



Açıklanabilir Yapay Zekâ Tabanlı Denetimsiz Öğrenme ile Ray Kusur Tespiti

Selçuk Sinan KIRAT^{ORCID}, İlhan AYDIN^{ORCID}

Fırat Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Elazığ, Türkiye

**iaydin@firat.edu.tr*

(Alınış/Received: 10.01.2023, Kabul/Accepted: 06.02.2023, Yayımlama/Published: 31.07.2023)

Öz: Demiryolları insanı ve yükünü geçmişten günümüze kadar taşımış, artan ilgi ve talep nedeniyle gelecekte de taşımaya devam edecektir. Demiryollarında güvenli seyir için ray sağlamlığının otonom olarak tespit edilip önceden önlem alınması önem arz etmektedir. Yapay zekâ tabanlı bilgisayarlı görüş uygulamaları kapsamında derin öğrenme modelleri ile otonom kusur tespiti yapılabilmektedir. Son yıllarda açıklanabilir yapay zeka yaklaşımı kusur (anomali) tespitinde popüler olmuştur. Sistem tarafından tespit edilen kusurun, niçin kusurlu olduğunun asıl karar verici olan insana açıklanması gerekmektedir. Bu çalışmada ray yüzey kusurlarını içeren etiketsiz görüntü veri seti ile sınıflandırıcı katmanları özelleştirilmiş Vgg16 ve MobileNetV3 Small ağları eğitilmiştir. Denetimsiz öğrenme ile etiketsiz verilerden sağlam rayların özelliklerini öğrenen ağlara, test için verilen görüntülerdeki kusurlar tespit ettirilmiştir. Kusurlar açıklama haritaları ile kullanıcıya gösterilmiştir. Ağların sınıflandırma başarısında Vgg16 %98, MobileNetV3 Small %96 doğruluk seviyesine ulaşırken, kusurlu bölgenin işaretlenmesini sağlayan açıklama haritalarında Vgg16'nin daha isabetli çıkarımlar yaptığı gözlemlenmiştir.

Anahtar kelimeler: Açıklanabilir yapay zekâ, Denetimsiz öğrenme, Transfer öğrenme, Kusur tespiti, Vgg16, MobileNetV3 Small

Rail Defect Detection with Explainable Artificial Intelligence Based Unsupervised Learning

Abstract: Railways have carried people and their loads from the past to the present, and will continue to carry them in the future due to the increasing interest and demand. For safe transportation on railways, it is important to determine the rail strength autonomously and to take precautions beforehand. Within the scope of artificial intelligence-based computer vision applications, autonomous defect detection can be done with deep learning models. In recent years, the explainable artificial intelligence approach has become popular in defect (anomaly) detection. The defect that is detected by the system should be explained to the person who is the main decision maker, why it is faulty. In this study, Vgg16 and MobileNetV3 Small networks with customized classifier layers are trained with an unlabeled image dataset containing rail surface defects. With unsupervised learning, the networks that learned the properties of solid rails from unlabeled data were detected in the images given for testing. Defects are shown to the user with annotation maps. While Vgg16 reached 98% accuracy and MobileNetV3 Small 96% accuracy in the classification success of the networks, it was observed that Vgg16 made more accurate inferences in the annotation maps that marked the defective region.

Keywords: Explainable artificial intelligence, Unsupervised learning, Transfer learning, Flaw detection, Vgg16, MobileNetV3 Small

1. Giriş

Demiryollarında geçmiş günümüze, günümüz ise geleceğe taşınmaya devam etmektedir. Trenler çok uzun yıllardır insanoğlunun ulaşım ihtiyacını alternatiflerine göre daha ekonomik ve güvenli bir şekilde karşılamaktalar. Gelişen teknolojiye paralel olarak tüm ulaşım araçları gibi trenler de gelişmekte ve değişmektedir. Gün geçtikçe daha hızlı ve konforlu trenlerin hayatımızda var olması yolculuk yapmak isteyen insanları cezbetmekte, seyahat tercihinde trenleri seçmelerini

Atıf için/Cite as: S.S. Kırat, İ. Aydın, "Açıklanabilir yapay zekâ tabanlı denetimsiz öğrenme ile ray kusur tespiti," *Demiryolu Mühendisliği*, no. 18, pp. 1-13, July 2023. doi: 10.47072/demiryolu.1231751

sağlamaktadır. Turizm ve tatil anlayışı değişen insanlar tren yolculuğunu farklı bir macera, egzotik bir yolculuk, yeni yerlerin keşfi vb. birçok farklı nedenle tercih edebilmektedir [1].

İnsanı ve yükünü taşıyan trenlerin güvenle seyredebilmesi için üzerinde ilerledikleri demiryollarının da güvenli olması gerekir. Bu güvenliğin bir boyutunu da rayların sağlamlığı oluşturmaktadır. Yıl boyunca zemine sabitlenmiş olarak dış ortamdaki her türlü fiziksel hava koşullarına, trenin uyguladığı baskı ve sürtünme kuvvetine maruz kalan raylarda çeşitli kusurlar meydana gelebilmektedir. Bu durum trenin raylardan çıkarak kaza yapmasına, can ve mal kayıplarına neden olabilmektedir. Federal Railroad Administration (FRA)'ya göre 2015 – 2020 yılları arasındaki tren kazalarının %23'ü raylardaki kusurlar nedeniyle meydana gelmiştir [2]. Bu nedenle ray kusurlarının erkenden tespit edilerek önlem alınması önem arz etmektedir. Ray kusurları radyografi, sıvı penetrant, manyetik indüksiyon, ultrasonik, görme tabanlı yöntemlerle tespit edilebilmektedir. Son yıllarda bilgisayarlı görü çalışmalarının popüler olması nedeniyle ray kusurlarının tespitinde görme tabanlı yöntemler üzerinde araştırmacılar yoğunlaşmaktadır [3]. Çağımızda sıkça duyar olduğumuz “yapay zekâ” kavramı insanda heyecan uyandırmaktadır. Yapay zekâ; akıllı bir makine, akıllı bir bilgisayar programı ortaya çıkarma bilimi, mühendisliği olarak tanımlanmaktadır [4]. Yapay zekâ çalışmalarında akıllı bir sistem geliştirmek için makine öğrenmesi algoritmaları kullanılmaktadır. Derin öğrenme, makine öğrenmesi algoritmalarından birisidir. Özellikle bilgisayarlı görü uygulamalarında veriden öznelik çıkarma işini otomatik hale getiren derin öğrenme, son yıllarda popüler olmuştur.

Hayatımızda birçok teknolojik aletin artık otonom olduğunu görebiliyoruz. Evlerde kullanılan süpürgeler, çocukların oyuncakları, kara, hava ve deniz araçları otonom olarak hareket edebilmekte, insan yerine bazı görevleri başarılı bir şekilde yerine getirebilmektedir. Makinelerin otonom davranış sergilemesi için çevrelerini algılaması, makine öğrenmesi vb. yöntemler ile algıladıkları çevreden gelen verilerle öğrenmesi ve aksiyon sergilemesi gerekir [5]. Bu durum yapay zekâ kavramının işaret ettiği akıllı makinelere örnek teşkil etmektedir. Demiryollarında hareket eden trenlerin de otonom hareket edebilmesi, engel tanınması, önünü görebilmesi, çevresinden topladığı verilere göre karar verip aksiyon sergilemesi adına çalışmalar yapılmaktadır. Lokomotifler, tramvaylar üzerinde yapılan çeşitli çalışmalarda engel algılama gibi çeşitli görevler yerine getirilmektedir. Buna rağmen halen tam otonom bir sürüş yapılmamakta, insana ihtiyaç duyulmaktadır. Demiryolları ile ilgili olarak yapılan yapay zekâ çalışmaları şimdilik makinist, kondüktör, demiryolu işletmesi sorumlusu gibi görevlilerin karar vermelerine destek olmaktadır [6].

Yapay zekâ çalışmalarında sıkça kullanılan derin öğrenme modelleri, farklı mimarilerde tasarlanmış derin sinir ağlarından oluşmaktadır. Derin sinir ağları milyonlarca nöron, öğrenilebilir parametre, çok fazla derin katman içermektedir. Derin öğrenme modelleriyle eğitilen sistem öğrenmekte, yeni karşılaştığı verilere çeşitli çıktılar üretmektedir [7]. Gittikçe daha karmaşık hale gelen derin öğrenme modelleri kara kutu olarak ifade edilmekte olup, bu modellerin ürettiği çıktıların insan tarafından anlaşılabilmesi ve yorum yapılabilmesi adına “Açıklanabilir Yapay Zekâ” (Explainable Artificial Intelligence - XAI) yaklaşımı ortaya çıkmıştır [8]. Makine tarafından tahmin edilen basit bir kedi / köpek gibi sınıflandırma sonuçlarının açıklanabilir olması gerekmeyebilir. Fakat tıp ve sağlık alanında karar vericilere yardımcı olacak sistemin, bir dokunun kanserli olup olmadığı ile ilgili tahminlerinin açıklanabilir olması gerekir. Aksi halde sadece makinenin tahmin kararına güvenen bir cerrah, sağlam bir kişiyi gereksiz yere ameliyat edebilir. Benzer şekilde demiryolları gibi yüksek bütçeli ciddi işlerde sistemin ürettiği çıktının da açıklanabilir olması önemlidir. Makinenin de hata payının olduğu unutulmamalıdır. Makinenin verdiği kararlar açıklanabilir olduğunda, asıl karar verici olan insan açıklama bilgisine bakarak kararın uygulanması noktasındaki iradesini kullanabilir. Bu sayede hatalı aksiyonlar engellenebilir. Açıklanabilir yapay zekâ ile sistemin verdiği kararların alt yapısını gösteren açıklama haritaları insana gösterilir. İnsan, açıklama haritalarını incelediğinde makine tarafından alınmış muhtemel bir yanlış kararı algısal olarak fark edip yanlış bir aksiyon gösterilmesini

engelleyebilir. Makine öğrenmesinde kullanılan modellerden bazıları denetimli (supervised) ve denetimsiz (unsupervised) öğrenme başlıkları altında incelenebilir. Denetimli öğrenme olarak ifade edilen modellerde girdi olarak kullanılan verilerin etiket bilgileri uzman kişi tarafından belirlenmiştir. Denetimsiz öğrenme modellerinde uzman tarafından etiketlenmiş veri yoktur. Elde etiket bilgileri olmayan salt veriler vardır. Model bu verilerle eğitilir. Model girişte aldığı bu verilere etiket tahmini yapmaz. Aslında denetimsiz öğrenmede öğrenilen bilgi, modelin aldığı veriler içerisindeki gizli örüntülerdir [9-11]. Denetimsiz öğrenmeye özellikle etiketli veri elde etmenin zor, sınırlı olduğu ya da bulunmadığı durumlarda başvurulmaktadır. Demiryollarında binlerce kilometrelik hatlardan alınabilecek milyonlarca ray görüntüsü verilerinde sağlam ve kusurlu ray görüntülerinin tek tek uzman birisi tarafından etiketlenmesi meşakkatli bir iştir. Çoğu zaman derin öğrenme modellerinin eğitilmesi için demiryolları ile ilgili etiketlenmiş büyük veri bulunamamakta, bulunan verilerde ise kusurlu / kusursuz görüntü sınıflarına dahil veri sayılarında dengesizlikler olabilmektedir. Bu durumlarda elde olan sınırlı sayıdaki etiketsiz görüntü verisi ile denetimsiz öğrenme modellerinin kullanılması uygun olacaktır.

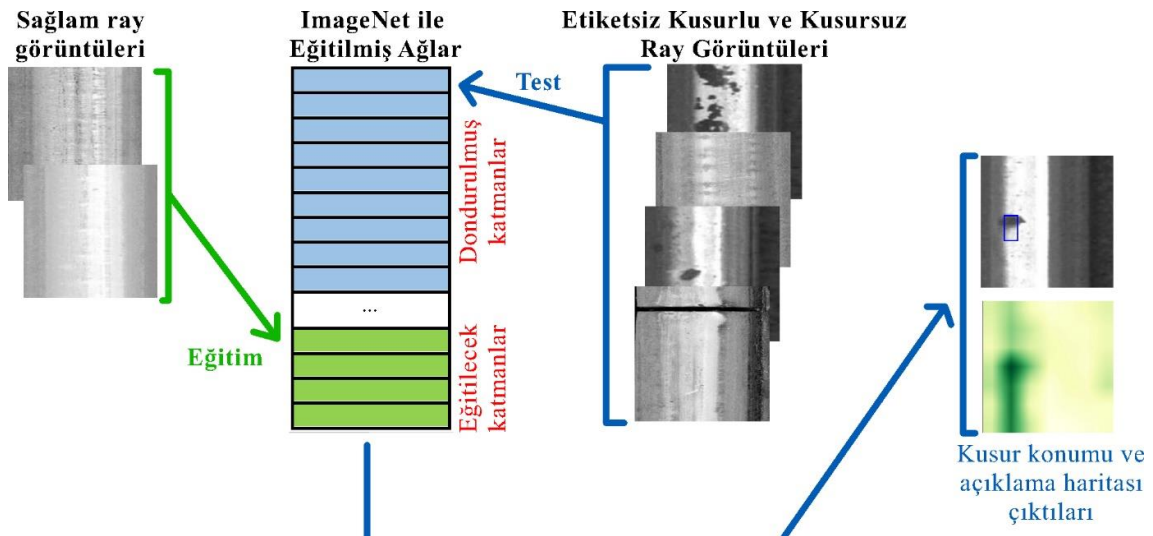
Ray kusurlarının tespit edilebilmesi amacıyla araştırmacılar çeşitli çalışmalar yapmışlardır. Mızrak [12] ray yüzey görüntülerindeki kusurları görüntü işleme teknikleri ve bulanık mantık ile daha belirgin hale getirmeye çalışmıştır. Görüntülerdeki ışık yansımaları ve rayların metalik parlak yapısı nedeniyle tespiti zor olan kusurları sıradan kenar çıkarma algoritmalarına göre daha iyi ortaya çıkarabilmiştir. Çelik [13] görüntü işleme tekniklerini kullanarak ray-tekerlek temas bölgesini tespit etmiştir. Görüntüler üzerinde geliştirdikleri Yatay-Dikey ve Yatay-Yatay algoritmalarını kullanıp son aşamada kusur tespiti için ortalama ve eşikleme yöntemleri kullanmıştır. Aydın vd. [14] ray yüzey kusurlarının tespit etmek için derin öğrenme yöntemlerini kullanmışlardır. Çalışmada GoogleNet ve SqueezeNet ağları 2000 ray görüntüsü ile eğitilmiştir. Eğitim neticesinde GoogleNet %96 doğruluk oranına ulaşmıştır. Wu vd. [15] demiryolu görüntülerinde rayları bölütlemek ve yüzey kusurlarını tespit etmek için hibrit bir derin ağ modeli önermişlerdir. RBGNet'in kullanıldığı çalışmada segmentasyon görevinin daha iyi gerçekleştirilebilmesi için ağı tahmin hatasının hesabında ikili çapraz entropi, yapısal benzerlik indeksi ölçümü, birleşim üzerinde kesişim (Intersection over Union-IoU) metriklerinin birlikte kullanıldığı hibrit bir kayıp hesaplama yöntemi önermişlerdir. Zhang vd. [16] hizmet dışı raylardan toplanan görüntülerin segmentasyonunda karşılaşılan düzensiz aydınlatma, karmaşık arka plan ve kusurlu numune görüntüsü toplama gibi sorunların üstesinden gelebilmek çözümler önermişlerdir. Özellikle karmaşık arka plan bilgisinin olduğu örüntülerde segmentasyon doğruluğunu artırmak için MCnet ağını önermişlerdir. Nieniewski vd. [17] görüntünün morfolojik özellikleri üzerinde uyguladığı algoritmalarla ray yüzey hatalarını tespit etmişlerdir. Çalışmada görüntü işleme teknikleri kullanılmıştır. Feng vd. [18] ray yüzey kusurlarını tespit etmek için MobileNetV2 ile MobileNetV3'ü görüntülerden özellik çıkarma görevinde omurga ağ olarak kullanmışlardır. Omurgada MobileNet sürümlerinin kullanıldığı çalışmanın algılama katmanlarında YOLOv3 ve FPN'den ilham alınarak tasarlanmıştır. Tasarlanan ağ ray yüzey kusurlarını yüksek doğrulukta tespit edebilmiştir. Wang vd. [19] ray yüzey kusurlarını tespit etmek için Mask R-CNN tabanlı yeni bir ağ önermişlerdir. Modelin başarısının değerlendirmesinde IoU metriğinin sınırlılıkları nedeniyle birleşim üzerinde tam kesişim (complete intersection over union-CIoU) metriğini önermişlerdir. Önerilen model çalışmada kullanılan veri setinde %98,7 ortalama kesinlik değerine ulaşmıştır. Ni vd. [20] ray yüzey kusurlarını tespit için bölümlenmiş kenar özellikleri (Partitioned Edge Features - PEF) ile yeni bir algoritma önermişlerdir. Görüntü işleme tekniklerinin kullanıldığı çalışmada kusurlar %88,49 hassasiyetle tespit edilmiştir. Zhang vd. [21] ray yüzey kusurlarını tespit etmek için evrimsel sinir ağlarına dayalı bir sistem önermişlerdir. Önerdikleri tespit sistemi tek atış çoklu kutu dedektörü (single shot multibox detector - SSD) ve YOLOv3 ağını içermektedir. Guo vd. [22] ray yüzey kusurlarının tespiti için özelleştirilmiş bir görüntü seti ile Mask R-CNN ağını eğitmişlerdir. Söz konusu ağın omurgasında farklı ağlar kullanarak deney yapmış, en iyi sonucun ResNet101 omurgası ile elde ettiklerini ifade etmişlerdir. Kou [23] araştırmacıların ray yüzey kusur tespiti ile ilgili yaptıkları çalışmaları derlemiştir. Klasik görüntü işleme yöntemleri ve derin öğrenme

yöntemleri ile ray yüzey kusuru tespit çalışmalarının araştırıldığı çalışmada, görüntü işleme yöntemleri ile doğruluk oranının yaklaşık %80'lere kadar ulaşabildiğini, derin öğrenme yöntemlerinde doğruluk oranının %5-7 oranında iyileştirildiği ifade edilmiştir. Yazar ray yüzey kusur tahmini ile ilgili olarak yapılan çalışmaların yetersiz olduğunu sonuçların istikrarsız olduğunu, derin makine öğrenmesi yöntemlerinin umut vadettiğini ifade etmiştir.

Alan yazın incelendiğinde ray yüzey kusurları ile ilgili olarak çeşitli çalışmalar yapılmasına rağmen halen yetersiz olduğu, araştırmacıların yaptıkları bazı çalışmalarda sadece klasik görüntü işleme tekniklerini kullanırken bazı çalışmalarda ise derin öğrenme yöntemleri ile görüntü işleme tekniklerini harmanladıkları, farklı derin ağları özelleştirerek çözüm önerisi sundukları gözlemlenmektedir. Derin ağların esnek yapısı sayesinde araştırmacılar ağa özgürce müdahale ederek değiştirebilmekte, birden fazla ağ birleştirerek hibrit modeller ortaya çıkabilmektedirler. Tüm bunların yanında ray yüzey kusuru tespiti için kullanılan veriler sayıca yeterli değildir. Ayrıca ray yüzeyindeki parlamalar, ışık dengesizlikleri ve karmaşık arka plan görüntüleri gibi faktörler de araştırmacıları zorlamıştır. Ülkemizde milli imkânlarla yapay zekâ tabanlı otonom lokomotiflerin geliştirilebilmesi için derin öğrenme modellerinin kullanıldığı bilgisayarlı gözü uygulamalarına ağırlık verilmesi büyük önem arz etmektedir. Bu çalışma, açıklanabilir yapay zekâ tabanlı denetimsiz öğrenme modeli ilkelerine uygun olarak derin öğrenme ağları ile ray yüzey kusurlarını tespit etmeyi amaçlamaktadır. Açıklanabilir yapay zekâ ve denetimsiz öğrenme yaklaşımının esas alındığı bu çalışma, ray yüzey kusur tespitinde araştırmacılara yeni bir bakış açısı sunmaktadır.

2. Metot

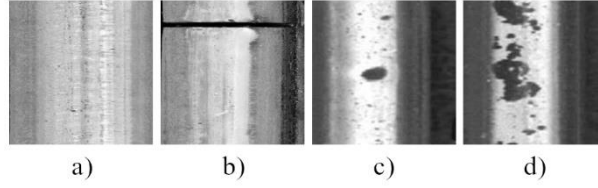
Çalışmada takip edilen yöntemde ilk olarak ImageNet verileri ile önceden eğitilmiş Vgg16 ve MobileNetV3 ağlarının özellik çıkarım katmanlarından bazıları dondurularak gerçekte sağlam olan etiketsiz ray görüntüleri ile alt katmanlar tekrar eğitilmiştir. Ardından sağlam ray görüntülerinin özelliklerini öğrenen ağlara ilk kez verilen test görüntülerinde kusur (anomali) tespiti yaptırılmıştır. Kusurlu bölgenin çerçeve içerisine alındığı çalışmada açıklama haritaları çıkarılmıştır. Bu süreç Şekil 1'de görülmektedir.



Şekil 1. Çalışma yöntemi şeması

2.1. Veri seti

Çalışmada kullanılan veriler lokomotif altına yerleştirilmiş kameralardan elde edilen 1838 görüntüden oluşmaktadır. Şekil 2'de veri setindeki bazı örnekler görülmektedir.



Şekil 2. Veri seti görüntü örnekleri a) Sağlam ray b) Birleşim kusuru c) Hafif çöküntü kusuru d) Büyük çöküntü kusuru

Şekil 2’de görüldüğü gibi veri seti, raylardaki 4 farklı durumu temsil eden görüntülerden oluşmaktadır. Görüntüler 224x224 boyutunda, 3 kanallı “.jpg” formatındadır. Veri setinde 492 adet sağlam, 408 adet birleşim kusuru, 608 adet hafif çöküntü kusuru, 330 adet büyük çöküntü kusuru görüntüsü bulunmaktadır. Bu görüntülerden 443 adet sağlam görüntü derin öğrenme modellerini eğitmek için kullanılmıştır. Geriye kalan 49 adet sağlam ve diğer kusurlu görüntüler etiketsiz olarak modellere test için verilmiştir. Denetimsiz öğrenme gereği modeller, kendilerine etiketsiz olarak verilen bu görüntülerden eğitim esnasında sağlam olanları öğrenecek, test esnasında ilgili görüntülerdeki anormal durumları tespit edecektir. Modeller eğitilirken paket boyutu (batch size) 10, öğrenme oranı (learning rate) 0.0001, devir (epoch) parametresi ise 10 olarak belirlenmiştir. Modellerin eğitimde kayıp fonksiyonu olarak çarpaz entropi (cross entropy) kullanılmıştır. Ayrıca %20 oranında 5 katlı k-fold çarpaz doğrulama ayarlanarak modeller eğitilmiştir..

2.2. Özelleştirilmiş Vgg16 modeli

Vgg16, Simonyan vd. [24] tarafından önerilmiş bir derin öğrenme modelidir. 11 - 19 sayıda katman içerebilecek şekilde farklı konfigürasyonlarla kullanılabilir. Vgg16 konfigürasyonu ImageNet tarafından 2014 yılında yapılan büyük ölçekli görüntü tanıma yarışmasında birinci olmuştur. Tablo 1’de VGG16’nın mimarisi verilmiştir.

Tablo 1. Vgg16 mimarisi

Seviye	Giriş Boyutu	Operatör	Çıkış Boyutu
1.1	$224^2 \times 3$	Conv. (k: 3^2) + ReLU	$224^2 \times 64$
1.2	$224^2 \times 64$	Conv. (k: 3^2) + ReLU	$224^2 \times 64$
1.3	$224^2 \times 64$	Max Pooling (k: 2^2)	$112^2 \times 128$
2.1	$112^2 \times 128$	Conv. (k: 3^2) + ReLU	$112^2 \times 128$
2.2	$112^2 \times 128$	Conv. (k: 3^2) + ReLU	$112^2 \times 128$
2.3	$112^2 \times 128$	Max Pooling (k: 2^2)	$56^2 \times 256$
3.1	$56^2 \times 256$	Conv. (k: 3^2) + ReLU	$56^2 \times 256$
3.2	$56^2 \times 256$	Conv. (k: 3^2) + ReLU	$56^2 \times 256$
3.3	$56^2 \times 256$	Conv. (k: 3^2) + ReLU	$56^2 \times 256$
3.4	$56^2 \times 256$	Max Pooling (k: 2^2)	$28^2 \times 512$
4.1	$28^2 \times 512$	Conv. (k: 3^2) + ReLU	$28^2 \times 512$
4.2	$28^2 \times 512$	Conv. (k: 3^2) + ReLU	$28^2 \times 512$
4.3	$28^2 \times 512$	Conv. (k: 3^2) + ReLU	$28^2 \times 512$
4.4	$28^2 \times 512$	Max Pooling (k: 2^2)	$14^2 \times 512$
5.1	$14^2 \times 512$	Conv. (k: 3^2) + ReLU	$14^2 \times 512$
5.2	$14^2 \times 512$	Conv. (k: 3^2) + ReLU	$14^2 \times 512$
5.3	$14^2 \times 512$	Conv. (k: 3^2) + ReLU	$14^2 \times 512$

5.4	$14^2 \times 512$	Max Pooling (k: 2^2)	$7^2 \times 512$
6.1	$7^2 \times 512$	Fully Connected + ReLU	1×4096
6.2	1×4096	Fully Connected + ReLU	1×4096
6.3	1×4096	Fully Connected + ReLU	1×1000
6.4	1×1000	Softmax	1000

Tablo 1’de görülen orijinal Vgg16 mimarisinde 6.1 seviyesine kadarki operatörler görüntülerden çeşitli özellik haritaları çıkarmaktadır. 6.1 ve sonrasındaki operatörler ise sınıflandırma görevini yerine getirmektedir. Operatörler içerisinde görünen “k: X^2 ” ifadeleri kernel (çekirdek) boyutunu belirtmektedir.

Vgg16 üzerinde yapılan özelleştirme 6.1 ve sonrasındaki operatörleri kapsamaktadır. 6.1 ve 6.2 operatörleri çıkarılarak yerine tek bir ortalama havuzlama katmanı getirilmiştir. Ortalama havuzlama katmanı modelin görüntü üzerinde aranan nesneyi bulması için katkı sağlamaktadır [25]. Bu görevde kusur tespiti yapıldığı için model çıkışının kusurlu ya da kusursuz iki seçeneği olması gerekir. Bu nedenle Softmax operatöründe model çıkışı 2 olarak değiştirilmiştir. Vgg16 sınıflandırıcı katmanlar üzerinde yapılan özelleştirmeler Tablo 2’de görülmektedir.

Tablo 2. Vgg16 Modeli sınıflandırıcı kısmında yapılan özelleştirme

Seviye	Giriş Boyutu	Operatör	Çıkış Boyutu
6.1	$7^2 \times 512$	Avg Pooling (k: 7^2)	$1^2 \times 512$
6.2	$1^2 \times 512$	Fully Connected	1×2
6.3	1×2	Softmax	2

Tablo 2’de özelleştirilen Vgg16 modelinin sonunda bulunan sınıflandırıcı kısmı görülmektedir. Ortalama havuzlama (Avg Pooling) operatöründe 7×7 ’lik kernel (k: 7^2) kullanılmıştır. Operatördeki görüntü giriş boyutu 7×7 ’lik olduğu için aynı boyutta ayarlanan çekirdek, giriş görüntüsündeki tüm piksel değerlerinin ortalamasını alarak tek bir değer elde eder. Bunun sonucunda 7×7 ’lik 512 adet görüntü için 512 sayısal değer elde edilir. Ayrıca yapılan özelleştirmede negatif değerler açıklama haritalarında önem arz ettiği için ReLU fonksiyonu çıkarılmıştır. ReLU fonksiyonu sıfırın altındaki değerleri elemine etmektedir [26]. Derin öğrenme modellerinin ilk katmanları genellikle kenar, köşe bilgisi gibi basit özellikleri çıkarmaktadır [27]. Bu nedenle modelin tamamını eğitmek yerine, önceden eğitilmiş modelin ilk katmanlarını dondurup, sadece son katmanlarının eğitilmesi sağlanmıştır. Yapılan bu işleme literatürde “Transfer Öğrenme” denilmektedir. Bu çalışmadaki transfer öğrenme yaklaşımı “Özellik Çıkarıcı Metot” olarak ifade edilmektedir [28]. Çalışmada Tablo 1’de görülen 4.3 operatörüne kadarki (4.3 dahil) katmanlar dondurulmuş, gerisi eğitim için ayrılan etiketsiz ray görüntüsü verileriyle eğitilmiştir.

2.3 Özelleştirilmiş MobileNetV3 Small modeli

MobileNetV3 Small, Howard vd. [29] tarafından 2019 yılında özellikle cep telefonu CPU’ları için tasarlanmış bir ağıdır. MobileNetV3 Small, özellikle düşük kaynak tüketimi ile ön plana çıkmaktadır. Modelin orijinal mimarisi Tablo 3’te görülmektedir.

Tablo 3. MobileNetV3 Small mimarisi

Seviye	Giriş Boyutu	Operatör	Çıkış Boyutu
1.1	$224^2 \times 3$	Conv. (k: 3^2) + h-swish	$112^2 \times 16$
2.1	$112^2 \times 16$	Bottleneck* (k: 3^2) + ReLU	$56^2 \times 16$

3.1	$56^2 \times 16$	Bottleneck (k: 3^2) + ReLU	$28^2 \times 24$
3.2	$28^2 \times 24$	Bottleneck (k: 3^2) + ReLU	$28^2 \times 24$
4.1	$28^2 \times 24$	Bottleneck* (k: 5^2) + h-swish	$14^2 \times 40$
4.2	$14^2 \times 40$	Bottleneck* (k: 5^2) + h-swish	$14^2 \times 40$
4.3	$14^2 \times 40$	Bottleneck* (k: 5^2) + h-swish	$14^2 \times 40$
4.4	$14^2 \times 40$	Bottleneck* (k: 5^2) + h-swish	$14^2 \times 48$
4.5	$14^2 \times 48$	Bottleneck* (k: 5^2) + h-swish	$14^2 \times 48$
4.6	$14^2 \times 48$	Bottleneck* (k: 5^2) + h-swish	$7^2 \times 96$
4.7	$7^2 \times 96$	Bottleneck* (k: 5^2) + h-swish	$7^2 \times 96$
4.8	$7^2 \times 96$	Bottleneck* (k: 5^2) + h-swish	$7^2 \times 96$
5.1	$7^2 \times 96$	Conv. * (k: 1^2) + h-swish	$7^2 \times 576$
6.1	$7^2 \times 576$	Avg Pooling (k: 7^2)	$1^2 \times 576$
6.2	$1^2 \times 576$	Conv. (k: 1^2) + h-swish	$1^2 \times 1024$
6.3	$1^2 \times 1024$	Conv. (k: 1^2)	$1^2 \times 1000$
6.4	$1^2 \times 1000$	Softmax	1000

Tablo 3'te görülen mimaride "Darboğaz (Bottleneck)" olarak adlandırılan yapı kendi içerisinde 1×1 'lik çekirdeğin uygulandığı evrişim katmanlarını, 3×3 ya da 5×5 'lik çekirdeğin uygulandığı evrişim katmanlarını barındırmaktadır. "Bottleneck*" yapısı ise içerisinde Squeeze and Excitation olarak adlandırılan bir dizi operatör bulundurmaktadır. Bu yapı içerisinde Ortalama Havuzlama (Avg Pool.), Conv. operatörleri ile ReLU ve hard-sigmoid aktivasyon fonksiyonları bulunmaktadır [29]. MobileNetV3 Small üzerinde yapılan özelleştirme Tablo 3'te görülen 6.1 ve sonraki operatörleri kapsamaktadır. Tablo 3'e bakıldığında modelde 6.1 operatörüne kadarki kısmı görüntü özelliğini çıkarırken 6.1 ve sonrası sınıflandırma görevini yerine getirmektedir. Bu çalışmada MobileNetV3 Small modeli sınıflandırıcısı üzerinde Tablo 4'teki özelleştirmeler yapılmıştır.

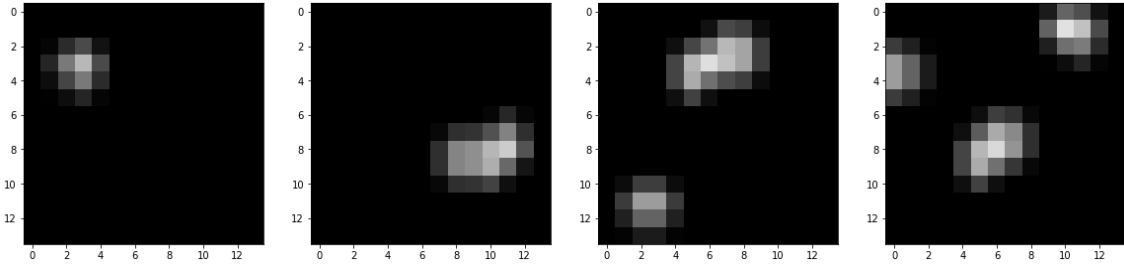
Tablo 4. MobileNetV3 Small Modeli sınıflandırıcı kısmında yapılan özelleştirme

Seviye	Giriş Boyutu	Operatör	Çıkış Boyutu
6.1	$7^2 \times 576$	Avg Pooling (k: 7^2)	$1^2 \times 576$
6.2	$1^2 \times 576$	Fully Connected	1×2
6.3	1×2	Softmax	2

Tablo 4'teki özelleştirmeler yapıldıktan sonra modelin Tablo 3'te görülen 4.6 seviyesine kadarki (4.6 dahil) operatörlerin ağırlıkları dondurulmuş, gerisi eğitim için ayrılan etiketsiz ray görüntüsü verileriyle eğitilmiştir. Burada ağırlıkların dondurulması için belirlenen katman, deneysel olarak en iyi sonucun alındığı katmandır.

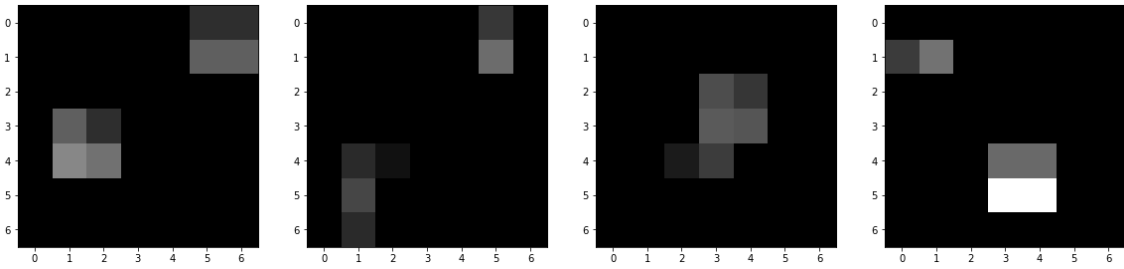
2.4. Açıklama haritalarının oluşturulması

Derin ağlarda basit kenar-köşe bilgileri ilk katmanlarda, daha önemli bilgiler ise son katmanlarda ortaya çıkmaktadır. Bu çalışmada Vgg16 modeli için Tablo 1'de görülen 5.3 operatöründeki evrişim (Conv.) katmanı çıktısı seçilmiştir. Seçilen katman 14×14 'lük 512 adet özellik haritası vermektedir. Şekil 3'te bunlardan birkaçı görülmektedir.



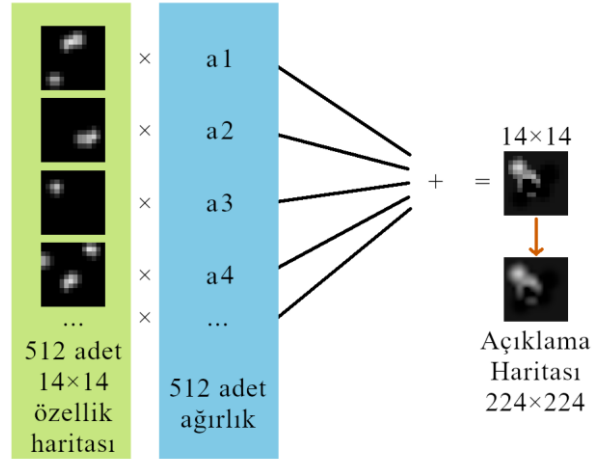
Şekil 3. Vgg16'dan elde edilen bazı özellik haritaları

Benzer biçimde MobileNetV3 Small ağında özellik haritası çıkarımı için Tablo 3'te görülen 5.1 operatöründeki evrişim katmanı seçilmiştir. Bu katman ise 7×7 'lik 576 adet özellik haritası vermektedir. Şekil 4'te bunlardan birkaçı görülmektedir.



Şekil 4. MobileNetV3 Small'dan elde edilen bazı özellik haritaları

Tablo 2'de görülen Vgg16 sınıflandırıcı özelleştirmesinde 6.2 operatörü 512 adet ağırlık değerine sahiptir. Bu ağırlıklar ağ tarafından öğrenilmişlerdir. Vgg16 modeli için özellik haritalarının çıkarılma sürecinde yapılan, bu ağırlık verileri ile Şekil 3'te birkaçı görülen toplamda 512 adet özellik haritasının çarpılarak toplanmasıdır. Şekil 5'te bu süreç gösterilmeye çalışılmıştır.



Şekil 5. Vgg16 modelinde açıklama haritasının oluşturulması

Şekil 5'ten anlaşılacağı gibi özellik haritalarındaki her bir piksel değeri ile karşısındaki ağırlık değeri çarpılarak yeni haritalar elde edilir. Elde edilen 512 adet harita toplanarak tek bir harita elde edilir. Nihayetinde ortaya çıkan açıklama haritası 224×224 boyutuna getirilerek ekranda gösterilir.

MobileNetV3 Small modeli için de açıklama haritalarının ortaya çıkarılma süreci Şekil 5'teki gibidir. Vgg16'dan farkı, Tablo 3'te görünen 5.1 operatörü özellik haritalarının 576 adet ve 7×7'lik boyutta olmasıdır. Elbette ki bu haritalara karşılık gelen ağırlıklar (Tablo 4'te 6.2 operatörü) da 576 adettir. Şekil 5'teki işlemin aynı uygulandığında sonuçta ortaya çıkacak 7×7'lik açıklama haritası 224×224 boyutuna getirilerek kullanıcıya gösterilir. Şekil 3 ve Şekil 4'teki özellik haritaları, Tablo 1 ve Tablo 3'te mimarileri verilen ağların görüntü verileri üzerinde çıkardıkları çeşitli özellikleri göstermektedir. Kullanıcıya gösterilen açıklama haritasının oluşturulmasında ağırlıklar etkili olmaktadır. Ağırlık ne kadar büyük ise kendisiyle çarpılan özellik haritası, açıklama haritasının biçimine o oranda etkili olmaktadır. Dolayısıyla küçük ağırlıklarla çarpılan özellik haritaları, açıklama haritasının biçiminin belirlenmesinde pek fazla etkili olmamaktadır. Esasında açıklama haritası olarak ifade edilen yapı, farklı ağırlıklardaki özellik haritalarının birleşimidir.

3. Bulgular

ImageNet verileriyle önceden eğitilmiş Vgg16 ve MobileNetV3 Small derin öğrenme modellerinin dondurulmayan katmanları 10 epoch eğitilmiştir. Modeller, Tesla T4 GPU ile 27,3 Gb RAM donanım kaynaklarına sahip Google Colab Pro ortamında eğitilmişlerdir. Her bir devir için ortalama eğitim süresi Tablo 5'te görülmektedir.

Tablo 5. Modellerin bir devir için ortalama eğitim süresi

Model	Süre (sn)
Vgg16	6,11
MobileNetV3 Small	5,62

Tablo 5'te görüldüğü gibi MobileNetV3 Small daha hızlı öğrenmektedir. Çapraz doğrulamalı eğitim sonucunda modellerin sınıflandırma başarısını gösteren karmaşıklık matrisi Şekil 6'da görülmektedir.

		Tahminler		Tahminler	
		Sağlam	Kusurlu	Sağlam	Kusurlu
Gerçekler	Sağlam	96	2	91	7
	Kusurlu	3	266	5	264
		a)		b)	

Şekil 6. Karmaşıklık matrisleri a) Vgg16 b) MobileNetV3 Small

Şekil 6 incelendiğinde Vgg16'nın etiketsiz görüntü verilerini daha iyi sınıflandırdığı anlaşılmaktadır. Modellerin doğruluk, kesinlik ve hassasiyet oranları Denklem 1, Denklem 2 ve Denklem 3'teki bağlantılarla bulunabilir.

$$\text{Doğruluk (Accuracy)} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$\text{Kesinlik (Precision)} = \frac{TP}{TP + FP} \quad (2)$$

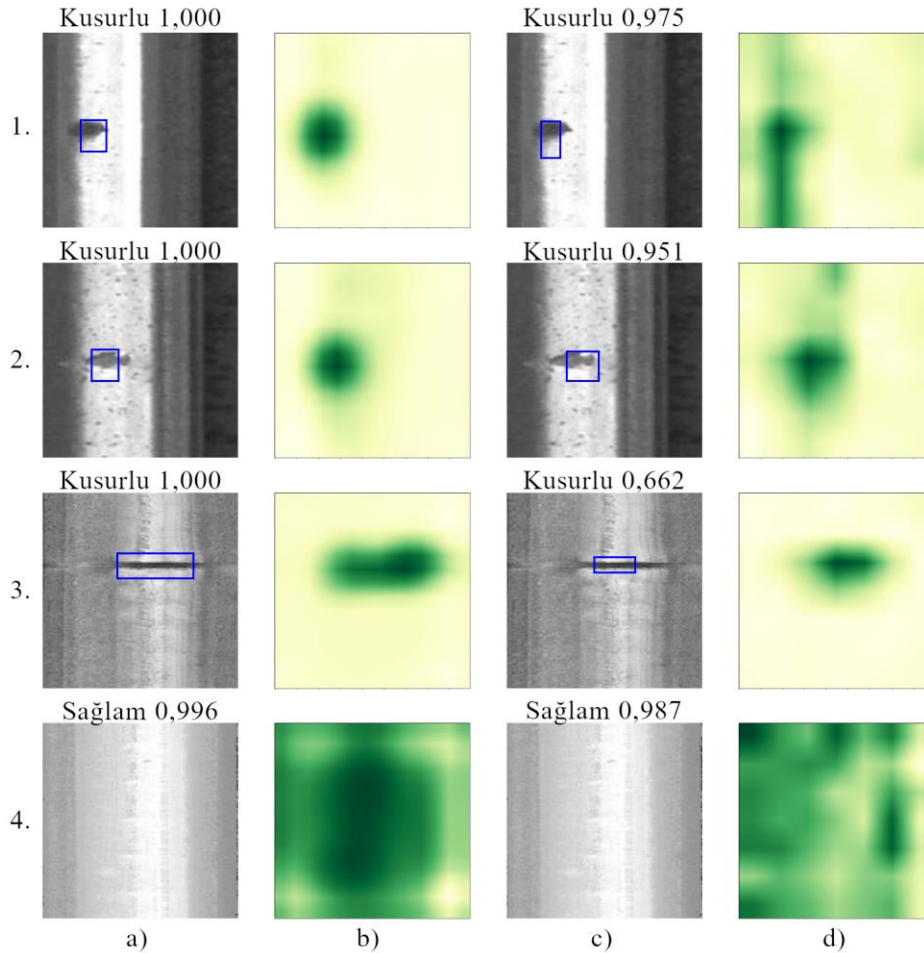
$$\text{Duyarlılık (Recall)} = \frac{TP}{TP + FN} \quad (3)$$

Şekil 6’da görülen karmaşıklık matrisinde gerçekte sağlam olup modeller tarafından sağlam olarak tahmin edilenler TP (True Positive), gerçekte kusurlu olup kusurlu tahmin edilenler TN (True Negative), gerçekte sağlam olup kusurlu tahmin edilenler FN (False Negative), gerçekte kusurlu olup sağlam tahmin edilenler FP (False Positif) olarak adlandırılmaktadır. Buna göre karmaşıklık matrisindeki verilerle Tablo 6’daki sonuçlar elde edilmiştir.

Tablo 6. Modellerin değerlendirme metrikleri

Metrik	Vgg16	MobileNetV3 Small
Doğruluk	0,98	0,96
Kesinlik	0,96	0,94
Duyarlılık	0,97	0,92

Tablo 6 incelendiğinde Vgg16’nın tüm metriklerde üstün geldiği görülmektedir. Şekil 7’de modellerin test görüntüleri için tahminleri ve açıklama haritaları görülmektedir.



Şekil 7. Modellerin test sonuçları a) Vgg16 kusur tespitleri b) Vgg16 açıklama haritaları c) MobileNetV3 Small kusur tespitleri d) MobileNetV3 Small açıklama haritaları

Şekil 7 incelendiğinde mavi çerçeve ile kusur tespiti yapan modellerin bu çerçeveyi neye göre çizdiği hemen yanındaki açıklama haritaları yardımıyla anlaşılmaktadır. Dördüncü satırdaki sağlam ray görüntüsü için modellerin ürettiği açıklama haritalarında yeşil renk, alanın geneline yayılmış iken kusurlu görüntülerde yeşil rengin belirli bir bölgede toplandığı gözlemlenmektedir. Bu durum denetimsiz öğrenme yaklaşımı ile öğrenen ağ modelleri için kusur olarak değerlendirilmekte ve ilgili kusur mavi sınırlayıcı kutu içerisine alınmaktadır. Vgg16'nın açıklama haritaları ve sınırlayıcı kutusunun daha isabetli olduğu, sınıf tahmini yüzdelerinin daha yüksek olduğu görülmektedir.

4. Sonuç

Bu çalışmada açıklanabilir yapay zekâ temelli bir yaklaşım ile ray yüzey kusurları tespit edilmiştir. Vgg16 ve MobileNetV3 Small modellerinin sınıflandırıcı katmanlarının özelleştirilmesiyle yapılan uygulamada, etiketsiz verilerle öğrenme ve ray yüzey kusurlarının açıklama haritaları ile tespitinde Vgg16 modeli 0,98 doğruluk oranı ile daha iyi başarı elde etmiştir. Sonuçlar demiryolları güvenliği için ileride geliştirilebilecek otonom sistemlerin alt yapısında açıklanabilir yapay zekâ uygulamalarının verimli bir şekilde kullanılabileceğini göstermektedir. Asıl karar verici olan insan için önemli veriler sunan açıklanabilir yapay zekâ, olası hataları önleme adına daha fazla katkı sağlayabilir.

Teşekkür

Bu çalışma Fırat Üniversitesi Bilimsel Araştırma Projeleri tarafından ADEP.22.02 proje numarası ile desteklenmiştir.

Kaynakça

- [1] V. Çakmak and A. Altaş, "Sosyal Medya Etkileşiminde Tren Yolculukları: DOĞU EKSPRESİ İle İlgili Youtube Paylaşım Videolarının Analizi," *Journal of Tourism and Gastronomy Studies*, vol. 6, no. 1, pp. 390–408, Mar. 2018, doi: 10.21325/jotags.2018.194.
- [2] P. Ravishankar, X. Zhang, and S. Hwang, "Detecting Defects of Railway Tracks by Using Computer Vision Methodology," in *IIE Annual Conference. Proceedings. Institute of Industrial and Systems Engineers (IISE)*, 2021, pp. 572–577.
- [3] O. Yaman, "Demiryolu Rayları İçin Gerçek Zamanlı Bulanık Otomata ile Görme Tabanlı Arıza Teşhis Sisteminin Geliştirilmesi," Doktora Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, 2018.
- [4] J. McCarthy, *What is AI?*, 2007. [Online]. Available: <http://www-formal.stanford.edu/jmc/>
- [5] M. S. Bingöl, Ç. Kaymak, and A. Uçar, "Derin Öğrenme Kullanarak Otonom Araçların İnsan Sürüşünden Öğrenmesi," *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 31, no. 1, pp. 177–185, 2019.
- [6] J. Gleichauf, J. Vollet, C. Pfitzner, P. Koch, and S. May, "Sensor Fusion Approach for an Autonomous Shunting Locomotive," in *Lecture Notes in Electrical Engineering*, vol. 495, no. January, Springer International Publishing, 2020, pp. 603–624. doi: 10.1007/978-3-030-11292-9_30.
- [7] R. A. S. Deliloğlu and A. Çakmak Pehlivanlı, "Hibrit Açıklanabilir Yapay Zeka Tasarımı ve LIME Uygulaması," *European Journal of Science and Technology*, no. 27, pp. 228–236, Aug. 2021, doi: 10.31590/ejosat.959030.
- [8] R. Terzi, "Sağlık Sektöründe Açıklanabilir Yapay Zeka," in *Yapay Zeka ve Büyük Veri Çalışmaları, Siber Güvenlik ve Mahremiyet*, Ş. Sağıroğlu and U. Demirezen, Eds. Ankara: Nobel Akademik Yayıncılık, 2021, pp. 157–175.
- [9] H. U. Dike, Y. Zhou, K. K. Deveerasetty, and Q. Wu, "Unsupervised Learning Based On Artificial Neural Network : A Review," in *2018 IEEE International Conference on Cyborg and Bionic Systems (CBS)*, 2018, pp. 322–327.
- [10] Z. Ghahramani, "Unsupervised Learning," in *Summer School on Machine Learning*, 2003, pp. 72–112.
- [11] M. Bilgin, "Gerçek Veri Setlerinde Klasik Makine Öğrenmesi Yöntemlerinin Performans Analizi," *Breast*, vol. 2, no. 9, pp. 683–688, 2017.

- [12] C. Mızrak, “Peridinamik Tabanlı Bulanık Mantık Algoritması Yardımıyla Ray Yüzeyindeki Kusurların Tam Spektrum Görüntü İşleme ile Tespiti,” *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, vol. 9, pp. 16–27, Jan. 2020, doi: 10.29130/dubited.831852.
- [13] A. Çelik, “Demiryolu Ray ve Kusurlarını Tespit Etmek İçin Geliştirilen İki Yeni Yöntem,” *Demiryolu Mühendisliği*, no. 12, pp. 52–63, Jul. 2020, doi: 10.47072/demiryolu.737624.
- [14] İ. Aydın, S. S. Kırat, and E. Akın, “Detection of Rail Surface Defects with Two Deep Learning Methods: Comparative Analysis,” in *2022 30th Signal Processing and Communications Applications Conference (SIU)*, May 2022, pp. 1–4. doi: 10.1109/SIU55565.2022.9864863.
- [15] Y. Wu, Y. Qin, Y. Qian, F. Guo, Z. Wang, and L. Jia, “Hybrid deep learning architecture for rail surface segmentation and surface defect detection,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 37, no. 2, pp. 227–244, Feb. 2022, doi: 10.1111/mice.12710.
- [16] D. Zhang, K. Song, J. Xu, Y. He, M. Niu, and Y. Yan, “MCnet: Multiple Context Information Segmentation Network of No-Service Rail Surface Defects,” *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021, doi: 10.1109/TIM.2020.3040890.
- [17] M. Nieniewski, “Morphological Detection and Extraction of Rail Surface Defects,” *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 9, pp. 6870–6879, Sep. 2020, doi: 10.1109/TIM.2020.2975454.
- [18] J. H. Feng, H. Yuan, Y. Q. Hu, J. Lin, S. W. Liu, and X. Luo, “Research on deep learning method for rail surface defect detection,” *IET Electrical Systems in Transportation*, vol. 10, no. 4, pp. 436–442, Dec. 2020, doi: 10.1049/iet-est.2020.0041.
- [19] H. Wang, M. Li, and Z. Wan, “Rail surface defect detection based on improved Mask R-CNN,” *Computers and Electrical Engineering*, vol. 102, no. April, p. 108269, Sep. 2022, doi: 10.1016/j.compeleceng.2022.108269.
- [20] X. Ni, H. Liu, Z. Ma, C. Wang, and J. Liu, “Detection for Rail Surface Defects via Partitioned Edge Feature,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 5806–5822, Jun. 2022, doi: 10.1109/TITS.2021.3058635.
- [21] H. Zhang et al., “MRSDI-CNN: Multi-Model Rail Surface Defect Inspection System Based on Convolutional Neural Networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 11162–11177, Aug. 2022, doi: 10.1109/TITS.2021.3101053.
- [22] F. Guo, Y. Qian, D. Rizos, Z. Suo, and X. Chen, “Automatic Rail Surface Defects Inspection Based on Mask R-CNN,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2675, no. 11, pp. 655–668, Nov. 2021, doi: 10.1177/03611981211019034.
- [23] L. Kou, “A Review of Research on Detection and Evaluation of the Rail Surface Defects,” *Acta Polytechnica Hungarica*, vol. 19, no. 3, pp. 167–186, 2022, doi: 10.12700/APH.19.3.2022.3.14.
- [24] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–14, Sep. 2014, [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [25] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, “Learning Deep Features for Discriminative Localization,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, vol. 2016-Decem, pp. 2921–2929. doi: 10.1109/CVPR.2016.319.
- [26] A. F. Agarap, “Deep Learning using Rectified Linear Units (ReLU),” no. 1, pp. 2–8, Mar. 2018, [Online]. Available: <http://arxiv.org/abs/1803.08375>
- [27] G. Özbek and H. K. Ekenel, “Initialization of convolutional neural networks by Gabor filters,” in *2018 26th Signal Processing and Communications Applications Conference (SIU)*, May 2018, no. May, pp. 1–4. doi: 10.1109/SIU.2018.8404757.
- [28] K. Fırıldak and M. F. Talu, “Evrişimsel Sinir Ağlarında Kullanılan Transfer Öğrenme Yaklaşımlarının İncelenmesi,” *Anatolian Journal of Computer Science*, vol. 4, no. 2, pp. 88–95, 2019.
- [29] A. Howard et al., “Searching for MobileNetV3,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, vol. 2019-Octob, pp. 1314–1324. doi: 10.1109/ICCV.2019.00140.

Özgeçmiş**Selçuk Sinan KIRAT**

1987 yılında Elazığ'da doğdu. 2009 yılında Fırat Ün. / Teknik Eğitim Fakültesi / Bilgisayar Öğretmenliği lisans programından, 2021 yılında Fırat Ün. / Mühendislik Fakültesi / Bilgisayar Mühendisliği lisans programından mezun olmuştur. Fırat Ün. / Fen Bilimleri Enstitüsü / Bilgisayar Mühendisliği anabilim dalında yüksek lisans yapmaktadır. Elazığ İl Milli Eğitim Müdürlüğü bünyesinde Bilişim Teknolojileri ve Yazılım Öğretmeni olarak görev yapmaktadır. Bilgisayarlı görü uygulamaları ile ilgilenmektedir.

E-Posta: selcuk.s.kirat@gmail.com

**İlhan AYDIN**

1981 yılında Elazığ'da doğdu. Lisans ve Yüksek Lisans derecelerini Fırat Üniversitesi Bilgisayar Mühendisliğinde 2001 ve 2006 yıllarında aldı. Doktora derecesini 2011 yılında Fırat Üniversitesi, Elazığ Elektrik ve Elektronik Mühendisliği bölümünden aldı. Halen Fırat Üniversitesi'nde Bilgisayar Mühendisliği bölümünde Doçent 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

Beyanlar:

Bu makalede bilimsel araştırma ve yayın etiğine uyulmuştur.

Yazarların katkıları: İlhan AYDIN: Kavramsallaştırma, Metodoloji, Yazılım. Selçuk Sinan KIRAT: Kaynaklar, Doğrulama, Yazılım, Yazma-gözden geçirme ve düzenleme.