

DERİN ÖĞRENME İLE ASFALT ÇATLAKLARININ TESPİTİNDE VERİ ARTIRIMI VE EVRİŞİMSEL BLOK SEÇİMİNİN ETKİSİ

Zahide TOPBAŞ¹, Özlem ERDAŞ ÇİÇEK^{2*}, Şaban GÜLCÜ²

¹Necmettin Erbakan Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Konya, 42090, Türkiye

²Necmettin Erbakan Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Konya, 42140, Türkiye

Geliş Tarihi/Received Date: 17.03.2024 Kabul Tarihi/Accepted Date: 15.07.2024 DOI: 10.54365/adyumbd.1453460

ÖZET

Petrolün rafine işlemi sonucu yan bir ürün olarak oluşan asfalt, yol çalışmalarında kullanılan en önemli malzemelerdendir. Asfalt yollarda trafik, iklim ve çevre, yapım ve tasarım hataları, malzeme hataları gibi etkenlerden dolayı deformasyonlar, ayrışmalar ve çatlaklar meydana gelmektedir. Bu bozulmalar asfalt yolun kalitesini düşürmekte ve kazalara sebep olmaktadır. Bu çalışma, onarım ve altyapı iyileştirmelerinin sağlanması için çatlaklar gibi asfalt hasarlarının tespitini iyileştirmeyi amaçlamaktadır. Çalışma kapsamında, Mendeley Data'nın "Asfalt Çatlak Veri Seti" üzerinde eğitilmiş bir derin öğrenme modelini kullanarak, çatlak görüntüleri %96'yı aşan bir doğruluk oranıyla sınıflandırılmıştır. Bu araştırma, derin sinir ağlarının endüstriyel uygulamalarda kullanımının ürün kalitesini iyileştirme potansiyelini ortaya koymaktadır.

Anahtar Kelimeler: Derin öğrenme, görüntü sınıflandırma, evrişimli sinir ağları, asfalt çatlağı tespiti, veri artırımı

THE EFFECT OF DATA AUGMENTATION AND CONVOLUTIONAL BLOCK SELECTION ON THE DETECTION OF ASPHALT CRACKS WITH DEEP LEARNING

ABSTRACT

Asphalt, formed as a byproduct through the refining process of oil, is one of the most crucial materials used in road construction. Deformations, separations, and cracks occur in asphalt roads due to factors such as traffic, climate and environment, construction and design errors, and material defects. These deteriorations reduce the quality of asphalt roads and lead to accidents. This study aims to improve the detection of asphalt damages such as cracks to enable repairs and infrastructure improvements. Within the scope of the study, crack images were classified with an accuracy rate exceeding 96% using a deep learning model trained on Mendeley Data's "Asphalt Crack Data Set". This research demonstrates the potential of using deep neural networks in industrial applications to improve product quality.

Keywords: Deep learning, image classification, convolutional neural networks, asphalt crack detection, data augmentation

1. Giriş

Ulaşım çeşitleri kullanım açısından değerlendirildiğinde en yüksek pay karayolu ulaşımına aittir. Karayollarında güvenliğin sağlanması, gereken kontrollerin sağlanarak kazaların azaltılması için altyapı inceleme çalışmaları oldukça önemlidir. Söz konusu çalışmaların sonucunda, problemlerin kök nedenlerinin tespit edilmesi bakım ve onarımın etkin bir şekilde yapılmasına olanak sağlayacaktır.

e-posta¹ : zahidetopbas@gmail.com ORCID ID: <https://orcid.org/0000-0002-9733-8213>

* e-posta² : ozlem.erdascicek@erbakan.edu.tr ORCID ID: <https://orcid.org/0000-0003-4019-7744> (Sorumlu Yazar)

e-posta³ : sgulcu@erbakan.edu.tr ORCID ID: <https://orcid.org/0000-0001-7714-8861>

Endüstri 4.0, 4. Sanayi Devrimi; nesnelerin interneti, internetin hizmetleri ve siber-fiziksel sistemlerden oluşan değerler bütünüdür [1]. Ülkemizde 2021 yılı itibari ile toplam karayollarının 68633 km. olduğu düşünüldüğünde devrimin getirdiği her bir verinin toplanması ve iyi bir şekilde izlenip analiz edilmesi modeli, karayolları üzerindeki etkisi zaman, maliyet ve insan gücü konusunda kaynak tüketimini azaltacaktır [2].

Asfalt, yol çalışmalarında kullanılan en önemli malzemelerden olmakla birlikte aynı zamanda havalimanı, demiryolu, liman inşaatı gibi alanlarda da kullanılmaktadır. Trafik yükü (araç trafiği titreşimleri, fren yükleri vb.), çevre ve iklim etkisi, işçilik hatası gibi etkenler yol üst yapısını bozarak çatlaklara, çukurlara ve çökmelere sebep olmaktadır [3]. Bu bozukluklar zamanında tespit edilemediğinde veya onarımında hata yapıldığında yol alt katmanlarının da bozulmasına, maliyetin artmasına ve trafik güvenliğinin azalmasına sebep olmaktadır [4].

Bu noktada bozuklukları zamanında gören, algılayan, işleyen, değerlendiren ve sonuç çıkaran mekanizmalara ihtiyaç vardır. En üst çatıda yapay zekâ dediğimiz bu kavram, insana özgü nitelikler olan akıl yürütme, anlam çıkarma, genelleme yapma, geçmiş deneyimlerden ders çıkarma gibi yetilerin bir makine tarafından yapılabilmesidir.

Son yıllarda gerçekleşen donanımsal alandaki gelişmeler yapay zekâ biliminin alt bilimi olan makine öğrenmesi yöntemlerinin de gelişmesini sağlamıştır. Çalışmamızda kullandığımız derin öğrenme ise makine öğrenmesi modellerinden yapay sinir ağlarının çok katmanlı mimarisidir [5].

Görüntü yani cihazlardan elde edilen sayısallaştırılmış görüntüler piksel adı verilen satır ve sütun olmak üzere bir birimlik alanlardır. Pikseller RGB yoğunluklarına göre 0 ile 255 arası değer olarak görüntüyü oluştururlar. Görüntü işleme ise çeşitli teknikler kullanarak görüntülerin iyileştirilmesi, değiştirilmesi, analiz edilmesidir. Görüntü işleme yöntemleri derin öğrenme teknikleri ile birlikte kullanıldığında nesne tanıma ve görüntülerin sınıflandırılması konularında iyi sonuçlar elde edilmektedir. Özellikle derin öğrenme modellerinden olan Evrişimsel Sinir Ağı (CNN) ile görüntülerdeki her bir pikselin kullanılması yerine, eklenen filtreler sayesinde öz nitelikler çıkarılarak ayırt edici noktaların tespit edilmesi sağlanmıştır [5]. Bu gelişmeler ürünlerin sınıflandırılması, kalite kontrol, üretimdeki kusur ve hataların tespiti, bozuk ürünlerin ayrıştırılması, hasar tespiti gibi birçok endüstriyel alanda fayda sağlamaktadır. Örneğin, çatlak ve çukur sayıları çok yüksek seviyelere ulaşan Birleşik Krallıkta geliştirmeleri devam eden, 2021 yılı sonunda kullanılmaya başlanması planlanan otonom onarım robotları sayesinde görüntü işleme ve yapay zekâ teknikleri ile yol üzerindeki çatlak ve çukurlar tespit edilerek otomatik olarak onarımı sağlanmaktadır [6].

Literatürde asfalt bozukluklarının tespitinde görüntü işleme ve makine öğrenmesi yöntemlerini kullanan çeşitli çalışmalar mevcuttur. Baygın ve diğerleri [7], asfalt yollardan kamera aracılığı ile elde edilen 3912 adet görüntü ile veri seti oluşturmuşlardır. Görüntülerdeki deforme olan asfaltın görüntü işleme ile tespiti sağlanarak sağlam ve arızalı olarak sınıflandırılması amaçlanmıştır. Asfalt görüntüleri ortaklama yöntemi ile boyutları azaltıldıktan sonra Yönlendirilmiş Degradelerin Histogramı (HOG) algoritması ile özellik çıkarımı yapılmıştır. Ki-kare yöntemi ile her bir görüntü için ağırlıklı özellikler belirlenerek özellik matrisi elde edilmiştir. Elde edilen özellik matrisi Destek Vektör Makinesi (DVM) ile sınıflandırılarak %96,5 doğruluk yakalanmıştır. Schmidt ve diğerleri [8] yaptıkları çalışmada, görüntülerdeki çukurların sınıflandırılabilmesi için HOG ile özellik çıkarımı yaptıktan sonra Yapay Sinir Ağı (YSA) ve DVM olmak üzere 2 adet sınıflandırıcı kullanılmıştır. Farklı veri boyutları ile yaptıkları deneylerde en yüksek veri boyutu ile DVM eğiterek %93 doğruluk değerini elde etmişlerdir. Shatnawi [9] dronlar aracılığı ile elde edilen görüntülerden nesne tanıma yöntemi ile yol çatlakları tespiti gerçekleştirmiştir. Süreçte sırası ile görüntü ön işleme ve iyileştirme, özellik çıkarma, sinir ağının modellenmesi ve sürecin doğrulaması aşamaları gerçekleştirilmiştir. YSA modeli kullanılmış ve %82,5 doğruluk oranı yakalanmıştır. Bazı örneklerin yanlış olarak tespit edilmesinin sebebinin görüntülerde bulunan ağaç dalı, şerit gibi nesnelerin oluşturduğu gürültü olduğu değerlendirilmiştir.

Daha güncel çalışmalarda derin öğrenme yöntemlerinin de asfalt kusurlarının tespitinde kullanıldığı görülmüştür. Mandal ve diğerleri [10] yollardaki çatlakların anında tespit edilebilmesini sağlamak için derin öğrenme modellerinden YOLO v2 kullanmışlardır. YOLO'nun performansını

artırmak için Microsoft COCO veri kümesi ağırlıkları kullanılarak transfer öğrenme gerçekleştirilmiştir. Her bir çatlak tipinin tahmin edilmesi için F1 puanı 0,7394, çatlak varlığının tahmini 0,8780 olarak elde edilmiştir. Cheng ve diğerleri [11] derin evrişimli sinir ağı modellerinden olan U-Net kullanılarak piksel düzeyinde yol çatlakları tespit sistemi önermişlerdir. U-Net ile görüntü bölütleme gerçekleştirildiğinde çatlak görüntülerin olduğu pikseller siyah, çatlak olmayan pikseller beyaz renklidir. Çalışma neticesinde çatlak tespitinde %86,09 F1 puanı ile büyük oranda başarı sağlanmıştır. Deveci ve diğerlerinin yaptığı çalışmada [12] yol anomalisinin tespiti için derin öğrenme mimarilerinden VGG-16 kullanılmıştır. VGG-16 modeli, farklı blok sayılarında (1, 2, 3) uygulanarak sistem performansı ölçülmüştür. En yüksek başarı %95,93 ile 3 bloklu sistemde elde edilmiştir. Liu ve diğerleri [13] asfalt çatlaklarının tespitinde gündüz, gece ve şafak vakti elde edilen görüntülerdeki farklılığı gidermek için orijinal görüntülerin yanı sıra kızılötesi görüntüleri de kullanarak derin ağlar eğitmişlerdir. En başarılı sonuçları 0,809 F1 puanı ile orijinal görüntülerde Öznetelik Piramidi Ağı (FPN) kullanarak elde etmişlerdir. Yine FPN mimarisi kullandıkları kızılötesi görüntülerde 0,642 F1 puanına ulaşabilmişlerdir. Orijinal ve kızılötesi görüntüleri birleştirildikleri ve füzyon adını verdikleri veri modelinde ise FPN mimarisi 0,806 F1 puanına sahip olmuştur. Que ve diğerleri [14] kaldırım çatlaklarını sınıflandırmak için Üretken Rekabetçi Ağları (GAN) ve değiştirilmiş bir derin öğrenme ağını (VGG) birleştiren bir yöntem önermişlerdir. Veri artırma adımı, GAN tabanlı bir model ile eğitim veri kümesini genişletmek için ek çatlak görüntüleri oluşturulmuştur. Çeşitli özellik çıkarım ağları karşılaştırılarak geliştirilmiş bir VGG modeli tasarlanmış ve farklı derin öğrenme yöntemleri ile karşılaştırılmıştır. Sonuçlar, GAN destekli verilerle eğitilen modelin, daha yüksek doğruluk ve daha düşük kayıpla geleneksel yöntemleri kullananlardan daha iyi performans gösterdiğini göstermiştir. Geliştirilmiş VGG modeli, orijinal VGG modeline kıyasla doğrulukta %5,9'luk bir artış (%96,30) ve F1 skorunda %5,78'lik bir artış (%96,23) elde etmiştir. Matarneh ve diğerleri [15] kaldırım çatlaklarının tespiti için çeşitli optimizasyon ve transfer öğrenme yöntemlerini kullanmışlardır. Çalıştıkları veri setinde enine, boyuna ve çapraz çatlak olmak üzere 3 sınıf bulunmaktadır. DenseNet201 ve gri kurt optimizasyon yöntemlerini bir arada kullanarak %94,12 doğrulukla en başarılı sonucu elde etmişlerdir. Yılmaz ve diğerleri [16] deprem sonrası oluşan asfalt çatlaklarının tespiti için bir veri seti oluşturmuşlardır. Relieff optimizasyon yöntemini kullanarak eğittikleri derin öğrenme yöntemleri arasından en başarılı sonucu veren %80,32 doğruluk oranı ile VGG-16 olmuştur.

Sunulan bu çalışmada, Balaji ve diğerlerinin oluşturduğu “Asfalt Çatlak Veri Seti” [17] kullanılmıştır. Aynı veri setini kullanan Balaji ve diğerleri [18] asfalt çatlaklarını tahmin etmede en iyi performans gösteren makine öğrenimi modelini tespit ederek hesaplama gücü düşük cihazlarda kullanılabilir bir uygulama geliştirmeyi amaçlamışlardır. İlk olarak 448×448 piksel çözünürlüğünde olan orijinal veri seti ile eğitim sağlanmıştır. Ardından 224×224, 112×112 şeklinde çözünürlük azaltılarak veri setleri oluşturulmuştur. DVM ile Özellik Çantası (BoF), DVM ile HOG, MobileNets ve Inception-V3 transfer öğrenme tekniklerinin kullanıldığı derin öğrenme modelleri oluşturulan veri setleri ile eğitilerek 12 test gerçekleştirilmiştir. 448×448 çözünürlüklü görüntülerin kullanıldığı MobileNets tabanlı model %99,17 doğruluk oranı ile en iyi performansı göstermiştir. 112×112 çözünürlüklü görüntülerin kullanıldığı HOG+DVM modeli %66,69 doğruluk ile kötü performans göstermiştir. Modelleri eğitirken verinin bölünmesi ile ilgili bilgi verilmediği için verilen performans değerlerinin eğitim kümesine ait olduğu veya tüm verinin eğitim için kullanıldığı düşünülmektedir. Ji [19] tarafından gerçekleştirilen asfalt kaplamalardaki çatlakların tespitini sağlayan çalışmada veri setinin %20'si test, eğitim veri setinde %80'i eğitim, %20'si geçirme olmak üzere bölünme sağlanmıştır. MobileNet mimarisi transfer öğrenmenin tercih edildiği bu modelde aynı veri setine ait test kümesinde %93 doğruluk elde edilmiştir. Kumar ve diğerleri [20] asfalt ve köprü üzerindeki çatlakların tespiti için deneyler yapmış ve aynı veri setini kullanmışlardır. Geliştirilmiş LeNet5 ağı kullanarak “Asfalt Çatlak Veri Seti” üzerinde %82 doğruluk oranına ulaşmışlardır. Bu çalışmada derin öğrenme modellerinden Evrişimsel Sinir Ağları (Convolutional Neural Networks - CNN) kullanılmıştır. Özelliklerin çıkarılmasında transfer öğrenme tekniği kullanılmış ve VGG16 mimarisi ImageNet ağırlıkları ile eğitilmiştir. Özellikle 2018 yılından sonra yapılan çalışmalarda [10-20] derin öğrenme mimarilerini değiştirmeden veya katman ekleyip çıkararak kullanıldığı, özellik seçim ve optimizasyon algoritmalarının katkısıyla farklı modellerin oluşturulduğu gözlemlenmiştir. Söz

konusu çalışmalardan farklı olarak bu çalışmada VGG16 mimarisinin bazı katmanları eğitilmiş, diğer katmanları orijinal ImageNet ağırlıkları ile kullanılmış ve evrimsel blok seçiminin performansa etkisi incelenmiştir [21-22]. Aynı zamanda veri artırımı uygulanarak başarımın yükseltilmesi hedeflenmiştir. Gerçekleştirilen testlerde %96 ve üzerinde test doğruluğu elde edilmiştir. Modelin geliştirilmesinde Python programlama dili ile Tensorflow ve Keras kütüphaneleri kullanılmıştır. Donanım kısıtlarından dolayı çalışmalar Google Colaboratory ortamında yapılmıştır.

Materyal ve Metot

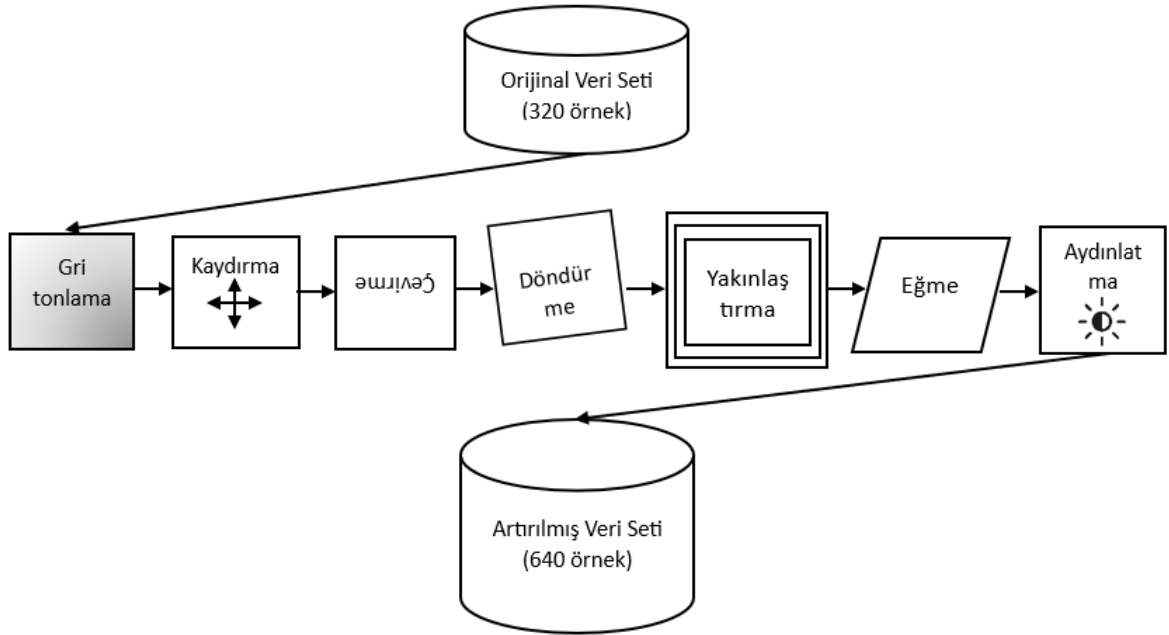
1.1. Veri Toplama

Veri seti, toplam 400 adet düzenli olarak dağıtılmış görsel verilerden oluşmaktadır [17-18]. Asfalt çatlaklarının bulunup bulunmamasına göre “Crack” ve “NonCrack” olmak üzere 2 sınıftan oluşmaktadır. Eğitim veri seti için 320 resim, geçerleme veri seti için 40 resim, test veri seti için 40 resim kullanılmıştır. Aynı zamanda mevcut eğitim veri setine veri artırımı ile elde edilen veri seti eklenerek eğitim veri seti için 320 “Crack”, 320 “Noncrack” olmak üzere 640 resim elde edilmiştir. Veri setindeki örnek sayısı az bulunduğu için eğitim-geçerleme-test verisi ayrımının yanı sıra 5-katlı çapraz geçerleme uygulanarak sonuçlar karşılaştırılmıştır. 5-katlı çapraz geçerlemede veri kümesi 5 eşit parçaya bölünür. Her deneyde bir parça test için diğer 4 parça ise eğitim için kullanılır ve deney 5 parça için tekrarlanır [23].

1.2. Ön İşleme ve Veri Artırımı

Görüntü verisinin artırılması, görüntülerin değiştirilmiş hallerini kullanarak veri kümesinin özellikle eğitim veri setinin yapay olarak boyutunun büyütülmesidir. Derin öğrenme sinir ağında veri sayısının artırılması modelin performansını ve genelleştirme yeteneğini artırır [24]. Çalışmada, modelin doğruluğunun artırılması ve küçük veri setlerinde oluşan aşırı öğrenmenin önüne geçilebilmesi amacıyla “Image Data Generator” kütüphanesine ait aşağıda belirtilen teknikler (Şekil 1) eğitim kümesine uygulanmıştır.

- Görüntüye ait 0-255 arasında bulunan pikseller 0-1 arasında boyutlandırılmıştır. Bu sayede hem çatlak yapılarının daha belirginleşmesi sağlanmış, hem de derin öğrenme mimarisinin çalışma hızı artırılmıştır.
- Görüntü boyutları sabit tutulurken, pikseller enine veya boyuna kaydırılmıştır.
- Piksel satır ve sütunlarının yerleri değiştirilerek görüntü çevrilmiştir.
- Görüntüler 20° açı ile döndürülmüştür.
- Görüntü piksel sayısı artırılarak ya da azaltılarak rasgele yaklaştırılmıştır ya da uzaklaştırılmıştır. İşlem gerçekleşirken interpolasyon yani değeri bilinmeyen piksellerin tahmin edilmesi yöntemi kullanılır. Yakınlaştırma aralığı 0,2 olarak belirlenmiş, dolayısıyla %80 yakınlaştırma, %120 uzaklaştırma arasında değişim sağlanmıştır.
- Rasgele parlaklık değiştirme tekniği ile görüntülerin farklı parlaklık seviyeleri artırılmıştır. 1,0'dan büyük değerler görüntüyü aydınlatırken, 1,0'den küçük değerler görüntüyü karartmaktadır.
- Görüntü 0,2 oranında eğimlendirilmiştir.
- Görseller kaydırılırken ve eğimlenirken oluşan bulanık bölgeler en yakın piksellerin değerine göre otomatik olarak doldurulmuştur.



Şekil 1. Veri artırım adımları

1.3. Derin Öğrenme ve Evrişimli Sinir Ağları (ESA-CNN)

Yapay zekâ, en genel anlamı ile insana ait düşünme, öğrenme, tanıma, karar verme, geçmişten ders çıkarma gibi yetilerin makinelere kazandırılmasını amaçlayan bilimdir. Bu bilimin alt kümesi olan makine öğrenmesi, makinelerin matematiksel modeller sayesinde veri analizleri gerçekleştirerek öğrenmesini sağlayan bir teknolojidir. Çalışmamızda kullandığımız derin öğrenme yöntemi ise makine öğrenmesinin alt kümesidir ve insandaki sinir sisteminden ilham alan yapay sinir ağları algoritmasının çok seviyeli, derin yaklaşımıdır.

Evrişimli sinir ağları, çok katmanlı yapay sinir ağlarının bir türüdür. Derin öğrenme bilimine ait temel mimari olarak kabul edilir. ESA’nda giriş verileri alındıktan sonra katman erişimli eğitim gerçekleştirilir. Üretilen sonuç ile istenen sonuç arasındaki fark yani hata değeri geri yayılım algoritması (backpropagation) ile bütün ağırlıklar üzerinde dağıtılır. Her iterasyonda (epoch) ağırlıklar güncellenerek hata giderilmeye çalışılır [25-26].

Evrişim (convolution) katmanında resim üzerindeki düşük ve yüksek seviyeli özelliklerin çıkarılabilmesi için çok boyutlu filtreler uygulanarak özellik haritası (feature map) çıkarılır. Evrişim işleminin ardından ReLU aktivasyon fonksiyonu uygulanarak görüntü matrisi üzerindeki negatif değerler sıfırlanır. Burada farklı bir aktivasyon fonksiyonu da tercih edilebilir fakat performans olarak ReLu daha iyi sonuç verdiği için bu yöntem tercih edilmektedir. Ortaklama (pooling) katmanı evrişim katmanları arasına çoğunluk olarak eklenen bir katmandır. Bu katman sinir ağının doğru olarak karar verebilmesi için boyutları azaltarak önemli bilginin edinilmesini sağlar. Filtre kayma işleminde kapsadığı alan içindeki en büyük sayıyı aldığı anda maksimum ortaklama (max pooling), ortalamasını aldığı anda ortalama ortaklama işlemi gerçekleştirilir [27].

Özelliklerin çıkarılması işleminden sonra sınıflandırma işleminin gerçekleştirilmesi için evrişim ve ortaklama katmanından gelen matrisler düzleştirme (flattening) katmanına gelir ve tek boyutlu dizi haline dönüştürülür.

Tam bağlı (fully-connected) katmanı öğrenmenin gerçekleştiği katmandır. Çok katmanlı sinir ağı yapısına benzerdir. Çıktılar sınıflandırmanın sağlanması için aktivasyon fonksiyonuna verilir. Nesnelerin sınıflandırılması sağlanır [5, 27].

Yapmış olduğumuz çalışmada sinir ağı oluşturulurken özelliklerin çıkarılması sürecinde transfer öğrenme tekniği kullanılmıştır. Ardından ESA mimarisine uygun olarak sınıflandırmanın sağlanabilmesi için katmanlar sıralı olarak eklenmiştir. Sinir ağı modelimize ImageNet verileri ile ağırlıklandırılmış VGG16 mimarisi block5_conv2, block2_conv2 ve block1_conv1 katmanından itibaren eğitilmiştir. Belirtilen katmanlardan önce gelen katmanlara ait parametrelere eğitim sırasında müdahale edilmemiştir. Modele düzleştirme katmanı uygulanmıştır. Ardından tahmin için tam bağlı katmanlar eklenerek model oluşturulmuştur.

VGG16 katmanlarından sonra modelimiz 128 ve 64 nörondan oluşan LeakyReLU ile aktive edilmiş 2 katmandan oluşmaktadır. Ardından 2 nörondan oluşan softmax ile aktive edilmiş çıkış katmanı gelmektedir. Bu aşamada kullanılan derin öğrenme mimarisi Şekil 2'de gösterilmektedir.

```
model.summary()
Model: "sequential"

```

Layer (type)	Output Shape	Param #
vgg16 (Functional)	(None, 7, 7, 512)	14714688
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 128)	3211392
leaky_re_lu (LeakyReLU)	(None, 128)	0
dense_1 (Dense)	(None, 64)	8256
leaky_re_lu_1 (LeakyReLU)	(None, 64)	0
dense_2 (Dense)	(None, 2)	130

```

=====
Total params: 17,934,466
Trainable params: 17,934,466
Non-trainable params: 0

```

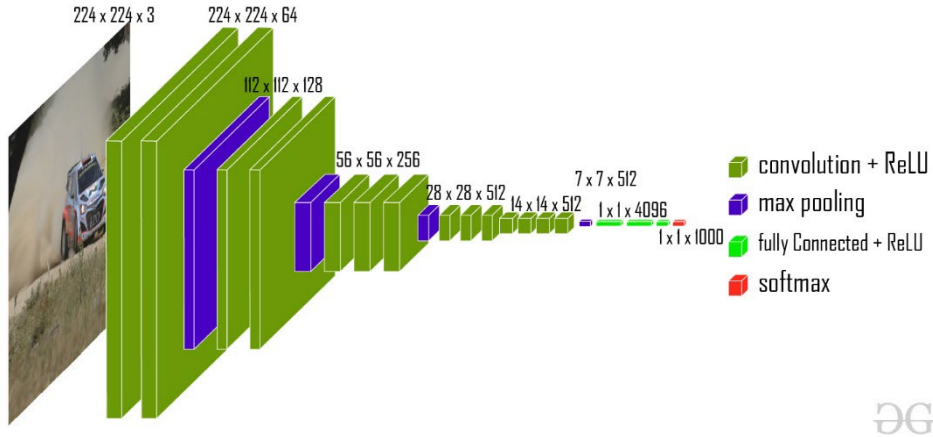
Şekil 2. Eğitim için hazırlanan model katmanları

1.4. Transfer Öğrenme

Transfer öğrenme yaklaşımı, insanın öğrenme modelini örnek alır. Nasıl ki insan daha önce karşılaşmamış olduğu problemlerin çözümü sürecinde geçmiş deneyimlerinden faydalanır, derin ağlarda da rastgele ağırlıklarla eğitilmiş bir ağı ağırlıkları kullanılarak öğrenme gerçekleştirilir.

VGG-16 mimarisi 2014 yılında 1000 sınıfa ait 14 milyonun üzerindeki görüntünün sınıflandırılmasını içeren ImageNet adlı yarışmada %92,7 performans göstererek dikkatleri üzerine çekmiştir [28].

Yapmış olduğumuz çalışmada asfalt çatlak görüntülerinin yer aldığı veri kümesine transfer öğrenme tekniği uygulanmış ve ImageNet verisi ağırlıkları ile önceden eğitilmiş VGG-16 mimarisi (Şekil 3) kullanılmıştır. Görüntülerin giriş boyutları (224, 224, 3) olarak belirlenmiştir (Şekil 4).



Şekil 3. VGG-16 mimarisi (geeksforgeeks.com'dan alınmıştır)

▶ OZELLIK_CIKARAN_MODEL.summary()

Model: "vgg16"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0

Total params:	14,714,688	
Trainable params:	14,714,688	
Non-trainable params:	0	

Şekil 4. ImageNet verisi ağırlıkları ile eğitilen VGG16 modeli

1.5. Aktivasyon Fonksiyonları

Bir sinir ağında nöronların giriş değerleri ile ağırlıklar çarpılır ve bias eklenerek çıktı elde edilir. Aktivasyon fonksiyonları üretilen çıktılar doğrusal yapıdan uzaklaştırılarak ve gerçeğe yakın olarak nasıl standardize edilebileceğini, nasıl bir değişimden geçmesi gerektiğini belirler. Modelimizde LeakyReLU ve softmax aktivasyon fonksiyonları kullanılmıştır.

Kullanılacak aktivasyon fonksiyonunun belirlenmesinde doğrusal olması, türevlenebilir olması, alt ve üst sınırlarının olması, monoton olarak artan ve azalan olması nitelikleri olumlu olarak etkilidir. Kullanacağımız fonksiyonları belirlerken bu kriterlere dikkat ederek seçim yaptık.

Fonksiyon türevi “0” olduğunda, geri yayılım esnasında parametreler güncellenmediği için öğrenme gerçekleşmez. Bu problem ölen ReLU olarak adlandırılır. Bu sorunla karşılaşmamak için fonksiyonlar incelenmiş ve tam bağlı katmanda Leaky ReLU fonksiyonu (Denklem 1) uygulanmıştır [29]. Burada k , sıfıra yakın küçük bir sayıdır. Öğrenmenin tüm değerler için gerçekleşmesi amaçlanmıştır.

$$Leaky\ RELU(x) = \max(x, kx) = \begin{cases} x, & \text{eğer } x > 0 \text{ ise} \\ kx, & \text{eğer } x \leq 0 \text{ ise} \end{cases} \quad (1)$$

Derin öğrenme modelimizde çıkış fonksiyonu olarak “softmax” kullandık. Bu aktivasyon fonksiyonu sayesinde “0” ile “1” aralığında olasılıksal olarak kayıp değeri üretilerek girdilerin hangi sınıfa ait olduğu belirlenir. Softmax kayıp değerini çapraz entropi (cross entropy) kullanarak hesaplar [30].

Bilginin kalitesinin ve doğruluğunun ölçülmesi için kullanılan yöntemlerden biri entropidir [31]. Entropi belirsizliğin ölçüsüdür ve düzensizlik anlamına gelir. Bilgi teorisinde entropi, bir olasılık dağılımından rasgele bir olayın olma olasılığının hesaplanabilmesi için bilgi ölçümüdür. Entropi denkleminde (Denklem 2) n olay sayısını ifade ederken p_i de i . olayın gerçekleşme olasılığını gösterir.

$$H = - \sum_{i=1}^n p_i \log(p_i) \quad (2)$$

Bir olayın olma olasılığı yüksek ise daha az şaşırtıcıdır ve düşük bilgiye sahiptir, olay her şeyin eşit olarak dağılımı gibi düşük olasılıklı bir olay ise çok şaşırtıcıdır ve yüksek bilgiye sahiptir. Sonuç olarak gerçekleşmesi beklenmeyen nadir olaylar daha bilgilendiricidir ve rasgele bir x değeri için bilginin hesaplanması, x değişkenine ait olayların olasılık dağılımını hesaplamak ile aynı şeydir. Çapraz entropi ise entropi temeline dayanan belirli bir rasgele değişken ya da olay kümesi için iki olasılık dağılımı arasındaki farkın ölçülmesini sağlayan yöntemlerden biridir.

1.6. Sinir Ağı Modelinin Bileşenleri ve Hiperparametreler

Modelin eğitim sürecini oluşturmamız için tanımlamamız gereken parametreler mevcuttur. Bu parametreler kayıp fonksiyonunun belirlenmesi, optimizasyon algoritmasının belirlenmesi, metrik değeri, yığın boyutu ve iterasyon sayısıdır.

1.6.1. Kayıp Fonksiyonları

Sinir ağındaki son katmanda kayıp fonksiyonu tanımlanır. Kayıp fonksiyonları, sinir ağı modelini bir bütün olarak değerlendirerek her iterasyonda sonuç olarak bir kayıp değeri sunar. Kayıp değerinin büyüklüğüne göre eğitim sonucu elde edilen ağırlık ve bias parametrelerinin modelimiz için ne kadar uygun olduğu hakkında bilgi sahibi oluruz ve çalışmalarımıza yön veririz. Kayıp fonksiyonu problem tipine göre belirlenir.

Sınıflandırma problemleri için “kategorik çapraz entropi (categorical_cross entropy)” veya “ikili çapraz entropi (binary_crossentropy)” fonksiyonları seçilebilir. Çalışmada kullanılan veri “crack” ve “noncrack” olmak üzere 2 sınıf içerdiği için “ikili çapraz entropi” fonksiyonu seçilmiştir [30].

1.6.2. Optimizasyon Algoritmaları

Sinir ağı modelimizi oluştururken amacımız tahmin işleminin doğru olarak gerçekleştirilerek kayıp değerinin en düşük olduğu noktanın bulunmasıdır. Sinir ağında bu işlem geri yayılım algoritması (backpropagation) ile ağırlık değerlerinin önceki katmanlara yayılımı ile gerçekleştirilir. Her iterasyonda hatanın giderilmesi için yapılan bu işlemde ağırlıklar optimizasyon fonksiyonları ile değiştirilir. Optimize edici, öğrenme oranı (learning rate) parametresi ile öğrenme oranını belirler.

Yapmış olduğumuz çalışmada daha az salınımlı, daha hızlı ve tutarlı bir optimizasyon algoritması kullanılması tercih edildiği için Rmsprop (root mean square error propability) optimizasyon tekniği $1e-5$ öğrenme sabiti ile birlikte kullanılmıştır [32-33].

1.6.3. Metrikler

Modelimizin eğitim sırasında nasıl bir performans gösterdiğini anlayabilmek ve her bir iterasyonda doğruluk ve kayıp değerlerini görebilmek için ‘accuracy’ parametresini seçtik. Doğruluk metriği, doğru tahmin edilen örnek sayısının toplam örnek sayısına oranı şeklinde tanımlanmıştır. Bu metrik model eğitilirken kullanılacaktır. Model performansının değerlendirilmesi sırasında kullanılacak ek metrikler “1.7. Model performansının değerlendirilmesi” alt bölümünde anlatılmıştır.

1.6.4. Yığın Boyutu (batch_size)

Sinir ağı modelinin aynı anda ne kadar veriyi işleyebileceğinin belirlendiği hiperparametredir. Hata oranının düşürülmesi için her iterasyonda geriye dönük olarak gradyan (gradient descent) hesabı ile ağırlıklar güncellenmektedir. Veri sayısına göre hesaplama süresi değişim göstermektedir. batch_size parametresini en küçük 1 olarak girdiğimizde modelimizin gürültüyü de öğrenmesine sebep oluruz. Bu şekilde “stochastic gradient descent” optimizasyon işlemini gerçekleştirmiş oluruz. Ters olarak boyutumuz büyük olduğunda ise aşırı öğrenme durumu oluşur. Yığın boyutu her bir iterasyondaki adım sayısı ile belirlenir. Adım sayısı 10 olarak belirlendiğinde 320 veri örneği için yığın boyutu 32 olarak belirlenmelidir. “2. Bulgular” bölümünde farklı deney kurulumları için kullanılan yığın boyutları belirtilmiştir.

1.6.5. Eğitim İterasyon Sayısı (epoch)

Öğrenme işleminin gerçekleşebilmesi için eğitimin tekrarlanarak hata değerinin düşürülmesi gerekmektedir. Tüm eğitim verisinin işlendiği bu tekrarlara epoch denilmektedir. Doğruluğun en yüksek olduğu yerde eğitim tamamlanacak şekilde epoch hiper parametre değeri ayarlanır.

1.7. Model performansının değerlendirilmesi

Asfalt çatlaklarının tahmin edilmesi için oluşturmuş olduğumuz modelin performansının değerlendirilmesi amacıyla doğruluk, kesinlik (precision), duyarlılık (recall), f-skor, ROC-AUC ve karışıklık matrisi (confusion matrix) metrik değerlerini elde ettik.

1.7.1. Karmaşıklık/Hata Matrisi

Karmaşıklık matrisi gerçek ve tahmin edilen sonuçların karşılaştırılabilmesi için kullanılır. Karşılaştırabilmek için TP (True-Positive), TN (True-Negative), FP (False-Positive), FN (False-Negative) değerlerini kullanır. Çalışmamız üzerinden tanımlayacak olursak “0” çatlak içeren görüntüleri, “1” çatlak olmayan görüntüleri göstermektedir [34].

Çatlak içermeyen bir görüntü hakkında çatlak olmadığı hakkında bir değerlendirme yapılıyorsa TP, gerçekte görüntü çatlak içermekte ancak tahmin yanlışsa FP olmaktadır. Gerçekte görüntü çatlak içermiyor ve tahmin doğru ise TN, gerçekte görüntü çatlak içermiyor ve tahmin yanlış ise FN şeklinde tanımlanır. Çizelge 1’i karmaşıklık matrisi üzerinden tanımlarsak, karmaşıklık matrisi üzerindeki değerler kullanılarak diğer metrik değerleri hesaplanır.

Çizelge 1. Karmaşıklık matrisi gösterimi

<i>Tahmin / Gerçek</i>	<i>Noncrack</i>	<i>Crack</i>
<i>Noncrack</i>	TP	FN
<i>Crack</i>	FP	TN

1.7.2. Doğruluk

Doğru tahmin edilen örneklerin sayısının tüm örneklerin sayısına oranıdır.

$$\text{Doğruluk} = \frac{TP + TN}{TP + FP + FN + TN} \quad (3)$$

1.7.3. Duyarlılık

Doğru tahmin edilen pozitif örneklerin sayısının gerçekteki pozitif örneklerin sayısına oranıdır.

$$\text{Duyarlılık} = \frac{TP}{TP + FN} \quad (4)$$

1.7.4. Kesinlik

Doğru olarak tahmin edilen pozitif örneklerin sayısının tüm pozitif olarak tahmin edilen örneklerin sayısına oranıdır.

$$\text{Kesinlik} = \frac{TP}{TP + FP} \quad (5)$$

1.7.5. F-Skor

Duyarlılık ve kesinlik değerlerinin harmonik ortalamasıdır. Modelin kesinlik ve duyarlılığının ölçüsüdür ve en az 0, en fazla 1 değerlerini alabilir.

$$F - \text{Skor} = \frac{2 \times \text{Duyarlılık} \times \text{Kesinlik}}{\text{Duyarlılık} + \text{Kesinlik}} \quad (6)$$

1.7.6. Eğri Altındaki Alan (ROC-AUC):

ROC eğrisi altındaki alanı göstermektedir. AUC değerinin yüksek olması modelin pozitif ve negatif değerleri ne kadar doğru sınıflandırdığının ölçüsüdür [35].

2. Bulgular

Gerçekleştirilen deneylerde eğitilen katman sayısının model başarısına olan etkisinin gözlemlenebilmesi için özellik çıkarma aşamasında VGG16 mimarisinden faydalanılarak block5_conv2 (Şekil 5), block2_conv2 (Şekil 6) ve tüm katmanlar (Şekil 7) olmak üzere modeller belirlenmiştir. Şekil 3'te verilen modele göre block5_conv2 katmanından başlayarak model oluşturduğumuzda eğitilen 2 evrişim, 1 ortaklama katmanı olmak üzere en az eğitilmiş model; block2_conv2 katmanından başlayarak model oluşturduğumuzda eğitilen 10 evrişim, 4 ortaklama katmanına sahip model; block1_conv1 katmanından başlamak üzere tüm katmanları kullandığımızda ise eğitilen 13 evrişim katmanı, 5 ortaklama katmanından oluşan en eğitilmiş model oluşturulmuştur. Eğitilen katman sayısı arttıkça eğitilebilir parametre sayısının da arttığı görülmüştür.

```
model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
vgg16 (Functional)	(None, 7, 7, 512)	14714688
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 128)	3211392
dense_1 (Dense)	(None, 64)	8256
dense_2 (Dense)	(None, 2)	130

=====
Total params: 17,934,466
Trainable params: 7,939,394
Non-trainable params: 9,995,072

Şekil 5. VGG16 block5_conv2 katmanı eğitilebilir parametre bilgisi

```
model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
vgg16 (Functional)	(None, 7, 7, 512)	14714688
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 128)	3211392
dense_1 (Dense)	(None, 64)	8256
dense_2 (Dense)	(None, 2)	130

=====
Total params: 17,934,466
Trainable params: 17,821,890
Non-trainable params: 112,576

Şekil 6. VGG16 block2_conv2 katmanı eğitilebilir parametre bilgisi

```

model.summary()

Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
vgg16 (Functional)          (None, 7, 7, 512)          14714688
flatten (Flatten)           (None, 25088)              0
dense (Dense)                (None, 128)                3211392
dense_1 (Dense)             (None, 64)                 8256
dense_2 (Dense)             (None, 2)                  130
-----
Total params: 17,934,466
Trainable params: 17,934,466
Non-trainable params: 0
-----

```

Şekil 7. VGG16 tüm katmanlar eğitilebilir parametre bilgisi

Gerçek veri ve artırılmış veri ile eğitilen modelleri kullanan 6 farklı deney kümesi Çizelge 2’de verilmiştir.

Çizelge 2. Deney kümeleri

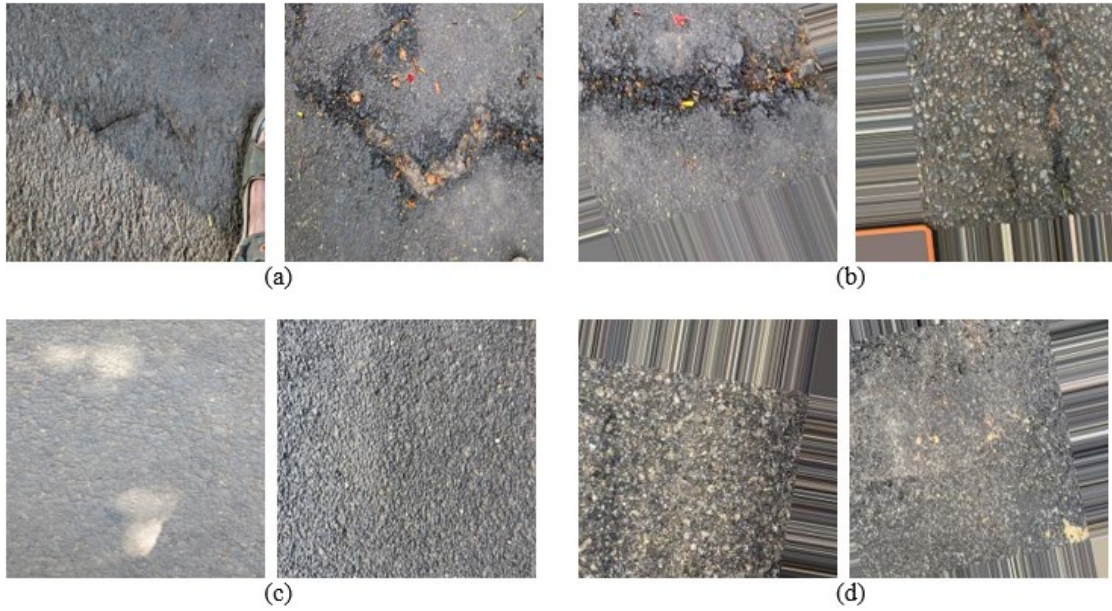
No	Özelliklerin çıkarılması (VGG16)	Veri kümesi	Açıklama
I.	block5_conv2	Gerçek	block5_conv2’den itibaren katmanlar eğitilmiştir.
II.	block2_conv2	Gerçek	block2_conv2’den itibaren katmanlar eğitilmiştir.
III.	block1_conv1	Gerçek	Tüm katmanlar eğitilmiştir.
IV.	block5_conv2	Gerçek+Artırılmış	block5_conv2’den itibaren katmanlar eğitilmiştir.
V.	block2_conv2	Gerçek+Artırılmış	block2_conv2’den itibaren katmanlar eğitilmiştir.
VI.	block1_conv1	Gerçek+Artırılmış	Tüm katmanlar eğitilmiştir.

Eğitim verisi belirlenen modellere verilirken iterasyon (epoch) sayısı 3, eğitim için yığın boyutu gerçek veride 32 ve artırılmış veride 64, iterasyon başına adım sayısı (steps_per_epoch) 10, geçerleme için yığın boyutu 10, geçerleme adımları 3 olarak tanımlandığında eğitim verisi doğruluk oranı I için 0,8675, II için 0,8995 ve tüm katmanlar (III) için 0,9656 şeklinde sonuçlanmıştır. Eğitilebilir parametre sayısı arttığı müddetçe eğitimde doğruluk oranının arttığı gözlemlenmiştir (Çizelge 3). Buna karşın, tüm katmanlar kullanıldığında eğitim doğruluğu artarken test ve geçerleme verilerinin doğruluğunda azalma gözlenmiştir.

Çizelge 3. Gerçek veri ile eğitilen modellerin doğruluk değeri

No	Özelliklerin çıkarılması (VGG16)	Eğitim	Test	Geçerleme
I.	block5_conv2	0,8675	1,0	1,0
II.	block2_conv2	0,8995	1,0	1,0
III.	block1_conv1	0,9656	0,98	0,969

Çalışmanın bu kısmında veri seti büyüklüğünün model başarımına olan etkisinin gözlemlenebilmesi için “Ön İşleme ve Veri Artırımı” başlığı altındaki işlemler gerçekleştirilmiştir. 320 eğitim verisinden oluşan veri seti artırılarak 640 “noncrack” ve “crack” etiketli eğitim verisi oluşturulmuştur. Yeni eğitim veri seti artırılan veri seti ve gerçek veri setinden oluşmaktadır. Test ve geçerlemede gerçek veri örnekleri kullanılmıştır. Şekil 8’de yeni veri setinden örnekler yer almaktadır. Artırılan eğitim verisi belirlenen modellere verilirken epoch değeri 3, eğitim için batch_size 64, steps_per_epoch değeri 10, geçerleme için batch_size 10, geçerleme adımları 3 olarak tanımlanmıştır. Veri sayısı artırıldıktan sonra elde edilen sonuçlar Çizelge 4’te sunulmuştur.

**Şekil 8.** Gerçek veri setinden çatlak örnekleri (a), artırılmış veri setinden çatlak örnekleri (b), gerçek veri setinden çatlak olmayan örnekler (c), artırılmış veri setinden çatlak olmayan örnekler (d)**Çizelge 4.** Artırılmış veri ile eğitilen modellerin doğruluk değeri

No	Özelliklerin çıkarılması (VGG16)	Eğitim	Test	Geçerleme
IV.	block5_conv2	0,8797	1,0	1,0
V.	block2_conv2	0,9601	0,98	0,969
VI.	block1_conv1	0,9562	0,98	0,969

Çizelge 3 ve Çizelge 4'teki sonuçlar incelendiğinde veri sayısındaki artışın eğitim doğruluk değerini olumlu yönde etkilemesi beklenirken V numaralı deneyde olumlu yönde etkilediği VI numaralı deneyde ise etkisinin olmadığı görülmüştür. Hem eğitim hem de test verisine ait doğruluk değerleri incelendiğinde test doğruluğu değerinin eğitim doğruluğu değerinden büyük olduğu, dolayısıyla modelin ezberlemediği ve aşırı öğrenme denilen durumun olmadığı değerlendirilmiştir.

Çalışmanın devamında I, II ve III numaralı deneyler için model performansı "Model performansının değerlendirilmesi" başlığı altında açıklanan metrik parametreleri kullanılarak değerlendirilmiştir. Çizelge 5 ve Çizelge 6'da eğitime tüm katmanlar dahil edildiğinde ve artırılmış veri modele verildiğinde modelin sınıflandırma başarısının (ROC-AUC) artması ya da değişmemesi beklenirken III ve VI numaralı modelde düştüğü gözlemlenmiştir.

Çizelge 5. Veri sayısı artırılmadan önce model performansı. P: Precision, R: Recall, F: F1-Score, RA: ROC-AUC, CM: Confusion Matrix şeklinde kısaltılmıştır.

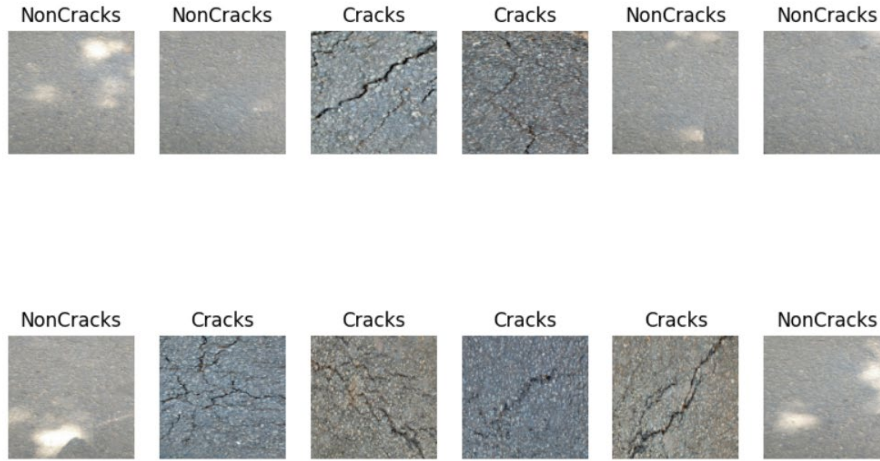
No	P	R	F	RA	CM
I.	1.0	1.0	1.0	1.0	[[20 0] [0 20]]
II.	1.0	1.0	1.0	1.0	[[20 0] [0 20]]
III.	1,0	0,95	0,97	0,975	[[20 0] [1 19]]

Çizelge 6. Veri sayısı artırıldıktan sonra model performansı. P: Precision, R: Recall, F: F1-Score, RA: ROC-AUC, CM: Confusion Matrix şeklinde kısaltılmıştır.

No	P	R	F	RA	CM
IV.	1.0	1.0	1.0	1.0	[[20 0] [0 20]]
V.	0.95	1,0	0,97	0,95	[[19 1] [0 20]]
VI.	0.95	1.0	0,97	0,95	[[19 1] [0 20]]

Test veri kümesi üzerinde model çalıştırıldığında çatlak ve çatlak olmayan yolların doğru olarak tespit edildiği gözlemlenmiştir. Modelin tahmin performansının başarılı örnekleri Şekil 9'da verilmektedir.

Çalışmanın son aşamasında, veri seti derin öğrenme modelini eğitmek için yetersiz sayıda görülebileceği için 5 katlı çapraz geçişleme uygulanmıştır. Gerçek veri her biri 80 örnek ve gerçek ile artırılmış verilerin toplandığı diğer küme her biri 128 parça içerecek şekilde rasgele 5 parçaya ayrılmıştır. Artırılmış veri seti parçalara bölünürken gerçek ve artırılmış verilerin dengeli bir şekilde (eşit sayıda çatlak içeren ve çatlak içermeyen görüntü) dağıldığı gözlemlenmiştir. Çizelge 2'de belirtilen 6 farklı deney 5 katlı çapraz geçişleme kullanılarak uygulanmıştır. Çizelge 6'da sunulan sonuçlar incelendiğinde en iyi doğruluk derecesinin 0,9688 test doğruluğu ile II. model tarafından elde edildiği görülmektedir. Tüm katmanların eğitildiği III. model de 0,9594 test doğruluğu ile yakın bir sonuca ulaşmıştır. Ortalama eğitim sürelerine bakıldığında eğitilen parametre sayısı 2 katına çıktığında eğitim süresinin neredeyse 3 katına çıktığı görülmüştür.



Şekil 9. Asfalt çatlaklarının tahmini

Çizelge 6. Veri sayısı artırılmadan önce 5 katlı çapraz geçerleme performansı ve ortalama eğitim süresi

No	Özelliklerin çıkarılması (VGG16)	Eğitim	Test	Ortalama eğitim süresi (sn)
I.	block5_conv2	0,9275	0,9156	616,2
II.	block2_conv2	0,9575	0,9688	1750,6
III.	block1_conv1	0,9719	0,9594	1768

Çizelge 7 incelendiğinde en iyi performansın 0,9656 test doğruluğu ile tüm katmanların eğitime katıldığı VI. model tarafından elde edildiği görülmektedir. Daha az katman ve parametre ile eğitilen IV ve V. modellerin sırasıyla 0,85 ve 0,932 test doğruluğu ile VI. modele kıyasla daha kötü performans elde ettiği gözlemlenmiştir. Şekil 8'e tekrar dönüldüğünde artırılmış verinin gerçek verinin değiştirilmiş hali olduğu görülmektedir. Kaydırma-döndürme-eğme işlemlerinden sonra boş kalan pikseller komşu piksellere benzetilerek doldurulmaktadır. İki veri grubu arasındaki farklılıkların ESA'nın görüntüden kaba özellikleri çıkaran ilk katman tarafından tespit edildiği, bu yüzden ilk katmanı eğitime dahil etmeyen IV ve V. modelin artırılmış veri ile yüksek başarı gösteremediği düşünülmektedir. Veri sayısı arttığı için eğitim süreleri de artmıştır.

Çizelge 7. Veri sayısı artırıldıktan sonra 5 katlı çapraz geçerleme performansı ve ortalama eğitim süresi

No	Özelliklerin çıkarılması (VGG16)	Eğitim	Test	Ortalama eğitim süresi
IV.	block5_conv2	0,884	0,85	998,2
V.	block2_conv2	0,93	0,932	2393,2
VI.	block1_conv1	0,968	0,9656	3050,2

3. Sonuç ve Tartışma

Ulaşım ağlarında en sık rastlanan sorunlardan biri asfalt çatlaklarıdır. Trafik yükü, malzeme özellikleri, çevresel etkenler, işçilik hatası gibi etkenlerden dolayı meydana gelen çatlaklar fark edilemediğinde zarar ilk başta küçük boyutlarda iken zamanla büyük boyutlara ulaşır. Büyük boyutlara ulaşan bu kusurlar, kazalara sebep olur, ayrıca sorun büyüdüğü için telafisi için daha fazla bakım maliyeti oluşturur. Kusurların fark edilmesini insan yetisine bırakmak ise yetersiz kalır. Bu nedenle yapmış olduğumuz bu çalışmada asfalt yol görüntüleri üzerindeki çatlakların derin öğrenme ile sınıflandırılması sağlanarak bilgisayarla görü alanındaki çalışmalara örnek sunulması amaçlanmıştır.

Oluşturmuş olduğumuz modelde iterasyon sayısı artırıldığında öğrenme süresinin uzadığı ancak başarımının arttığı gözlenmiştir. Oluşturulan derin öğrenme modelinde eğitilen katman sayısı artırıldığında modelin eğitim başarısı artarken tüm katmanların kullanıldığı modelde test başarısının düştüğü gözlemlenmiştir. Veri seti büyüklüğünün model başarımına etkisinin nasıl olduğu sorgulandığında ise çeşitli tekniklerle artırılan veri sayısının başarıma beklenildiği gibi olumlu bir katkıda bulunmadığı görülmüştür. Yapılan çalışmalarda test veri kümesinde %96 ve üzerinde doğruluk oranları yakalanmıştır.

Literatürde yer alan çalışmalar ve kendi çalışmamız değerlendirildiğinde görüntülerin tanınması ve sınıflandırılması konusunda derin öğrenme algoritmalarının [10-20] klasik makine öğrenmesi yöntemlerine [7-9] kıyasla oldukça başarılı olduğu görülmüştür. Makine öğrenmesi yöntemlerinden en başarılısı 0,965 ile DVM [7] önerdiğimiz modele en yakın doğruluk değerini elde etmiştir. Ancak verilerin ön işleme ve en iyi özelliklerin çıkarılması için çeşitli yöntemler uygulayarak bu başarıyı elde etmişlerdir. Derin öğrenme yöntemlerinin kullanıldığı çalışmalara baktığımızda 0,963 doğruluk ile veri artırımında GAN ve özellik çıkarımında geliştirilmiş VGG-16 kullanan modelin [14] en iyi performansı elde ettiğini görüyoruz. Bahse konu çalışmalarda farklı veri setleri kullanıldığı için sonuçlarını bu çalışma ile karşılaştırmak uygun bulunmamıştır. Aynı veri setini kullanan çalışmalar [18-20] incelendiğinde, en başarılı sonucu 0,997 doğruluk ile Balaji ve diğerlerinin [18] 448x448 çözünürlüklü görüntüler üzerine uyguladığı MobileNet modelinin elde ettiği görülmüştür. Aynı çalışmada, 224x224 çözünürlüklü görüntüler üzerinde 0,9525 doğruluk ile en iyi performansı ise Inception-v3 modeli elde etmiştir. Bu çalışmada ise VGG-16 ağı 224x224 boyutlu girdiler kabul ettiği için orta çözünürlüklü görüntüler kullanılmış ve daha yüksek başarımlar elde edilmiştir. Ayrıca, sonuçlar hem 5 katlı çapraz geçişleme hem de eğitim-geçerleme-test kümesi bölümlenmesi kullanılarak doğrulanmıştır.

Sonuç olarak, yol kusurlarının tespiti konusu ile birlikte birçok sektörde uzman kişilerin görüşü alınarak ihtiyaçların doğru bir şekilde belirlenmesi, gerçek verilerle çalışılması; ayrıca görüntü işleme teknikleri ile çeşitliliğinin artırılması ve bu etkenlere göre derin öğrenme algoritmaları hiper parametre ince ayarlarının yapılması ile oldukça faydalı sonuçlar elde edileceği değerlendirilmiştir.

Çıkar Çatışması Beyanı

Makale yazarları herhangi bir kurum, kuruluş, kişi ile kişisel ve finansal çıkar çatışması olmadığını beyan etmektedirler.

Kaynaklar

- [1] B. Kesayak, "Endüstri Tarihine Kısa Bir Yolculuk," <https://www.endustri40.com/endustri-tarihine-kisa-bir-yolculuk/>.
- [2] "Yol Ağı Bilgileri," <https://www.kgm.gov.tr/Sayfalar/KGM/SiteTr/Kurumsal/YolAgi>.
- [3] "Asfalt Yollarda Meydana Gelen Kusur ve Hasar Çeşitleri," <https://insapedia.com/asfalt-yollarda-meydana-gelen-kusur-ve-hasar-cesitleri>.
- [4] A. Erbaş and D. İlek, "Son dakika haberi... Karadeniz Sahil Yolu'nda 'çatlak' tehlikesi," <https://www.hurriyet.com.tr/gundem/yolda-catlak-tehlikesi-ekipler-hemen-harekete-gecti-41715204>, Rize, 2021.

- [5] A. Şeker, B. Diri, and H. H. Balık, “Derin Öğrenme Yöntemleri ve Uygulamaları Hakkında Bir İnceleme,” *Gazi Mühendislik Bilimleri Dergisi*, vol. 3, no. 3, pp. 47–64, 2017.
- [6] J. Chadwick, “Autonomous pothole-repairing robots will hit Britain’s streets by 2021,” <https://www.dailymail.co.uk/sciencetech/article-8883627/Autonomous-pothole-repairing-robots-hit-Britains-streets-2021.html>, 2020.
- [7] M. Baygın, O. Yaman, and T. Tuncer, “Akıllı Şehirler için Özellik Çıkarımı ve Makine Öğrenmesi Tabanlı Asfalt Durum İzleme Yaklaşımı,” *European Journal of Science and Technology*, Jan. 2021, doi: 10.31590/ejosat.844592.
- [8] L. de A. Schmidt, M. Pasin, and S. M. Peres, “Feasibility Study for an Automatic Architecture for Pothole Detection in Asphalt Images: a Trade-off between Performance and Quality,” in *2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP)*, IEEE, Sep. 2020, pp. 307–314. doi: 10.1109/ICCP51029.2020.9266229.
- [9] N. Shatnawi, “Automatic Pavement Cracks Detection using Image Processing Techniques and Neural Network,” *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 9, 2018, doi: 10.14569/IJACSA.2018.090950.
- [10] V. Mandal, L. Uong, and Y. Adu-Gyamfi, “Automated Road Crack Detection Using Deep Convolutional Neural Networks,” in *2018 IEEE International Conference on Big Data (Big Data)*, IEEE, Dec. 2018, pp. 5212–5215. doi: 10.1109/BigData.2018.8622327.
- [11] J. Cheng, W. Xiong, W. Chen, Y. Gu, and Y. Li, “Pixel-level Crack Detection using U-Net,” in *TENCON 2018 - 2018 IEEE Region 10 Conference*, IEEE, Oct. 2018, pp. 0462–0466. doi: 10.1109/TENCON.2018.8650059.
- [12] E. Deveci and B. Ergen, “Yol Yüzey Anormalisinin Tespiti ve Analizi,” *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, vol. 10, no. 3, pp. 1187–1194, Sep. 2021, doi: 10.17798/bitlisfen.942386.
- [13] F. Liu, J. Liu, and L. Wang, “Asphalt Pavement Crack Detection Based on Convolutional Neural Network and Infrared Thermography,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 22145–22155, Nov. 2022, doi: 10.1109/TITS.2022.3142393.
- [14] Y. Que *et al.*, “Automatic classification of asphalt pavement cracks using a novel integrated generative adversarial networks and improved VGG model,” *Eng Struct*, vol. 277, p. 115406, Feb. 2023, doi: 10.1016/j.engstruct.2022.115406.
- [15] S. Matarneh, F. Elghaish, F. Pour Rahimian, E. Abdellatef, and S. Abrishami, “Evaluation and optimisation of pre-trained CNN models for asphalt pavement crack detection and classification,” *Autom Constr*, vol. 160, p. 105297, Apr. 2024, doi: 10.1016/j.autcon.2024.105297.
- [16] M. Yılmaz *et al.*, “Improving the Classification Performance of Asphalt Cracks After Earthquake With a New Feature Selection Algorithm,” *IEEE Access*, vol. 12, pp. 6604–6614, 2024, doi: 10.1109/ACCESS.2023.3343619.
- [17] A. Jayanth Balaji, G. Thiru Balaji, M. S. Dinesh, N. Binoy, and D. S. Harish Ram, “Asphalt Crack Dataset,” Available: <https://data.mendeley.com/datasets/xnzjh3x8v4/2>, vol. 2. 2019.
- [18] A. J. Balaji, G. Thiru Balaji, M. S. Dinesh, B. B. Nair, and D. S. Harish Ram, “A Machine Learning Based Approach to Crack Detection in Asphalt Pavements,” in *2018 15th IEEE India Council International Conference (INDICON)*, IEEE, Dec. 2018, pp. 1–4. doi: 10.1109/INDICON45594.2018.8987039.
- [19] B. Ji, “A Study on Crack Detection in Asphalt Road Pavement Using Small Deep Learning,” *Journal of the Korean GEO-environmental Society*, vol. 22, no. 10, pp. 13–19, 2021.
- [20] A. Kumar, A. Kumar, A. K. Jha, and A. Trivedi, “Crack Detection of Structures using Deep Learning Framework,” in *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, IEEE, Dec. 2020, pp. 526–533. doi: 10.1109/ICISS49785.2020.9315949.
- [21] “ImageNet,” <https://www.image-net.org/about.php>.
- [22] “VGG16 and VGG19,” <https://keras.io/api/applications/vgg/>.
- [23] T. Fushiki, “Estimation of prediction error by using K-fold cross-validation,” *Stat Comput*, vol. 21, no. 2, pp. 137–146, Apr. 2011, doi: 10.1007/s11222-009-9153-8.

- [24] C. Shorten and T. M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *J Big Data*, vol. 6, no. 1, p. 60, Dec. 2019, doi: 10.1186/s40537-019-0197-0.
- [25] M. Toğaçar, K. A. Eşidir, and B. Ergen, “Yapay Zekâ Tabanlı Doğal Dil İşleme Yaklaşımını Kullanarak İnternet Ortamında Yayınlanmış Sahte Haberlerin Tespiti,” *Journal of Intelligent Systems: Theory and Applications*, vol. 5, no. 1, pp. 1–8, Oct. 2021, doi: 10.38016/jista.950713.
- [26] Ö. İnik and E. Ülker, “Derin Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri ,” *GAZİOSMANPAŞA BİLİMSEL ARAŞTIRMA DERGİSİ (GBAD)*, vol. 6, no. 3, pp. 85–104, 2017.
- [27] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.
- [28] K. Fırıldak and M. F. Talu, “Evrşimsel Sinir Ağlarında Kullanılan Transfer Öğrenme Yaklaşımlarının İncelenmesi,” *Computer Science*, vol. 4, no. 2, pp. 88–95, 2019.
- [29] J. Xu, Z. Li, B. Du, M. Zhang, and J. Liu, “Reluplex made more practical: Leaky ReLU,” in *2020 IEEE Symposium on Computers and Communications (ISCC)*, IEEE, Jul. 2020, pp. 1–7. doi: 10.1109/ISCC50000.2020.9219587.
- [30] U. Ruby, P. Theerthagiri, I. J. Jacob, and Y. Vamsidhar, “Binary cross entropy with deep learning technique for Image classification,” *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 4, pp. 5393–5397, Aug. 2020, doi: 10.30534/ijatcse/2020/175942020.
- [31] M. A. Koşan, A. Coşkun, and H. Karacan, “Yapay Zekâ Yöntemlerinde Entropi,” *Journal of Information Systems and Management Research*, vol. 1, no. 1, pp. 15–22, 2019.
- [32] G. Ser and C. T. Bati, “Derin Sinir Ağları ile En İyi Modelin Belirlenmesi: Mantar Verileri Üzerine Keras Uygulaması,” *Yüzüncü Yıl Üniversitesi Tarım Bilimleri Dergisi*, vol. 29, no. 3, pp. 406–417, Sep. 2019, doi: 10.29133/yyutbd.505086.
- [33] R. V. Kumar Reddy, B. Srinivasa Rao, and K. P. Raju, “Handwritten Hindi Digits Recognition Using Convolutional Neural Network with RMSprop Optimization,” in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, IEEE, Jun. 2018, pp. 45–51. doi: 10.1109/ICCONS.2018.8662969.
- [34] H. M and S. M.N, “A Review on Evaluation Metrics for Data Classification Evaluations,” *International Journal of Data Mining & Knowledge Management Process*, vol. 5, no. 2, pp. 01–11, Mar. 2015, doi: 10.5121/ijdkp.2015.5201.
- [35] E. ERGÜN and K. KILIÇ, “Derin Öğrenme ile Artırılmış Görüntü Seti üzerinden Cilt Kanseri Tespiti,” *Black Sea Journal of Engineering and Science*, vol. 4, no. 4, pp. 192–200, Oct. 2021, doi: 10.34248/bsengineering.938520.