection based on noised image. In *2011 4th International congress on image and signal processing* (Vol. 3, pp. 1197-1200). IEEE.
- [54] Nguyen, T. B., & Ziou, D. (2000). Contextual and non-contextual performance evaluation of edge detectors. *Pattern Recognition Letters*, 21(9), 805-816.
- [55] Johannes Traa. Least-Squares Intersection of Lines (UIUC 2013).
- [56] Gu, G., Ko, B., Go, S., Lee, S. H., Lee, J., & Shin, M. (2021). Towards Real-time and Light-weight Line Segment Detection. *arXiv preprint arXiv:2106.00186*.
- [57] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961-2969).
- [58] Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.
- [59] Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
- [60] Bolya, D., Zhou, C., Xiao, F., & Lee, Y. J. (2019). Yolact: Real-time instance segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 9157-9166).
- [61] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740-755). Springer, Cham.
- [62] Chieng, H. H., Wahid, N., Ong, P., & Perla, S. R. K. (2018). Flatten-T Swish: a thresholded ReLU-Swish-like activation function for deep learning. *arXiv preprint arXiv:1812.06247*.
- [63] Ahmed, B., Gulliver, T. A., & alZahir, S. (2020). Image splicing detection using mask-RCNN. *Signal, Image and Video Processing*, 14(5), 1035-1042.
- [64] Johnson, J. W. (2018). Adapting mask-rcnn for automatic nucleus segmentation. *arXiv preprint arXiv:1805.00500*.
- [65] Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1), 62-66.
- [66] Ojala, T., Pietikainen, M., & Maenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7), 971-987.

- [67] Fan, H., Cosman, P. C., Hou, Y., & Li, B. (2018). High-speed railway fastener detection based on a line local binary pattern. *IEEE Signal Processing Letters*, 25(6), 788-792.
- [68] Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011, November). ORB: An efficient alternative to SIFT or SURF. In *2011 International conference on computer vision* (pp. 2564-2571). Ieee.
- [69] Aytekin, Ç., Rezaeitabar, Y., Dogru, S., & Ulusoy, I. (2015). Railway fastener inspection by real-time machine vision. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(7), 1101-1107.
- [70] He, B., Luo, J., Ou, Y., Xiong, Y., & Li, B. (2020). Railway Fastener Defects Detection under Various Illumination Conditions using Fuzzy C-Means Part Model. *Transportation Research Record*, 0361198120977182.
- [71] Ma, A., Lv, Z., Chen, X., Li, L., Qiu, Y., Zheng, S., & Chai, X. (2020). Pandrol track fastener defect detection based on local convolutional neural networks. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 0959651820953679.
- [72] Fan, H., Cosman, P. C., Hou, Y., & Li, B. (2018). High-speed railway fastener detection based on a line local binary pattern. *IEEE Signal Processing Letters*, 25(6), 788-792.
- [73] Taheri, N., Nejad, F. M., & Zakeri, H. (2019). A Brief Overview and New Knowledge Based System for Rail Direct Fastening Evaluation Using Digital Image Processing. *Archives of Computational Methods in Engineering*, 1-19.
- [74] Boulkroune, A. (2016). A fuzzy adaptive control approach for nonlinear systems with unknown control gain sign. *Neurocomputing*, 179, 318-325.
- [75] Zadeh, L. A. (1965). Zadeh, Fuzzy sets. *Inform Control*, 8, 338-353.
- [76] X. Dong et al., "Fast efficient algorithm for enhancement of low lighting video," in *2011 IEEE International Conference on Multimedia and Expo*, 2011, pp. 1–6.
- [77] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2341–2353, 2011.
- [78] P. L. Suarez, A. D. Sappa, and B. X. Vintimilla, "Infrared image colorization based on a triplet DCGAN architecture," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 18–23.
- [79] I. J. Goodfellow et al., "Generative Adversarial Nets," *Neurips.cc*. [Online]. Available: <https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf>. [Accessed: 31-Oct-2021].
- [80] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv [cs.LG]*, 2015.
- [81] J. Bromley et al., "Signature verification using a 'Siamese' time delay neural network," *Intern. J. Pattern Recognit. Artif. Intell.*, vol. 07, no. 04, pp. 669–688, 1993.
- [82] A. F. Agarap, "Deep Learning using Rectified Linear Units (ReLU)," *arXiv [cs.NE]*, 2018.
- [83] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

## ÖZGEÇMİŞ

# Emre GÜÇLÜ

\_\_\_\_\_

| Category   | Value   |
|------------|---------|
| Category 1 | Value 1 |
| Category 2 | Value 2 |
| Category 3 | Value 3 |
| Category 4 | Value 4 |
| Category 5 | Value 5 |
| Category 6 | Value 6 |
| Category 7 | Value 7 |

\_\_\_\_\_

\_\_\_\_\_  
 \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

© 2006 The Authors

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

\_\_\_\_\_

- [REDACTED]  
[REDACTED]  
[REDACTED]
- [REDACTED]  
[REDACTED]  
[REDACTED]
- [REDACTED]  
[REDACTED]  
[REDACTED]
- [REDACTED]  
[REDACTED]  
[REDACTED]

[REDACTED]

- [REDACTED]  
[REDACTED]
- [REDACTED]  
[REDACTED]