



## YOLOv5 ile Topluluk Öğrenmesine Dayalı Olarak Ray Yüzeyindeki Kusurların Tespiti

Mehmet SEVİ<sup>1</sup>, İlhan AYDIN<sup>2</sup>, Erhan AKIN<sup>2</sup>

<sup>1</sup> Muş Alparslan Üniversitesi, Bilgi İşlem Daire Başkanlığı, Muş, Türkiye

<sup>2</sup> Fırat Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği, Elâzığ, Türkiye

\*m.sevi@alpaslan.edu.tr

(Alınış/Received: 15.11.2022, Kabul/Accepted: 15.12.2022, Yayınlama/Published: 31.01.2023)

**Öz:** Demiryolu ulaşımı son yıllarda demiryolu hat uzunluğunun artmasıyla beraber kapasitesini arttırmıştır. Hızlı trenlerin gelişmesi de bu duruma katkı sağlamıştır. Yolcu ve yük kapasitesinin artması güvenlik tedbirlerinin önemini daha da arttırmıştır. Demiryolu hatlarının güvenliğini sağlamak için hatların belirli aralıklarla denetlenmesi gerekmektedir. Demiryolu hattı bakımında ray üzerinde bulunan kusurların tespiti son derece önemlidir. Bu çalışmada demiryolu bakımının önemli bir parçası olan ray bileşeni üzerindeki kusurların tespitine odaklanılmıştır. Çalışmada ray üzerinde bulunan kusurları bir nesne tespiti yöntemi olan YOLO ile tespit etme yoluna gidilmiştir. Farklı YOLO modelleri için topluluk öğrenmesine dayalı bir yöntem önerilmiştir. Deneysel sonuçları, 8 farklı kusur içeren veri seti üzerinde bütün sınıfları içeren tespit oranının %80'in üzerinde olduğunu göstermiştir.

**Anahtar kelimeler:** Demiryolu, Ray, Kusur tespiti, Bilgisayarlı görme, YOLO, Derin öğrenme algoritmaları

### Detection of Rail Surface Defects Based on Ensemble Learning of YOLOv5

**Abstract:** Railway transportation has increased its capacity with the increase in railway line length in recent years. The development of high-speed trains also contributed to this situation. The increase in passenger and cargo capacity has further increased the importance of security measures. In order to ensure the safety of the railway line, it is necessary to inspect the line at certain intervals. Detection of defects on the rail is extremely important in the maintenance of the railway line. This study focuses on the detection of defects on the rail component, which is an important part of railway maintenance. In the study, it was tried to detect the defects on the rail with YOLO, which is an object detection method. In the study, it has been shown that model ensembling gives better results than YOLO models that validate alone. A method based on ensemble learning is proposed for different YOLO models. Experiment results showed that the detection rate including all classes on the data set containing 8 different defects was over 80%.

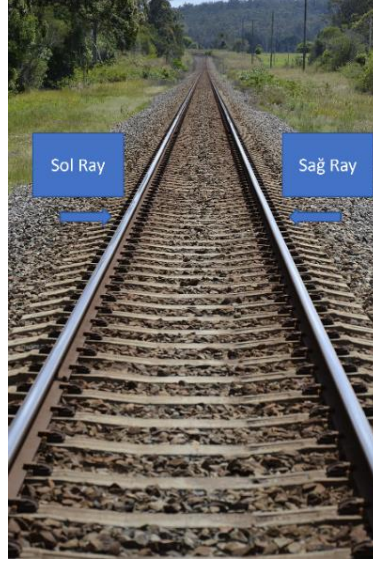
**Keywords:** Railway, Rail, Defect detection, Computer vision, YOLO, Deep learning algorithms

## 1. Giriş

Türkiye’de hızlı tren kullanımı demiryolu ulaşımına yapılan yatırımlarla beraber artmıştır. Ulaşımın yanı sıra demiryolu hatları yük taşımacılığında da yüksen oranda kullanılmaktadır. Maddi ve manevi hasarları minimuma indirmek için demiryolu hatlarının güvenliği son derece önemlidir. Bu yüzden demiryolu hatları düzenli bir şekilde izlenmeli, demiryolu hattında oluşabilecek kusurlar hızlı bir şekilde tespit edilmelidir. Günümüzde kullanılan demiryolu kusur tespit sistemleri manuel şekilde yapılmaktadır. Güncel sistemlerde demiryolu hattının izlenmesi ve denetimi bir uzman tarafından gözle gerçekleştirilmektedir [1]. Bu yöntemde demiryolu hattının kontrolü uzmanın görme kapasitesi ile sınırlıdır. Bu sebeple denetim ve izleme işleminde bir standart yakalamak mümkün değildir. Günümüzde kullanılan bir diğer yöntem ise demiryolu hattı ile temas sağlayarak yapılmaktadır. Bu yöntemin dezavantajı ise demiryolu hattında temastan kaynaklı kusurlar oluşabilmektedir. Ayrıca bu işlem sırasında demiryolu trafiği sekteye uğramaktadır. Bu yöntemler zaman, para ve iş gücü kaybına neden olmaktadır. Bu nedenle

Atıf için/Cite as: M. Sevi, İ. Aydın, E. Akın, “Detection of rail surface defects based on ensemble learning of YOLOv5,” *Demiryolu Mühendisliği*, no. 17, pp. 115-132, Jan. 2023. doi: 10.47072/demiryolu.1205483

demiryolu ulaşımının güvenliği sekteye uğrayabilir. Demiryolu ulaşımı, tüm dünyadaki en güvenli ulaşım türlerinden biri olarak kabul edilmektedir. Demiryolu bileşenlerinin kalitesi, trenin çalışma güvenliği için önemlidir. Demiryolu ulaşımının güvenle devam edebilmesi için demiryolu hattını oluşturan bileşenlerin bakımı düzenli bir şekilde yapılmalıdır. Demiryolu taşımacılığında araçların (tren, vagon) tekerlekleri ile üzerinde hareket ettikleri paralel biçimde yerleştirilmiş metal çubuklara ise ray denilmektedir. Bir demiryolu hattında sol ve sağ ray olmak üzere toplam iki adet ray vardır. Şekil 1'de sağlıklı sol ve sağ ray görülmektedir.



Şekil 1. Sol ve sağ ray bileşeni

Rayların sağlık durumu demiryolu operasyonlarının güvenliğini doğrudan etkilemektedir [2]. Sol ve sağ raylar genellikle hasarlı, kırık veya eksik gibi kusurlar içermektedir [3]. Demiryolu ulaşımında güvenliği sağlamak için akademik literatürde birtakım çalışmalar yapılmıştır. Bu çalışmalar genellikle görüntü işleme tabanlı uygulamalardır. Yapay zekâ alanının alt dallarından biri olan derin öğrenme yöntemleri, görüntülerden nesne tespiti yapma alanında kullanılan görüntü işleme algoritmaları birlikte kullanılmaktadır. Derin öğrenme sadece demiryolu alanında değil inşaat, sağlık gibi pek çok alanda kullanılmaktadır [4]. Literatürdeki derin öğrenme ağlarını kullanan çözümler demiryollarındaki kusurları tespit etmek için video ve resimlerden oluşan veri kümelerine ihtiyaç duymaktadır. Demiryolu hattındaki kusurları tespit eden yeni sistemler genellikle yeni teknolojilerin sunduğu hız, yüksek doğruluk ve maliyet gibi avantajlardan faydalanmaktadırlar. Feng ve ark. demiryolu hattındaki eksik bağlantı elemanlarını tespit etmek için otomatik bir görsel inceleme sistemi önermişlerdir. Önerdikleri sistemde bir tren vagonunun altına yerleştirilmiş, her iki rayı da kayıt altına alan iki kamera ile görüntüler toplanmıştır. Bu görüntüleri kullanarak önce bağlantı elemanları tespit edilmiş sonra ise geliştirdikleri STM modeli sayesinde bağlantı elemanlarını %98 oranında ayırt edebilmişlerdir [5]. Guo ve ark. YOLOv4-hibrit modelini kullanarak demiryolu bileşenlerini tespit etmişlerdir. Çalışmada herkese açık bir veri seti kullanmışlardır. Özelleştirilmiş YOLOv4-hibrit modeli ile 94,4 ortalama hassasiyet (mAP) ve saniyede 78,7 kare (FPS) performansı yakalamıştır [6]. Derin öğrenme tabanlı bir diğer çalışmada ray ve bağlantı elemanları YOLOv5 modeli ile tespit edilmiştir. Zheng ve ark. yaptıkları çalışmada ray yüzeyindeki ve bağlantı elemanlarındaki kusurları araştırmışlardır. Çalışma için RSDD veri setini etiketlenmiştir. Daha sonra derin öğrenme ağı ile demiryolu bileşenlerinin görüntü üzerindeki konumunu saptamışlardır. Son olarak ray ve bağlantı elemanları üzerindeki kusurları Mask R-CNN ve ResNet ağı ile tespit etmişlerdir [7]. Bir diğer çalışmada Adaboost tabanlı kırık bağlantı elemanlarını tespit eden bir yöntem önerilmiştir. Öncelikle demiryolu görüntüsünden bağlantı elemanlarının pozisyonları tespit edilmiştir. Tespit edilen bağlantı elemanı görsel olarak dört parçaya bölünerek Adaboost algoritması tarafından

tanınmıştır [8]. Fan ve ark. düşük aydınlatma şartlarında bağlantı elemanlarını başarıyla tespit eden ve hangi tür kusur içerdiğini belirleyen bir yöntem önermişlerdir [9]. Literatürdeki çalışmalar incelendiğinde ray üzerindeki kusurları tespit etme ve içerdiği kusura göre sınıflandırmada yapay zekâ ve görüntü işleme tabanlı yöntemlerin kullanıldığı görülmüştür [10-11]. Yanan ve ark. bu doğrultuda rayların içerdikleri kusurları YOLOv3 modelini kullanarak tespit etmişlerdir [12]. Bu çalışmada, rayların içerdikleri kusurlar derin öğrenme yoluyla saptanmıştır. Demiryolu ulaşımı güvenliği için yüksek doğruluk oranıyla bu işlemin yapılması esastır. Bu işlemi geleneksel yöntemlerle gerçekleştirmek zaman ve maliyet açısından son derece dezavantajlıdır. Bu nedenle, demiryolu ulaşımı güvenliği için bu verilerin otomatik analizi daha verimlidir. Bu çalışmada sahadan toplanan demiryolu görüntülerini kullanarak rayların içerdiği kusurları tespit eden YOLOv4, YOLOv5 ve YOLOv6 tabanlı bir yöntem önerilmiştir. Makalenin ikinci kısmında önerilen yöntemden, veri setinden ve ön işleme adımlardan bahsedilmiştir. Üçüncü kısımda ise deneysel sonuçlara değinilmiştir. Son kısımda ise önerilen yöntemin sonuçlarına değinilmiştir.

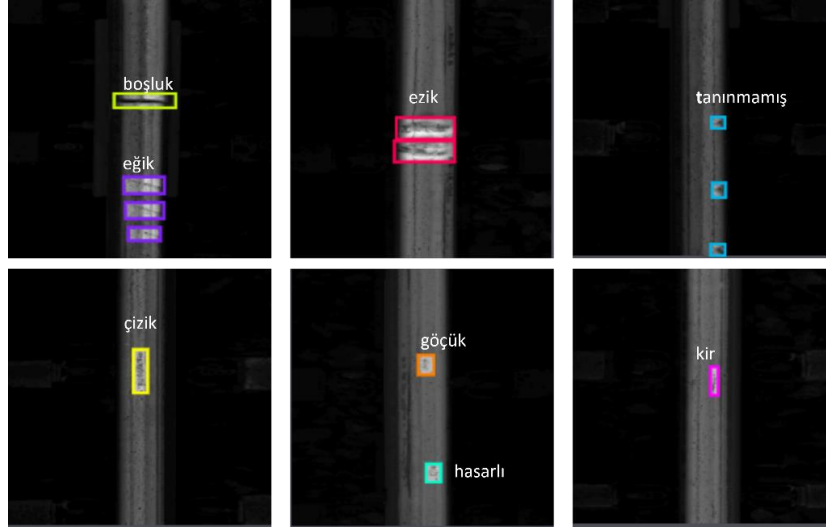
## 2. Metot

Önerilen yöntemde, ray üzerindeki kusurların tespiti için YOLOv4, YOLOv5 ve YOLOv6 modelleri uygulanmıştır. Şekil 2'deki görüntüde görüldüğü üzere ray üzerindeki kusurlar farklı kusur çeşitlerini içermektedir. Aynı zamanda tek görüntüde birden fazla kusur bulunabilmektedir. YOLO tabanlı modellerin tercih edilmesinin sebebi hızlı tahminler yaparken aynı zamanda da çok iyi sonuçlar verebilmeleridir. Örneğin R-CNN gibi bölge bazlı nesne tespit algoritmaları önce muhtemel alanları belirleyip ardından ayrı ayrı evrimsel sinir ağı (CNN) sınıflandırıcıları yürütmektedir. Bu yöntem her ne kadar iyi sonuçlar verse de bir görüntü iki ayrı işleme tabi tutulduğu için görüntü üzerindeki işlem sayısı artar ve düşük bir FPS (saniye başına kare) alınmasına sebep olmaktadır.

Çalışmada kullanılan veri seti 399 adet görüntüden oluşmaktadır [13]. Kusurlu demiryolu ray görüntüleri görüntü artırım teknikleriyle 939'a artırılmıştır. Bu görüntü artırım teknikleri görüntü çevirme, parlaklık ve gürültüdür [14]. Görüntü çevirme, x ekseni veya y ekseni boyunca görüntünün çevrilmesiyle yeni bir görüntü elde eder. Gürültü ekleme işlemi, orijinal görüntüden farklı güçlendirilmiş bir görüntü elde etmek için orijinal görüntünün her pikseline rastgele ek bilgiler eklenerek görüntünün gürültü artırmasını amaçlar. Bir görüntünün parlaklık değişimi, görüntü örneğinin her pikseli üzerinde doğrudan bir doğrusal dönüşüm işlemidir. Görüntü parlaklığı dönüşüm faktörü olarak  $\lambda$  kullanılarak, parlaklık değişimiyle genişletilen görüntü örneği,  $I=\lambda I$  olarak ifade edilebilir. Burada  $0 < \lambda < 1$  koyulaştırmayı ve  $\lambda > 1$  parlaklaştırmayı temsil eder. Veri setinde 810 görüntü eğitim, 90 görüntü değerlendirme ve 39 görüntüde test için ayrılmıştır. Veri setinde toplam 8 çeşit ray kusuru bulunmaktadır. Bunlar:

- Çizik, ray kafasının yan düzlemlerinde küçük/hafif çizik,
- Ezik, demiryolunun yanal düzlemlerinin büyük/şiddetli aşınması,
- Kir, rayın yüzeyini kaplayan boya veya çamur,
- Boşluk, bir demiryolu hattında birbirini takip eden raylar arasında kalan boşluklardır,
- Tanınmamış, tanımlanamayan kusurlar,
- Hasarlı, zarar görmüş ray yüzeyi,
- Eğik, yatay pozisyonda aşağıya doğru eğimli kusurlar,
- Göçük, ray yüzeyindeki oyuklar. Yatay veya uzunlamasına değil daha küçük oval kusurlardır.

Şekil 2'de veri setinden örnek kusurlar görülmektedir.

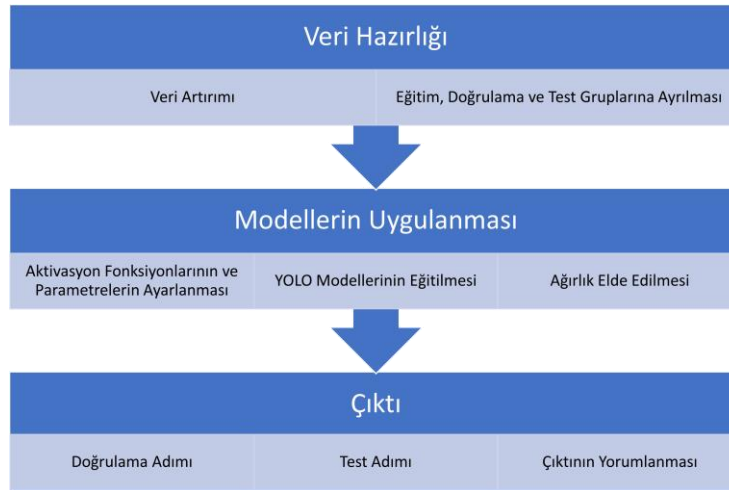


Şekil 2. Veri setinden örnekler

Eğitim aşaması tüm modellerde Google Colab aracılığıyla yapılmıştır. Deney ortamı Tesla P100-PCIE-16GB grafik işlemci birimi (GPU) ve Intel(R) Xeon(R) merkezi işlem birimi (CPU) @ 2.20GHz CPU sisteminden oluşmaktadır. Etiketlenmiş veri seti Roboflow uygulamasına yüklenmiştir. Var olan 399 görüntüye görüntü çevirme, parlaklık ve gürültü ekleme işlemleri uygulandıktan sonra 939 adet görüntü Google Colab'a aktarılmıştır. Daha sonra adım adım YOLO algoritmalarının kodları çalıştırılarak eğitim, doğrulama ve test işlemleri gerçekleştirilmiştir. Tüm modellerde kullanılan eğitim parametreleri ayarları:

- Girdi görüntü boyutu: 416
- Döngü sayısı: 200
- Bir iterasyon içerisinde her aşamada incelenen görüntü sayısı: 32

Önerilen modelin akış diyagramı Şekil 3'teki gibidir.



Şekil 3. Önerilen modelin akış diyagramı

## 2.1. YOLO

YOLO (Sadece Bir Kez Bak), sınırlayıcı kutuları ve sınıf olasılıklarını tahmin etmek için tek bir sinir ağı kullanan, GoogLeNet'ten ilham alan bir derin evrimsel sinir ağıdır. Bu, görüntünün sadece bir kez ağ içinden geçip algılama görevini tamamladığı anlamına gelir. Literatürdeki en

hızlı genel amaçlı nesne algılama mimarisidir. Ayrıca, rapor edilen ilk gerçek zamanlı evrişimsel sinir ağı tabanlı nesne algılama modelidir. Tüm algılama hattı tek bir ağda olduğundan doğrudan algılama performansı diğer sinir ağı mimarilerinin çoğundan yüksektir [15]. Şekil 4'te YOLO'nun tespit sistemi görülmektedir.

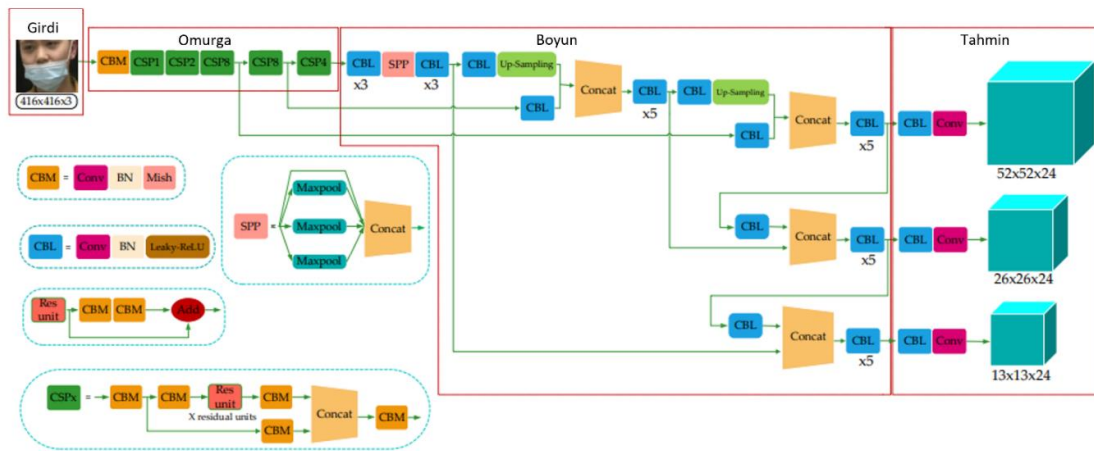


Şekil 4. Nesne tespitinden sorumlu YOLO tespit sistemi [16]

YOLO mimarisi, resim sınıflandırması için GoogLeNet modelinden esinlenmiştir. YOLO sinir ağı modeli 24 evrişim katmanına ve ardından 2 tamamen bağlı katmana sahiptir. Ağın, başlangıçtaki evrişim katmanları görüntünün özelliklerini çıkartırken, tamamen bağlantılı katmanlar çıkış olasılıklarını ve koordinatları öngörürler. YOLO, C ve CUDA ile yazılmış açık kaynaklı bir sinir ağı uygulama çatısı olan "Darknet" üzerinde uygulanmaktadır. YOLO bu şekilde uygulanarak, gerçek zamanlı video akışını işlemek için GPU işlem gücünden yararlanmaktadır.

## 2.2. YOLOv4

YOLOv4, Nisan 2020'de yayınlanan ve COCO veri setinde üzerinde iyi bir performansla ulaşan gerçek zamanlı bir nesne algılama modelidir. YOLOv4 modeli YOLOv3'e dayalı optimize edilmiş bir modeldir. Şekil 5'te görüldüğü üzere üç kısımdan oluşmaktadır; omurga, boyun ve tahmin. Omurga kısmında CSPDarknet53 ağı kullanılmıştır. Omurga kısmında ayrıca BoF (Bag of Freebies) ve BoS (Bag of Specials) adlı iki bölüm daha bulunmaktadır. Bu bölümler veri setini zenginleştirmek için kullanılmaktadır [17]. Boyun, nesnelere farklı ölçeklerde algılamak için kullanılır. Kafa bölümünü besleyen bilgileri zenginleştirmek için aşağıdan-yukarı, yukarıdan-aşağı akıştan gelen özellik haritaları kafa bölümünü beslemeden önce eleman bazında veya birleştirilerek birbirine eklenir [17]. Boyun yapısı Uzamsal Piramit Havuzu (SPP) ve Yol Toplama Ağı (PANet) yöntemlerinden oluşmaktadır. Kafa bölümü ise tahmin katmanıdır.

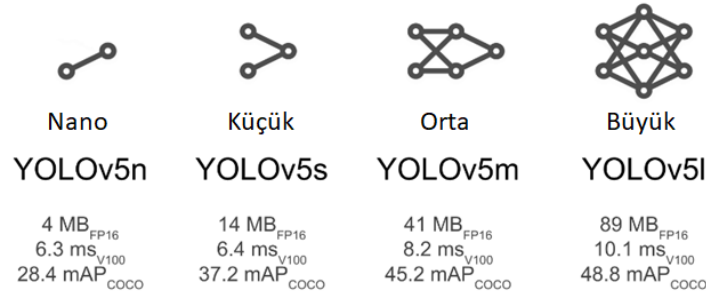


Şekil 5. YOLOv4 mimarisi [18]

### 2.3. YOLOv5

YOLOv4'ün yayınlanmasından kısa bir süre sonra Glenn ve ekibi tarafından mimarinin yeni sürümü olan YOLOv5 yayınlanmıştır. Mimarinin önceki sürümlerine göre işlem süresini önemli ölçüde düşürmesiyle birlikte YOLOv5, Pytorch'ta yeni bir eğitim ortamı altında derlenerek eğitim sürecini Darknet'ten daha kolay bir hale getirmiştir. Bu ağ modelinin algılama doğruluğu ve hızı önceki sürümlere kıyasla oldukça yüksektir. Buna ek olarak, YOLOv5'in ağ modelinin ağırlık dosyasının boyutları da önceki sürüm olan YOLOv4'ten yaklaşık %90 oranla daha küçüktür.

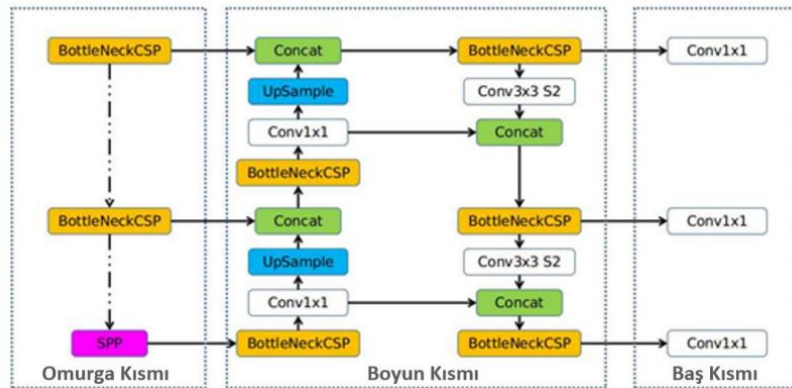
Bütün bu avantajlar YOLOv5 modelinin gerçek zamanlı tespit uygulamaları için gömülü cihazlar üzerinde çalışması için uygun olduğunu göstermektedir. YOLOv5 mimarisi boyut ve model parametrelerinin miktarına göre artan dört alt mimariyi içermektedir. Bunlar; YOLOv5s, YOLOv5n, YOLOv5m ve YOLOv5l mimarileridir. Bu mimariler sinir ağının belirli bir yerindeki özellik çıkarma modüllerinin ve evrişim çekirdek miktarlarının farklı olmasıyla birbirinden ayrılırlar. Model boyutları ve parametre sayısı da sırayla artmaktadır. Bu artış Şekil 6'da görülmektedir.



Şekil 6. YOLOv5 mimarisinin alt mimarileri [19]

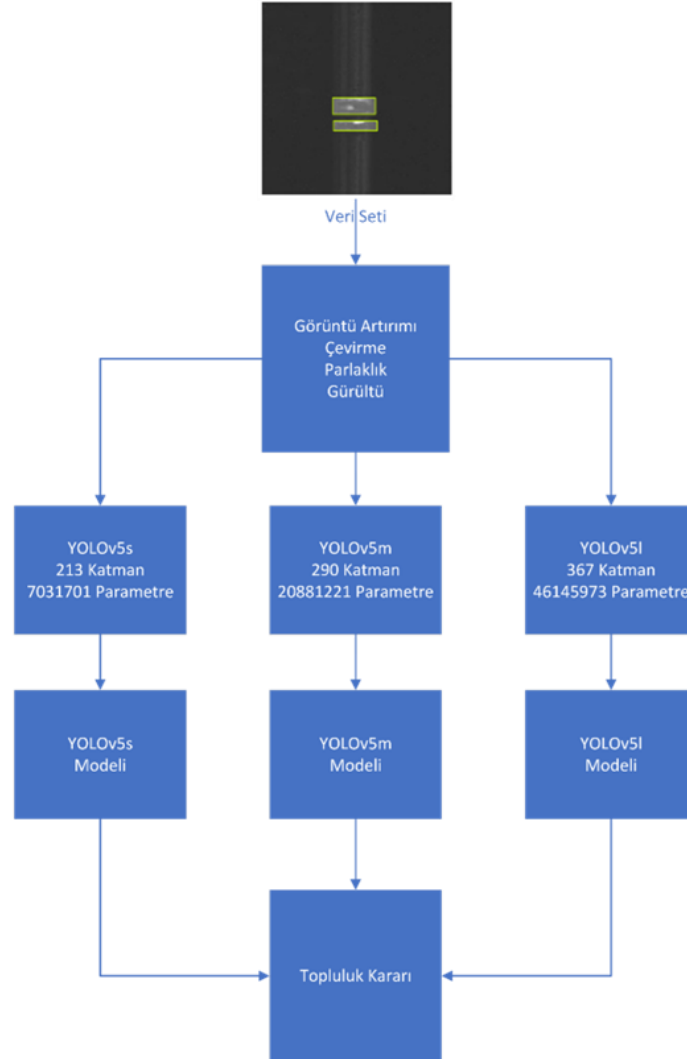
Bu çalışma kapsamında geliştirilen sürücü asistan sistemi uygulamaları gömülü platform üzerinde gerçek zamanlı olarak çalışacağından ve sistemin minimum kaynak kullanımı ile maksimum performansa ulaşması istendiğinden ağı eğitimi YOLOv5s [20] ile gerçekleştirilmiştir. Genel yapısı Şekil 7'de verilen YOLOv5 ağ mimarisi omurga, boyun ve baş kısmı olmak üzere üç bölüme ayrılmıştır.

Omurga kısmında CSPDarknet [20], boyun kısmında PANet [20] ve baş kısmında ise YOLO katmanı kullanılmaktadır. Özetle, veriler özellik çıkarımı için ilk olarak CSPDarknet'den geçirilerek özellik birleştirme için PANet'e verilir. Bunun sonucunda da YOLO katmanı sınıf, skor, konum ve boyut bilgisi gibi çıktıları vermektedir.



Şekil 7. YOLOv5 mimarisinin genel yapısı [21]

Bu çalışmada ayrıca YOLOv5 ile eğitilmiş ağırlıkları kullanarak topluluk modellemesi yöntemi kullanılmıştır. Topluluk modelleme işleminin mAP.5 ve geri çağırma değerleri üzerinde pozitif etki yaptığı saptanmıştır. Topluluk modellemesi birçok farklı modelleme algoritması kullanarak ya da farklı eğitim veri setleri kullanarak bir sonucu tahmin etmek için çok çeşitli modellerin oluşturulduğu bir süreçtir. Topluluk modeli her bir temel modelin tahminini toplar ve görünmeyen veriler için nihai bir tahminle sonuçlanır. Topluluk modellerini kullanma amacı, tahminin genelleme hatasını azaltmaktır. Temel modeller çeşitli ve bağımsız olduğu sürece, topluluk yaklaşımı kullanıldığında modelin tahmin hatası azalmaktadır. Şekil 8’de önerilen topluluk modeli görülmektedir. Önerilen yapıda bireysel olarak YOLOv5 alt mimarilerin eğitimi gerçekleştirilmiştir. Topluluk öğrenimi, tüm sonuçları tahmin etmek ve entegre etmek için birden fazla model kullanır. Topluluk modelinde görüntüler, yalnızca bir modelin sınırlandırılmasını önlemek için farklı modellerden elde edilen sonuçları birleştirebilen ray üzerindeki kusurları algılamak için çeşitli parametre ayarları ile eğitilmiş birden çok modele gönderilir. Ardından ardışık düzen sınırlayıcı kutunun tahminlerinin ortalamasını alır ve tüm modellerin kategori tahminlerini oylar. Topluluk modeli, model içinde birden fazla temel modele sahip olsa da tek bir model olarak hareket eder ve çalışır [22]. Çalışmada YOLOv5’in alt mimarileri olan YOLOv5s, YOLOv5n, YOLOv5m, YOLOv5l kullanılarak veri seti eğitilmiştir. Eğitim sonucu oluşan ağırlıklar kullanılarak topluluk modellemesi yapılmıştır.



Şekil 8. Önerilen topluluk modeli

## 2.4. YOLOv6

YOLOv6 modeli Haziran 2022'de Meituan tarafından piyasaya sürülmüştür. COCO veri kümesi karşılaştırmasında iyi sonuçlar vermiştir. YOLOv6, modeli YOLO mimarisi temelinde inşa edilmiş ve YOLO ailesinin diğer modellerine göre çeşitli iyileştirmeler ve yeni yöntemler sunmuştur. YOLOv6, PyTorch'ta yazılmıştır. YOLOv6'nın üç önemli güncellemesi var: donanım dostu omurga, boyun tasarımı ve verimlilik için ayrılmış baş ve etkili eğitim stratejileri. YOLOv6'nın nesne algılama performansının, algoritmanın aşamalı sürümleriyle hem hız hem de doğruluk iyileştirmeleri ile diğer CNN tabanlı algoritmalarla karşılaştırılabilir olduğu gösterilmiştir [23]. Çalışmada farklı görüntü boyutlarının YOLOv6 alt mimarilerinin performanslarına etkisi gösterilmiştir. Çalışmada YOLOv6 modeli için kullanılan veri seti 416x416 ve 640x640 boyutlarında iki versiyona türetilmiştir.

## 2.5. Metrikler

Model eğitimi tamamlandıktan sonra modelin test edilmesi için eğitilen ağırlıklar kullanılır ve model birçok yönden değerlendirilir. Ray kusurlarını içeren veri setimiz için test sonuçları üç kategoride sınıflandırılabilir: TP (gerçek pozitif), test setindeki kategorilerin test sonuçlarıyla aynı olduğu anlamına gelir; FP (yanlış pozitif), tespit edilen nesne kategorisindeki örneklerin sayısının gerçek nesne kategorisiyle tutarsız olduğu anlamına gelir ve FN (yanlış negatif), gerçek numunenin ters sonuç olarak veya tespit edilmeyen kategoride tespit edildiğini gösterir. Model tarafından değerlendirilen tüm pozitif vakalar için sayı (TP + FP), bu nedenle gerçek vakaların (TP) oranına kesinlik oranı denir ve bu, pozitif vakalardaki gerçek vaka örneklerinin, tarafından tespit edilen numuneler arasındaki oranını temsil eder. Denklem 1'de gösterilmiştir.

$$Kesinlik = \frac{TP}{TP + FP} \quad (1)$$

Test setindeki tüm pozitif örnekler için sayı (TP + FN)'dir. Bu nedenle, Denklem 2'de gösterildiği gibi, geri çağırma oranı, modelin test setindeki gerçek durumları tespit etme yeteneğini ölçmek için kullanılır [24].

$$Geri \text{ Çağırma} = \frac{TP}{TP + FN} \quad (2)$$

Modelin kesinliğini karakterize etmek için bu makale, Denklem 3 ve Denklem 4'te gösterildiği gibi modelin doğruluğunu değerlendirmek için AP (ortalama kesinlik) ve mAP (ortalama kesinlik değerlerinin ortalaması) göstergelerini tanıtmaktadır [24]. Denklem 3 ve 4'te bulunan P, R, N değerleri sırasıyla tüm kategorilerdeki kesinliği, geri çağırma oranını ve toplam nesne sayısını temsil etmektedir [24].

$$AP = \int_0^1 P(R) dR \quad (3)$$

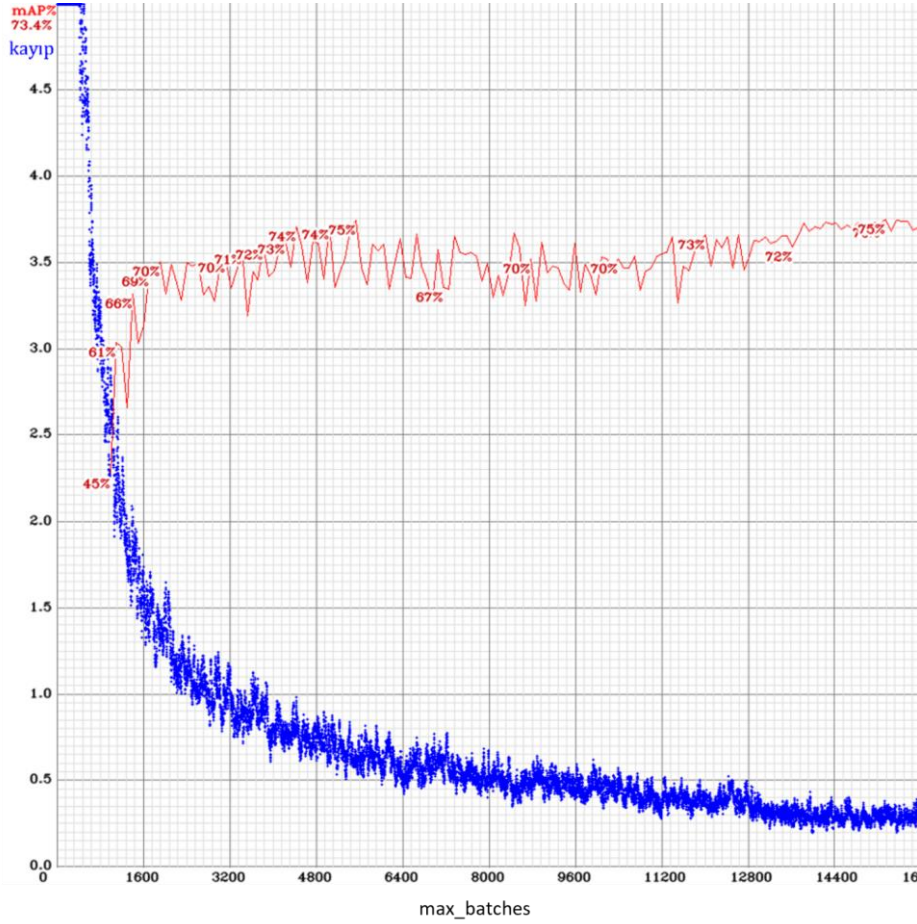
$$mAP = \frac{\sum_{i=1}^N AP_i}{N} \quad (4)$$



### 3. Deneysel Sonuçlar

#### 3.1. YOLOv4 sonuçları

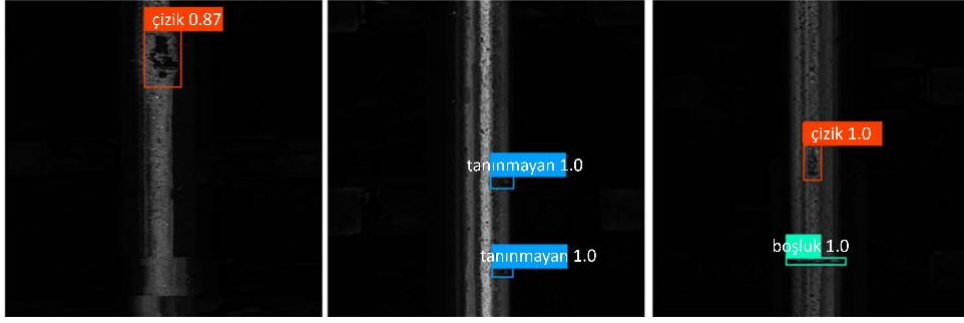
Çalışmada eğitimini gerçekleştirdiğimiz YOLOv4 modeli 0,7338 mAP.5 değeri yakalamıştır. Ayrıca 0,73 kesinlik, 0,69 geri çağırma oranı elde etmiştir. YOLOv4 modelinin kayıp grafiği Şekil 9'da görülmektedir.



Şekil 9. YOLOv4 modelinin kayıp grafiği

Şekil 9'da görüldüğü gibi max\_batch sayısı arttıkça kayıp kademeli olarak azalmıştır. Toplam iterasyon sayısı max\_batch ile belirtilir. Toplam iterasyon yeterli sayılara ulaştığında kayıp ve mAP üzerinde büyük etkisi olmuştur. Yalnızca toplam iterasyon sayısı çok küçük olduğunda düşük eğitim kaybı ve doğrulama mAP oranları veriyor gibi görünmektedir.

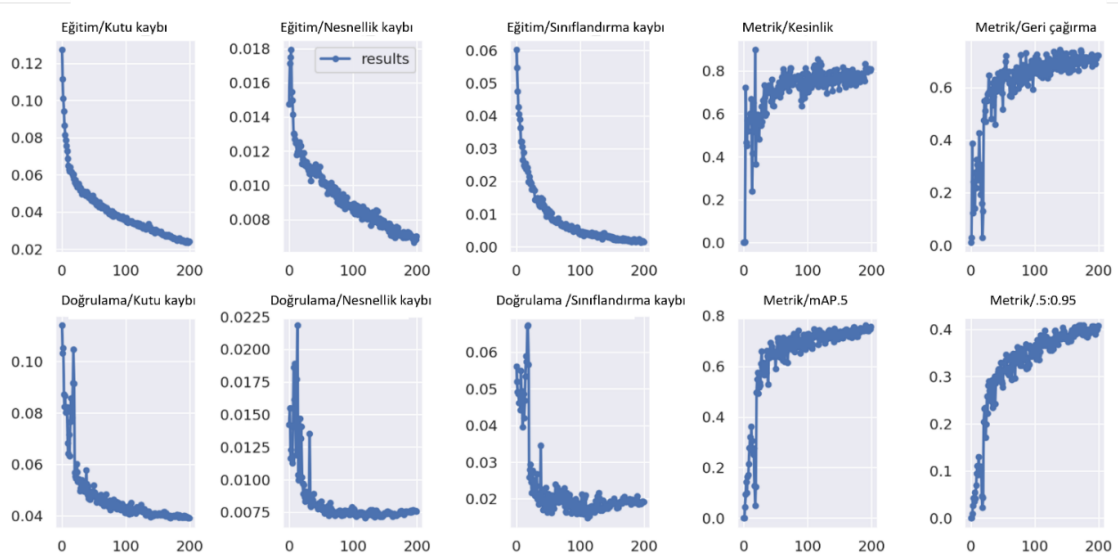
Eğitim sonucunda elde edilen ağırlık ile test edilen ray kusuru içeren görüntüleri görsel ortama aktardığımızda Şekil 10'daki sonuçlar ve mAP.5 oranları elde edilmiştir.



Şekil 10. YOLOv4 modelinin test sonuçları

### 3.2. YOLOv5 sonuçları

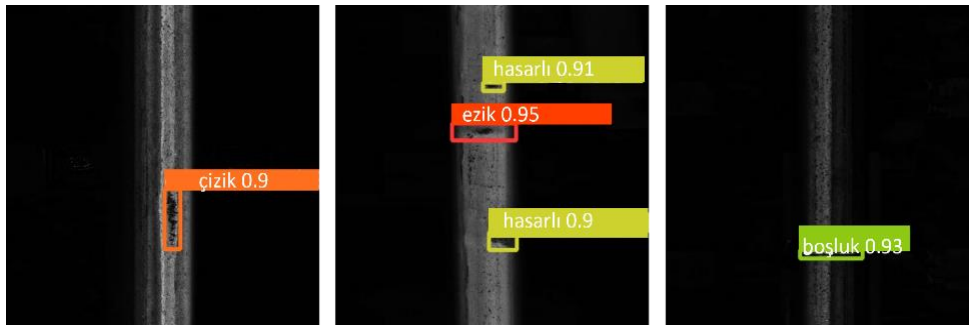
Çalışmada YOLOv5 modelinin alt mimarileri nesne tespiti için kullanılmıştır. Bunlar; YOLOv5s, YOLOv5n, YOLOv5m, YOLOv5l'dir. Şekil 11'de YOLOv5s alt mimarisine ait eğitim ve doğrulama grafikleri görülmektedir.



Şekil 11. YOLOv5s mimarisinin eğitim ve doğrulama sonuçları

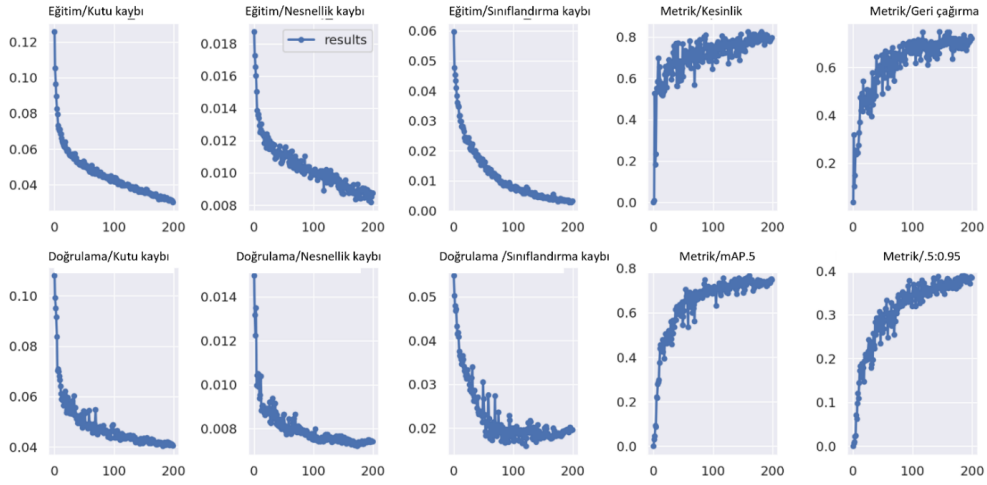
YOLOv5s alt mimarisi eğitimi sonucunda 0,7555 mAP.5 değeri yakalamıştır. Ayrıca 0,744 kesinlik, 0,742 geri çağırma oranı elde etmiştir.

Eğitim sonucunda elde edilen ağırlık ile test edilen ray kusuru içeren görüntüleri görsel ortama aktardığımızda Şekil 12'deki görüntüler elde edilmiştir.



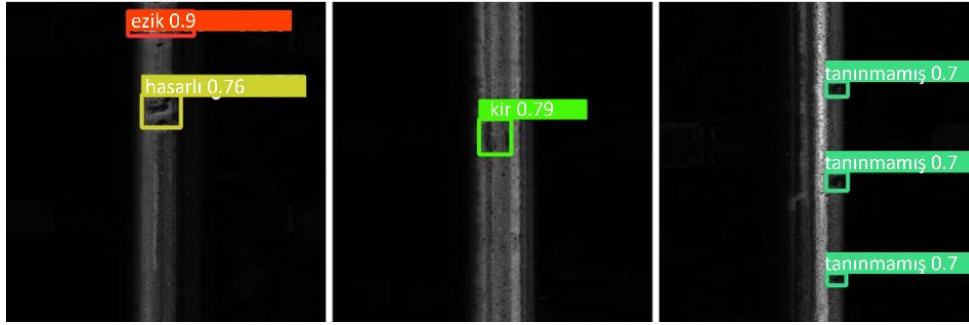
Şekil 12. YOLOv5s alt mimarisinin test sonuçları

Şekil 13'te YOLOv5n alt mimarisine ait eğitim ve doğrulama grafikleri görülmektedir.



Şekil 13. YOLOv5n alt mimarisinin eğitim ve doğrulama sonuçları

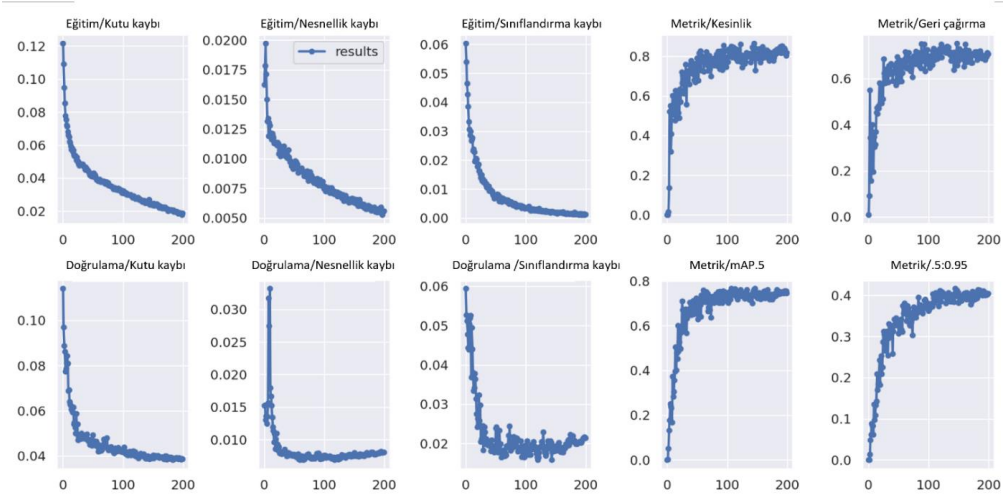
YOLOv5n alt mimarisinin eğitimi sonucunda elde edilen ağırlık ile test edilen ray kusuru içeren görüntüleri görsel ortama aktardığımızda Şekil 14'teki görüntüler elde edilmiştir.



Şekil 14. YOLOv5n alt mimarisinin test sonuçları

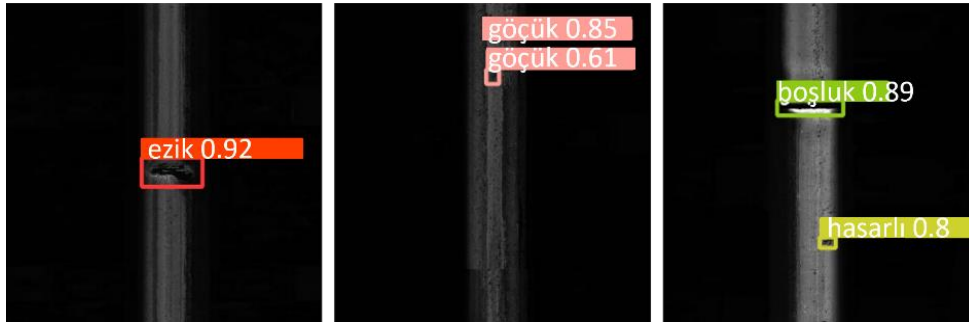
YOLOv5n alt mimarisi eğitimi sonucunda 0,755 mAP.5 değeri yakalamıştır. Ayrıca 0,794 kesinlik, 0,704 geri çağırma oranı elde etmiştir.

Şekil 15'te YOLOv5m alt mimarisine ait eğitim ve doğrulama grafikleri görülmektedir.



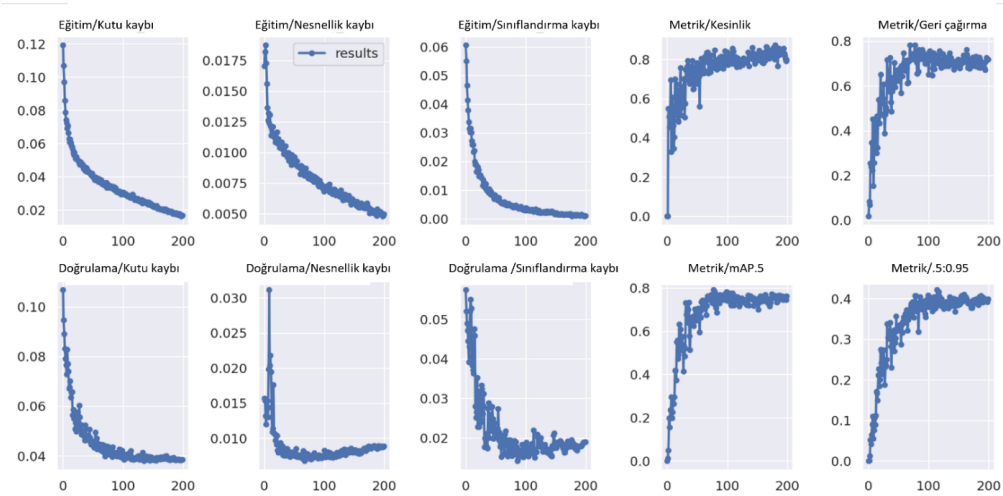
Şekil 15. YOLOv5m mimarisinin eğitim ve doğrulama sonuçları

YOLOv5m alt mimarisi eğitimi sonucunda 0,752 mAP.5 değeri yakalamıştır. Ayrıca 0,812 kesinlik, 0,68 geri çağırma oranı elde etmiştir. YOLOv5m alt mimarisinin eğitimi sonucunda elde edilen ağırlık ile test edilen ray kusuru içeren görüntüleri görsel ortama aktardığımızda Şekil 16'daki görüntüler elde edilmiştir.



Şekil 16. YOLOv5m alt mimarisinin test sonuçları

Şekil 17'de YOLOv5l alt mimarisine ait eğitim ve doğrulama grafikleri görülmektedir.



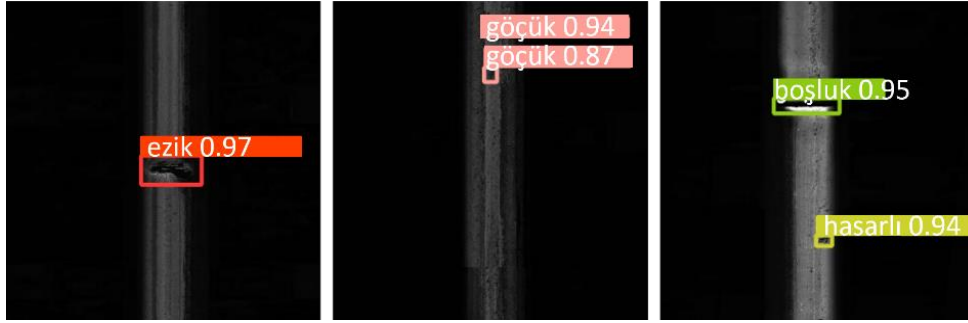
Şekil 17. YOLOv5l mimarisinin eğitim ve doğrulama sonuçları

YOLOv5l alt mimarisi eğitimi sonucunda 0,759 mAP.5 değeri yakalamıştır. Ayrıca 0,813 kesinlik, 0,723 geri çağırma oranı elde etmiştir. Tablo 1’de YOLOv4 ve YOLOv5 modellerinin eğitimi sonucu sınıf bazında elde edilen mAP.5 değerleri görülmektedir.

**Tablo 1.** YOLOv5m Alt Mimarisini Eğitim Sonucunda Sınıf Bazında Elde Ettiği mAP.5 Değerleri

Kusur Çeşidi	YOLOv4 mAP.5 Değeri	YOLOv5s mAP.5 Değeri	YOLOv5n mAP.5 Değeri	YOLOv5m mAP.5 Değeri	YOLOv5l mAP.5 Değeri
Ezik	0,930	0,912	0,869	0,969	0,922
Göçük	0,650	0,688	0,797	0,722	0,794
Çizik	0,770	0,832	0,786	0,829	0,800
Eğik	0,680	0,728	0,842	0,681	0,730
Hasarlı	0,380	0,349	0,253	0,325	0,327
Kir	0,600	0,712	0,718	0,712	0,719
Boşluk	1,000	0,995	0,977	0,990	0,979
Tanınmamış	0,830	0,821	0,799	0,786	0,805

YOLOv5l alt mimarisinin eğitimi sonucunda elde edilen ağırlık ile test edilen ray kusuru içeren görüntüleri görsel ortama aktardığımızda Şekil 18’deki görüntüler elde edilmiştir.



**Şekil 18.** YOLOv5l alt mimarisinin test sonuçları

Tablo 2’de YOLOv5 alt mimarilerinin doğrulama performansları ve topluluk modelleme ile elde edilen doğrulama performansları görülmektedir. Topluluk modelleme, eğitim sonucu elde edilen ağırlıkların doğrulama performanslarının mAP.5 değerlerini yukarıya çektiği görülmektedir.

**Tablo 2.** YOLOv5 Modelleri ile Topluluk Modelleme Yönteminin Doğrulama Performansları

Model	P	R	mAP.5
YOLOv5n	0.794	0.704	0.756
YOLOv5s	0.744	0.742	0.753
YOLOv5m	0.812	0.68	0.752
YOLOv5l	0.875	0.716	0.783
YOLOv5n+ YOLOv5s+ YOLOv5m+ YOLOv5l	0.843	0.763	0.809
YOLOv5s+ YOLOv5m+ YOLOv5l	0.849	0.751	0.811

### 3.3. YOLOv6 sonuçları

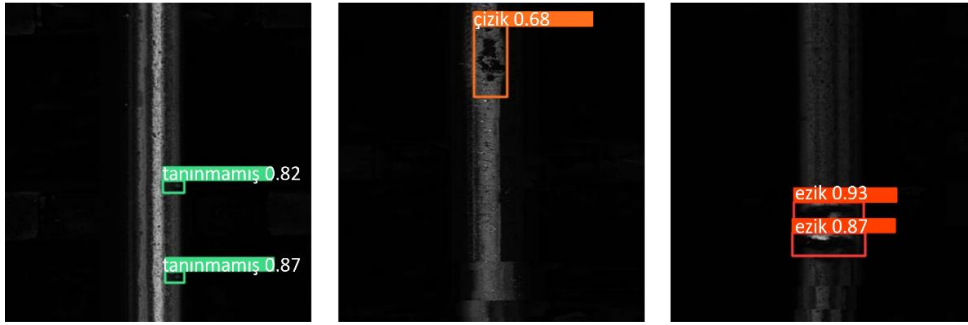
Çalışmada YOLOv6 modelinin alt mimarileri olan YOLOv6s, YOLOv6n ve YOLOv6t veri setini eğitmek için kullanılmıştır. Kullanılan alt YOLOv6 alt mimarilerinin eğitim sonuçları Tablo 3’te görülmektedir. Tablo 3’te görüldüğü gibi artan girdi boyutu ray üzerindeki kusurların tespiti konusunda başarıyı arttırmıştır. mAP.5 değerindeki artış bunun ispatıdır. Şekil 19-24’te YOLOv6 alt mimarilerinin eğitim sonucu oluşturulan ağırlıkları kullanılarak elde edilen test sonuçları görülmektedir.

Tablo 3. YOLOv6 Alt Mimarisinin Eğitim Sonuçları

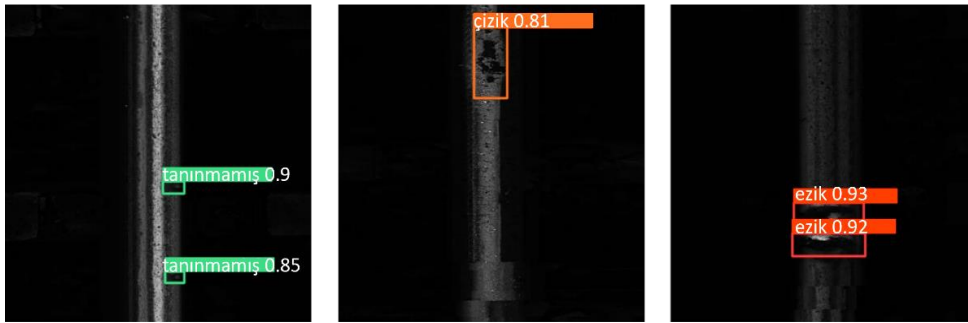
Model	Girdi Boyutu	mAP.5
YOLOv6s	416	0.7074
YOLOv6n	416	0.7063
YOLOv6t	416	0.6858
YOLOv6s	640	0.7802
YOLOv6n	640	0.7473
YOLOv6t	640	0.7792



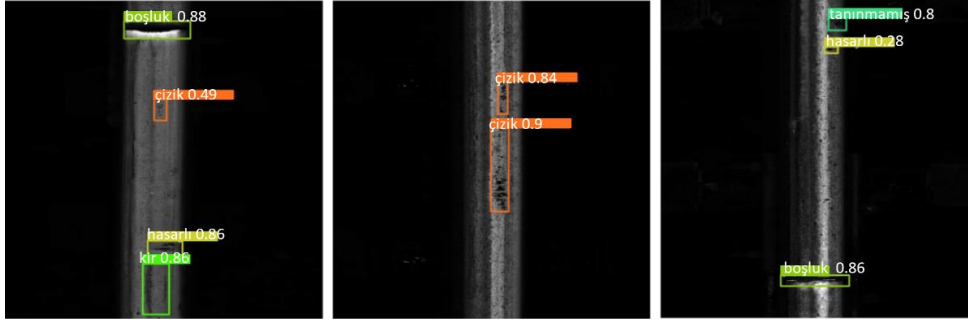
Şekil 19. 416x416 girdi boyutlu YOLOv6s alt mimarisinin test sonuçları



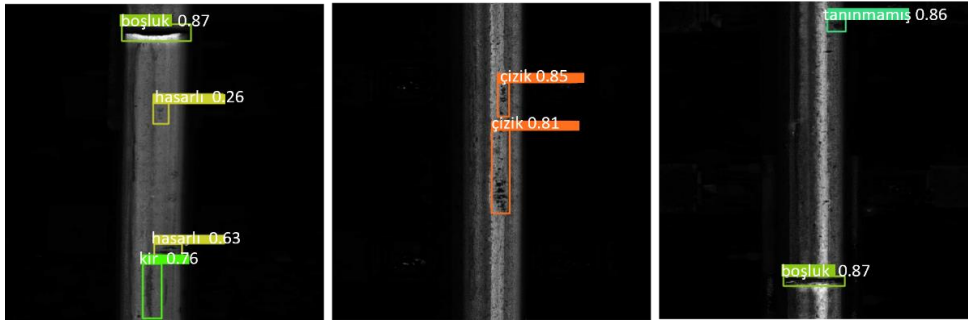
Şekil 20. 416x416 girdi boyutlu YOLOv6n alt mimarisinin test sonuçları



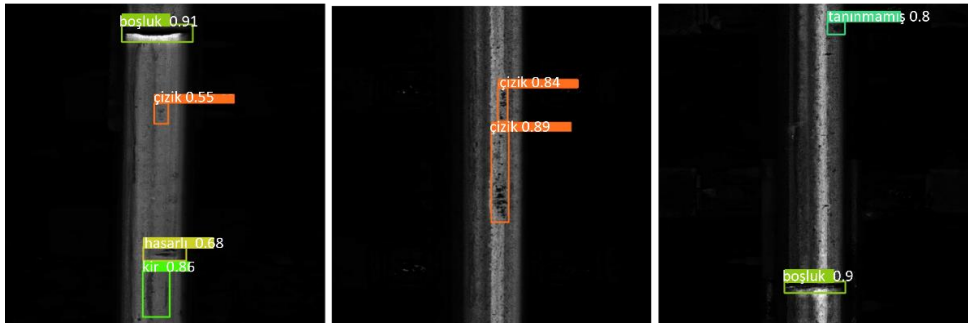
Şekil 21. 416x416 girdi boyutlu YOLOv6t alt mimarisinin test sonuçları



Şekil 22. 640x640 girdi boyutlu YOLOv6s alt mimarisinin test sonuçları



Şekil 23. 640x640 girdi boyutlu YOLOv6n alt mimarisinin test sonuçları



Şekil 24. 640x640 girdi boyutlu YOLOv6t alt mimarisinin test sonuçları

YOLOv4 modeli eğitim sonucunda 0,7338 mAP.5'lik oranı yakalamıştır. YOLOv5'in alt mimarileri YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l sırasıyla 0,7555, 0,755, 0,752, 0,752 mAP.5 oranı yakalamışlardır. YOLOv6'nın alt mimarisi olan YOLOv6s modeli girdi veri boyutunu attığımızda Tablo 3'te görüldüğü gibi 0.7802'lik mAP.5 oranı yakalamıştır. Çalışmada görüldüğü gibi YOLOv5 ve YOLOv6 modelleri YOLOv4'e göre daha başarılı sonuçlar vermişlerdir. İki modelin doğrulama performansları Tablo 4'teki gibidir. Doğrulama işlemi 90 adet görüntüyle yapılmıştır.

Tablo 4. YOLOv5 Modelleri ile Topluluk Modelleme Yönteminin Doğrulama Performansları

Model	Görüntü Sayısı	Girdi Boyutu	mAP.5
YOLOv5n	939	416	0.756
YOLOv5s	939	416	0.753
YOLOv5m	939	416	0.752
YOLOv5l	939	416	0.783
YOLOv5n+ YOLOv5s+ YOLOv5m+ YOLOv5l	939	416	0.809

YOLOv5s+ YOLOv5m+ YOLOv5l	939	416	0.811
YOLOv6s	939	416	0.734
YOLOv6n	939	416	0.702
YOLOv6t	939	416	0.704
YOLOv6s	939	640	0.770
YOLOv6n	939	640	0.738
YOLOv6t	939	640	0.785

Tablo 4'te görüldüğü gibi topluluk modelleme YOLOv5 alt mimarilerine ait doğrulama performanslarını arttırmıştır. YOLOv6 modelinde ise girdi veri boyutunu attırmak da doğrulama performansına pozitif etki yapmıştır. Buradaki dezavantaj ise eğitim süresinin artması olmuştur. Ortalama eğitim süresi 1,8 saatten 3,2 saate çıkmıştır. Çalışmanın bir diğer katkısı da literatürde demiryolu alandaki çalışmalarda YOLOv5 topluluk modelleme ve YOLOv6 yöntemleri fazla kullanılmamıştır. Tablo 5'te görüldüğü gibi önerilen topluluk modelleme yöntemi literatürdeki bazı yaklaşımlarla karşılaştırıldığında daha yüksek kesinlik ve geri çağırma değerleri elde etmiştir.

**Tablo 5.** YOLOv5 Topluluk Modelleme Yöntemi ile Literatürdeki Yaklaşımların Performans

Referans	Yaklaşım	Karşılaştırması		
		Görüntü Sayısı	P	R
[25]	BSM	195	0.278	0.726
[25]	CTFM	195	0.841	0.732
[25]	REM	195	0.039	0.472
[26]	Yolov4	278	0.880	0.666
[26]	ASFF-Yolov4	278	0.725	0.439
Önerilen Metot-I	YOLOv5n+ YOLOv5s+ YOLOv5m+ YOLOv5l	939	0.843	0.763
Önerilen Metot-II	YOLOv5s+ YOLOv5m+ YOLOv5l	939	0.849	0.751

#### 4. Sonuç

Bu çalışmada, demiryolu hattından toplanan görüntülerden demiryolu rayı üzerindeki kusurların tespiti için YOLOv4, YOLOv5 ve YOLOv6 tabanlı bir yaklaşım önerilmektedir. 8 farklı ray kusuru içeren ve 399 görüntüden oluşan veri seti görüntü artırım işlemine tabi tutulmuştur. Bu görüntü artırım teknikleri görüntü çevirme, parlaklık ve gürültü ekleme işlemidir. Gürültü ekleme işlemi daha zor şartlarda modelin performansını ölçmeyi sağlamıştır. Eğitim işleminde 810 adet ray kusuru içeren görüntü kullanarak eğitim işlemi tamamlanmıştır. Doğrulama işlemi için 90 adet görüntü, test işlemi için 39 adet görüntü kullanılmıştır. Yüzde olarak belirtecek olursak eğitim için %86, doğrulama için %10 ve test içinde %4'lük veri seti ayrılmıştır. Topluluk modelleme yöntemiyle 8 adet farklı ray kusuru tespit eden YOLOv5 modeli %81 mAP.5, %84 kesinlik ve %75 geri çağırma oranı yakalamıştır. Ölçüm sonuçları önerilen modellerin rayların içerdikleri kusurları tespit etmede topluluk modelleme ve girdi verinin boyutunu arttırmanın başarımlarına etki ettiğini göstermiştir.

#### Teşekkür

Bu çalışma, 120E097 numaralı TÜBİTAK projesi tarafından desteklenmiştir.

#### Kaynaklar

- [1] O. Yaman, Demiryolu Rayları İçin Gerçek Zamanlı Bulanık Otomata ile Görme Tabanlı Arıza Teşhis Sisteminin Geliştirilmesi, PhD Thesis, Fırat University, 2018.



- [2] C. Taştımur, M. Karaköse, E. Akın and İ. Aydın, "Rail defect detection with real time image processing technique," 2016 IEEE 14th International Conference on Industrial Informatics (INDIN), 2016, pp. 411-415, doi: 10.1109/INDIN.2016.7819194.
- [3] D. Barke, & W. K. Chiu, "Structural Health Monitoring in the Railway Industry: A Review," *Structural Health Monitoring*, 2005, 4(1), 81–93.
- [4] D. Çetintaş, T. Tuncer, "Determining the type of document read using eye movement properties by hybrid CNN method," *Traitement du Signal*, vol. 39, No. 4, pp. 1099-1108, 2022, doi: 10.18280/ts.390402.
- [5] H. Feng, Z. Jiang, F. Xie, P. Yang, J. Shi and L. Chen, "Automatic Fastener Classification and Defect Detection in Vision-Based Railway Inspection Systems," in *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 4, pp. 877-888, 2014.
- [6] F. Guo, Y. Qian, and Y. Shi, "Real-time railroad track components inspection based on the improved yolov4 framework," *Automation in Construction*, 2021.
- [7] D. Zheng et al., "A Defect Detection Method for Rail Surface and Fasteners Based on Deep Convolutional Neural Network," *Computational Intelligence and Neuroscience*, vol. 2021, p. e2565500, Aug. 2021, doi: 10.1155/2021/2565500.
- [8] Y. Xia, F. Xie and Z. Jiang, "Broken Railway Fastener Detection Based on Adaboost Algorithm," 2010 International Conference on Optoelectronics and Image Processing, 2010, pp. 313-316, doi: 10.1109/ICOIP.2010.303.
- [9] H. Fan, P. C. Cosman, Y. Hou and B. Li, "High-Speed Railway Fastener Detection Based on a Line Local Binary Pattern," in *IEEE Signal Processing Letters*, vol. 25, no. 6, pp. 788-792, June 2018, doi: 10.1109/LSP.2018.2825947.
- [10] L. Shang, Q. Yang, J. Wang, S. Li and W. Lei, "Detection of rail surface defects based on CNN image recognition and classification," 2018 20th International Conference on Advanced Communication Technology (ICACT), pp. 45-51, 2018.
- [11] H. Yu et al., "A Coarse-to-Fine Model for Rail Surface Defect Detection," in *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 3, pp. 656-666, March 2019, doi: 10.1109/TIM.2018.2853958.
- [12] S. Yanan, Z. Hui, L. Li and Z. Hang, "Rail Surface Defect Detection Method Based on YOLOv3 Deep Learning Networks," 2018 Chinese Automation Congress (CAC), 2018, pp. 1563-1568, doi: 10.1109/CAC.2018.8623082.
- [13] Li H, Wang F, Liu J, Song H, Hou Z, et al. (2022) Ensemble model for rail surface defects detection. *PLOS ONE* 17(5), doi: 10.1371/journal.pone.0268518.
- [14] Shorten, C., Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. *J Big Data* 6, 60 (2019). <https://doi.org/10.1186/s40537-019-0197-0>.
- [15] A. Akdağ, Derin öğrenme algoritmaları kullanılarak gerçek zamanlı silah tanıma uygulaması, Master Thesis, Necmettin Erbakan University, 2017.
- [16] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.
- [17] E. Güçlü, İ. Aydın, K. Şahbaz, E. Akın ve M. Karaköse, "Demiryolu Bağlantı Elemanlarında Bulunan Kusurların YOLOv4 ve Bulanık Mantık Kullanarak Tespiti", *Demiryolu Mühendisliği*, vol. 14, pp. 249-262, 2021, doi:10.47072/demiryolu.939830.
- [18] Z. Zakria, J. Deng, R. Kumar, M. S. Khokhar, J. Cai and J. Kumar, "Multiscale and Direction Target Detecting in Remote Sensing Images via Modified YOLO-v4," in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 1039-1048, 2022, doi: 10.1109/JSTARS.2022.3140776.
- [19] YOLOv5 Custom Training. <https://github.com/ultralytics/yolov5/wiki/Train-Custom-Data> (accessed August. 8, 2022).
- [20] E. Güney, Sürücü asistan sistemleri için mobil gpu tabanlı gerçek zamanlı durum analizi ve tespit uygulamaları, Master Thesis, Sakarya University, 2021.
- [21] R. Xu, H. Lin, K. Lu, L. Cao, Y. Liu, "A Forest Fire Detection System Based on Ensemble Learning," *Forests*. 2021; 12(2):217, doi: 10.3390/f12020217
- [22] V. Kotu, B. Deshpande, "Chapter 2 - Data Science Process." ScienceDirect, Morgan Kaufmann, 2019.
- [23] G. Conley, S. C. Zinn, T. Hanson, K. McDonald, N. Beck, and H. Wen, "Using a deep learning model to quantify trash accumulation for cleaner urban stormwater," *Computers, Environment and Urban Systems*, 2022.

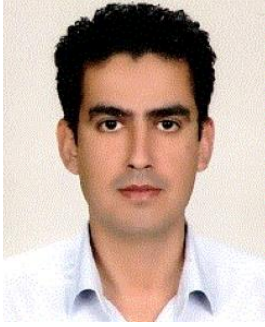
- [24] R. Padilla, S. L. Netto and E. A. B. da Silva, "A Survey on Performance Metrics for Object-Detection Algorithms," 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), 2020, pp. 237-242, doi: 10.1109/IWSSIP48289.2020.9145130.
- [25] H. Yu et al., "A Coarse-to-Fine Model for Rail Surface Defect Detection," in IEEE Transactions on Instrumentation and Measurement, vol. 68, no. 3, pp. 656-666, March 2019, doi: 10.1109/TIM.2018.2853958
- [26] C. Zhang, X. Hu, J. He, N. Hou, "Yolov4 High-Speed Train Wheelset Tread Defect Detection System Based on Multiscale Feature Fusion", Journal of Advanced Transportation, vol. 2022, Article ID 1172654, 13 pages, 2022, doi: 10.1155/2022/1172654.

### Özgeçmiş



#### Mehmet SEVİ

1988 yılında Denizli’de doğdu. Lisans derecesini Doğu Akdeniz Üniversitesi Bilgisayar Mühendisliğinde, Yüksek Lisans derecesini ise Fırat Üniversitesi Bilgisayar Mühendisliği Bölümünde aldı. Şu an Fırat Üniversitesi Bilgisayar Mühendisliği Bölümünde doktora öğrencisidir. Araştırma alanları arasında görüntü işleme ve derin öğrenme bulunmaktadır. Aynı zamanda Muş Alparslan Üniversitesi Bilgi İşlem Daire Başkanlığında mühendis olarak çalışmaktadır.  
E-Posta: m.sevi@alparslan.edu.tr



#### İlhan AYDIN

1981 yılında Elâzığ’da doğdu. Lisans ve Yüksek Lisans derecelerini Fırat Üniversitesi Bilgisayar Mühendisliğinde 2001 ve 2006 yıllarında aldı. Doktora derecesini 2011 yılında Fırat Üniversitesi, Elâzığ Elektrik ve Elektronik Mühendisliği bölümünden aldı. Halen Fırat Üniversitesi’nde Bilgisayar Mühendisliği bölümünde Profesör Doktor olarak görev yapmaktadır. Araştırma alanları arasında optimizasyon, gerçek zamanlı sistemler, hata teşhisi ve durum izleme, sinyal ve görüntü işleme ve derin öğrenme bulunmaktadır.  
E-posta: iaydin@firat.edu.tr



#### Erhan AKIN

Erhan Akın, lisans derecesini Elazığ Fırat Üniversitesi’nden elektrik mühendisliği bölümünden 1984 yılında, yüksek lisans derecesini Elazığ Fırat Üniversitesi’nden bilgisayar mühendisliği bölümünden 1987 yılında ve doktora derecesini 1994 yılında Fırat Üniversitesi elektrik mühendisliği alanından almıştır. Halen Fırat Üniversitesi Bilgisayar Mühendisliği Bölümü’nde Profesör Doktordur. Fırat Üniversitesi Bilgisayar Mühendisliği bölüm başkanıdır. Araştırma ilgi alanları arasında yumuşak bilgi işlem, elektrik motoru sürücüler, bulanık kümeler ve sistemler, doğrusal olmayan kontrol sistemleri, hata teşhisi ve bilgisayar görüşü yer almaktadır.  
E-posta: eakin@firat.edu.tr

### Beyanlar:

Bu makalede bilimsel araştırma ve yayın etiğine uyulmuştur.

Yazarların katkıları: Yazar katkıları belirtilmemiştir.