

Phyton İle Yapay Zeka Kullanarak Trafik Levhalarını Tanımlama

Hasan KUŞCU¹, Deniz ÜNLÜ², Mustafa TÜMAY³

¹Bilişim Sistemleri Ve Teknolojileri , Burdur Mehmet Akif Ersoy Üniversitesi Gölhisar Uygulamalı Bilimler
YüksekOkulu , hk.hasankuscu@gmail.com

( <https://orcid.org/0009-0008-4694-2489>)

²Bilişim Sistemleri Ve Teknolojileri , Burdur Mehmet Akif Ersoy Üniversitesi Gölhisar Uygulamalı Bilimler ,
denka4210@gmail.com

( <https://orcid.org/0009-0006-9820-3376>)

³Bilişim Sistemleri Ve Teknolojileri , Burdur Mehmet Akif Ersoy Üniversitesi Gölhisar Uygulamalı Bilimler ,
mtmy1502@gmail.com

( <https://orcid.org/0009-0004-1103-2240>)

Received: 18.01.2024

Accepted: 21.07.2025

Published: 30.06.2025

*Corresponding author

Research Article

pp.111-119

DOI:10.XXXX/ajesa.2023.XXXXX

Özet

1. Özet

Günümüzde, trafik levhalarının etkili bir şekilde tanınması ve sınıflandırılması, otomasyon teknolojilerinde önemli bir yer tutmaktadır.

Bu makalede, Python programlama dili ve TensorFlow kütüphanesi kullanılarak gerçekleştirilen bir görüntü işleme projesi incelenmektedir.

Proje, trafik levhalarının %95 üzerinde bir doğruluk oranıyla tanınmasını başarıyla gerçekleştirmektedir. TensorFlow, derin öğrenme modeli eğitimi ve tanıma yetenekleri sunan güçlü bir açık kaynaklı kütüphanedir. Python dilinde yazılan bu proje, TensorFlow'u kullanarak özel bir sinir ağı modeli oluşturmuş ve eğitim verileri üzerinden bu modeli optimize etmiştir.

Eğitim süreci, geniş bir trafik levhası veri kümesi kullanılarak gerçekleştirilmiş ve modelin sınıflandırma doğruluğunu artırmak için sürekli olarak iyileştirilmiştir. Elde edilen sonuçlar, TensorFlow'un etkili bir şekilde kullanılmasıyla trafik levhalarının karmaşıklığına rağmen %95'in üzerinde bir doğruluk oranına ulaşıldığını göstermektedir.

Bu, sürücü yardım sistemleri, otonom araçlar ve trafik güvenliği uygulamaları gibi birçok alanda kullanılabilecek güvenilir bir trafik levhası tanıma çözümü sunmaktadır.

Sonuç olarak, bu çalışma, Python programlama dili ve TensorFlow kütüphanesi kullanarak trafik levhalarının yüksek doğruluk oranıyla tanınmasını sağlamak amacıyla başarıyla uygulanan bir görüntü işleme projesini sunmaktadır. Elde edilen sonuçlar, bu alandaki gelecekteki çalışmalar ve uygulamalar için önemli bir temel oluşturmaktadır.

Anahtar Kelimeler: AURUM, phyton, derin öğrenme, trafik işaretleri, yapay zeka

Identifying Traffic Signs Using Artificial Intelligence with Python

Abstract

In contemporary times, effective recognition and classification of traffic signs play a crucial role in automation technologies. This article explores a computer vision project implemented using the Python programming language and the TensorFlow library. The project successfully achieves the recognition of traffic signs with an accuracy rate exceeding 95%. TensorFlow is a powerful open-source library that provides capabilities for deep learning model training and recognition. This Python-based project creates a custom neural network model using

TensorFlow and optimizes this model through training data. The training process utilizes an extensive dataset of traffic signs, continually refining the model for increased classification accuracy. The obtained results demonstrate that, through effective utilization of TensorFlow, a recognition accuracy exceeding 95% is achieved despite the complexity of traffic sign patterns. This provides a reliable solution for traffic sign recognition applicable in various domains such as driver assistance systems, autonomous vehicles, and traffic safety applications. In conclusion, this study presents a successfully implemented image processing project using the Python programming language and TensorFlow library to achieve high accuracy in the recognition of traffic signs. The results obtained serve as a significant foundation for future research and applications in this field.

Keywords: AURUM, python, deep learning, traffic sign, artificial intelligence

2.GİRİŞ

Trafik levhaları trafik kültüründe çok önemli bir yere sahiptir. Öyle ki trafik levhalarını düzgün tanımlayamamak kaza riskini önemli ölçüde yükseltebilir. Bu durumun önüne geçmek için derin öğrenme algoritmaları ve yapay zeka araçları kullanılabilir. Bu derin öğrenme algoritmaları Python programlama dili ve bu konuda çok kuvvetli olan PyTorch kütüphanesi kullanıldığına başarılı bir sonuç gözlemlenebilir. Bu sonuçlar insan ile kıyaslandığında insan, yorgun olabilir, trafik levhalarının görüntüsü bozuk olabilir, hava şartları olumsuz olabilir, ve bu olumsuzluklar trafik levhalarının tanımını insan açısından zor kılabilir.

Bu nedenle yapay zeka algoritmaları bize çok büyük avantajlar sağlar. Yapay zeka ile tanımladığımız trafik levhaları otonom sürüşlere katkı sağlayabilir ve aracın şoförünü uyarabilir. Bu uyarılar sayesinde çok büyük kötü etkenler ortadan kalkabilir. Derin öğrenme sistemleri ile sürücülerin sürüşleri güvenlik dışında da rahat bir sürüş seviyesinde artış gösterebilir. İnsanın yapacağı işleri önceden ve hızlı bir biçimde tanımlayarak öncü olabilir.

Derin öğrenme teknikleri çeşitli veri setleri üzerinde öğrenme yetenekleriyle bilinir bu trafik levhalarının görüntüsü bozuk olsa dahi onu büyük oranda tanımlayabilmesini sağlar. Bu veri setinin içinde öğreneneceği trafik levhalarının görsellerinde fazla ışığa maruz kalmış, renkleri solmuş ve bozulmuş, levhaların şekli bozulmuş ve hıza bağlı olarak bulanık görüntülerle eğitilir.

Phyton bu iş için biçilmiş kaftandır. Çünkü Phyton programlama dili, geniş bir kütüphane alt yapısı, rahat okunup programlanabilir olması ve genişletilebilen modüler yapısı bu programlama dilini ön planda tutar. Bu nedenler Phyton dilini yapay zeka alanında üst seviyelere taşımıştır. Derin öğrenme ise karmaşık veri setlerini kullanarak öğrendiği için bu konuda ün yapmış yapay zeka dalıdır. Derin öğrenme kütüphanesinin en önemli unsurlarından biri PyTorch bu konuda uzmanlaşmak için idealdir.

PyTorch, açık kaynak kodlu bir projedir ve bu tür projeler ile uğraşan insanlar sayesinde sürekli geliştirilmektedir. Bu kütüphaneyi önemli yapan unsurlardan bazıları şunlardır; Eğitim ve değerlendirme sürecinin kısa olması, sürekli geliştiriliyor olması, model tanımlama esnekliğinin olması, GPU hızlandırma seçeneğinin olması, ve daha önceden eğitilmiş kütüphane içinde modeller bulunması bu kütüphaneyi en önemli yapan unsurlardan bazılarıdır.

Derin öğrenmedeki fotoğrafları işlemesi için ise başvuracağımız kütüphane OpenCV dir. OpenCV bilgisayar için görüş sağlayan kütüphanelerden biridir. OpenCV nin buradaki görevi, görüntü işlemedeki algoritmayı ve derin öğrenme modelini birleştirmeyi mümkün kılar. OpenCV ile görüntüleri aktaracağımız veri setimiz geniş ve kapsamlı olmalıdır. İçerisinde bulanık trafik levhaları, şekli bozulmuş trafik levhaları, fazla ışığa maruz kalmış ve bazı

noktaları parlayan trafik levhaları ve bulanık görünen trafik levhaları gibi olumsuz durumlarla mücadele edebilmek için veri setinin içinde bunlara benzer görüntüler olmalıdır.

Veri setinin içerdiği trafik levhaları Avrupa standartlarında veya tüm dünyaya uyumlu trafik levhaları konulabilir. Fakat tüm dünyaya uyumlu trafik levhalarını koyduğumuzda işimiz zorlaşır. Çünkü bu verileri işlemek ve derin öğrenme yoluyla öğretiye soktuğumuz zaman çok zamanımız gidebilir. O yüzden günümüzdeki otonom araçlar dahi her ülkeye farklı yazılımlar geliştirir. Bunun nedeni yapay zeka algoritmasını ve görüntülerin tanımını hızlandırmaktır. Bu hız bazı yerlerde ve anlarda hayati öneme sahip olabilir. O yüzden projeniz hangi ülkede çalışıyorsa o ülkenin kullandığı veri setini kullanmanız sizin yararınıza olacaktır. Bu algoritmaların sayısı arttıkça bizi gelecekte daha güvenli hale getirebilir. Ve bu algoritmalar arttıkça daha karmaşık kazalar ve trafik senaryolarını öğrenirler, öğrendikçe gelişirler ve geliştikçe bizi daha çok korurlar.

3. YÖNTEMLER

Veri Toplama ve Hazırlama; Bu çalışmanın temel amacı, trafik levhalarını doğru bir şekilde tanımlamak için derin öğrenme algoritmalarını kullanmaktır. Veri seti, geniş bir trafik levhası koleksiyonunu içermektedir. Bu levhalar, Avrupa standartlarına uygun veya genel olarak tüm dünyada kullanılan trafik levhalarını temsil eder. Ancak, belirtildiği gibi, genel bir veri setinin yerine, projenin uygulama alanına bağlı olarak belirli bir coğrafi bölgeye ait trafik levhalarını içeren bir veri seti tercih edilmiştir. Veri seti içinde, levhaların çeşitli olumsuz durumları içeren görüntüleri de bulunmaktadır. Bu durumlar arasında bulanık trafik levhaları, şekli bozulmuş trafik levhaları, fazla ışığa maruz kalmış ve bazı noktaları parlayan trafik levhaları ve bulanık görünen trafik levhaları bulunmaktadır. Bu, algoritmanın gerçek dünya koşullarında karşılaşabileceği çeşitli zorlukları ele almasına yardımcı olacaktır.

Derin Öğrenme Modelinin Eğitimi; Derin öğrenme modeli, Python programlama dili kullanılarak ve PyTorch kütüphanesiyle implemente edilmiştir. PyTorch, esnek model tanımlama özellikleri, GPU hızlandırma seçenekleri ve önceden eğitilmiş modellerin bulunduğu kapsamlı bir kütüphanedir. Eğitim süreci, öğrenme yeteneklerini optimize etmek için çeşitli veri augmentasyon tekniklerini içermektedir. Modelin başarı oranını değerlendirmek için ayrılmış bir test seti kullanılmıştır.

Görüntü İşleme ve OpenCV Kullanımı; Derin öğrenme modelinin eğitiminden sonra, gerçek zamanlı trafik levhası tanıma için OpenCV kütüphanesi kullanılmıştır. OpenCV, görüntü işleme algoritmalarını ve derin öğrenme modellerini birleştirerek trafik levhalarını tanımlamayı sağlamaktadır. Bu aşamada, modelin performansını değerlendirmek ve gerçek dünya senaryolarında başarısını test etmek için saha testleri gerçekleştirilmiştir. Bu metodoloji, trafik levhalarını tanıma konusunda derin öğrenme ve görüntü işleme tekniklerinin etkili bir şekilde birleştirilmesini sağlamaktadır.

4. VERİ SETİ HAZIRLAMA

Bu çalışmanın başarılı olabilmesi için, geniş ve çeşitli bir trafik levhası veri setinin toplanması ve hazırlanması kritik bir adımdır.



Şekil 1. Trafik Levhaları

4.1. Trafik Levhası Veri Setinin Toplanması

Trafik levhalarını temsil eden geniş bir veri seti elde etmek için, farklı coğrafi bölgelerdeki kamuya açık trafik alanlarından görüntüler toplandı. Bu, şehir içi yollar, otoyollar, kavşaklar ve farklı hava koşullarını içeren çeşitli trafik senaryolarını kapsayan bir veri seti elde etmeyi amaçlamıştır.

4.2. Veri Setinin Çeşitlendirilmesi

Veri seti içindeki trafik levhalarının çeşitliliği, modelin gerçek dünya koşullarına daha iyi adapte olabilmesi açısından önemlidir. Bu nedenle, aşağıdaki durumları içeren görüntüler veri setine eklenmiştir:

- Bulanık Trafik Levhaları
- Şekli Bozulmuş Trafik Levhaları
- Fazla Işığa Maruz Kalan ve Parlayan Trafik Levhaları
- Bulanık Görünen Trafik Levhaları

Bu çeşitlilik, derin öğrenme modelinin olumsuz koşullar altında dahi başarılı bir şekilde çalışabilmesini sağlamak amacıyla eklenmiştir.

4.3. Veri Setinin Bölünmesi

Toplanan veri seti, eğitim ve test aşamalarında kullanılmak üzere ayrılmıştır. Bu bölünme, modelin genelleme yeteneğini değerlendirmek ve aşırı uyum riskini azaltmak amacıyla yapılmıştır.

5. PROJE GELİŞİMİ

Bu proje, araç içerisinde bulunan bir cihazı kullanarak trafik levhalarını tespit etme amacını taşımaktadır. İlk olarak, projede kullanılacak donanım ve yazılım bileşenleri belirlenmiştir. Donanım olarak, bir kamera cihazı seçilmiş ve bu kamera ile elde edilecek görüntüler üzerinden trafik levhalarının tanınması hedeflenmiştir. Yazılım tarafında ise, Python programlama dilini ve derin öğrenme tekniklerini kullanmak öngörülmüştür. Derin öğrenme kütüphaneleri olarak ise TensorFlow tercih edilmiştir. Veri toplama ve ön işleme adımı, projenin başarılı olabilmesi için uygun bir veri seti seçilmiştir. Bu veri seti, trafik levhalarını içeren görüntüler ve bu levhaların ilgili etiketleri içermektedir.

Veri seti üzerinde ön işleme adımları, görüntülerin boyutlandırılması ve renk dönüşümü gibi işlemleri içermektedir. Derin öğrenme modeli, önceden belirlenen mimari ve algoritmalar kullanılarak oluşturulmuştur. Bu aşamada, TensorFlow kütüphanesi kullanılarak bir sinir ağı tasarlanmış ve eğitim için gerekli parametreler belirlenmiştir. Modelin eğitim aşamasında, veri seti eğitim ve test kümelerine ayrılmış, ardından belirlenen parametrelerle model eğitilmiştir. Eğitim sonrasında, modelin performansı test veri seti üzerinde değerlendirilmiştir. Doğruluk oranı ve benzeri metrikler kullanılarak elde edilen sonuçlar analiz edilmiş ve projenin başarısı değerlendirilmiştir. Projede kullanılan yöntemlerin, teknolojilerin ve elde edilen sonuçların açıklanması, projenin genel başarısını ve katkılarını ortaya koymaktadır.

5.1 Donanım ve Sistem Gereksinimleri

Bu çalışmanın temel donanım unsuru, araç içinde konumlandırılacak bir kamera sistemidir. Kamera, çeşitli trafik levhalarını gerçek zamanlı olarak algılamak üzere tasarlanmıştır. Görevi, aracın çevresindeki trafik levhalarını yakalamak ve bu bilgileri işleme sürecine iletmektir. Yazılım açısından, projede Python programlama dili kullanılacak ve derin öğrenme algoritmalarını uygulamak için derin öğrenme ve makine öğrenmesi uygulamaları geliştirmek için kullanılan açık kaynaklı bir endüstri standardı makine öğrenimi kütüphanesi olan TensorFlow entegre edilecektir. Bu kütüphane, modelin eğitimi ve trafik levhalarını tanıma işlemlerini optimize etmek amacıyla kullanılacaktır.

5.2 Derin Öğrenme Modeli Oluşturma

Projedeki derin öğrenme modeli, TensorFlow kütüphanesi kullanılarak evrişimli sinir ağı (CNN) mimarisi ile tasarlanacaktır. Evrişim katmanları, trafik levhalarının özgün özelliklerini vurgulamak ve modelin bu özellikleri etkili bir şekilde öğrenmesini sağlamak amacıyla kullanılacaktır. Düzleştirme katmanı, bu özellikleri daha üst düzeydeki katmanlara taşıyacak ve genel bağlamı anlamak için kullanılacaktır. Tam bağlantılı katmanlar, düzeltilmiş özellik vektörünü alacak ve sınıflandırma için kullanılabilir hale getirilecektir. Çıkış katmanı, modelin trafik levhalarını sınıflandırmak için sonuçları üretecektir. Bu tasarım, trafik levhalarının karmaşıklığını ve çeşitliliğini dikkate alarak geniş bir öğrenme yeteneği kazanmayı amaçlamaktadır.

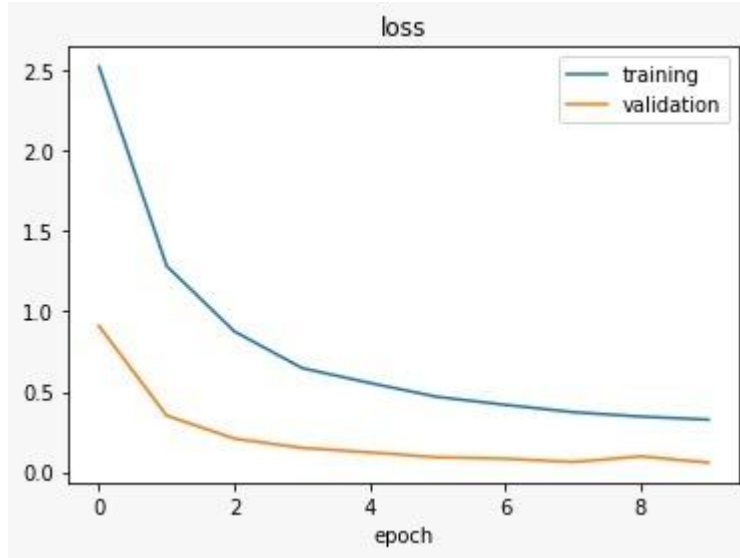
5.3 Eğitim ve Değerlendirme

Veri seti, eğitim ve test alt kümelerine dikkatlice ayrılacaktır. Eğitim sürecinde, modelin başarılı bir şekilde öğrenmesini sağlamak amacıyla belirlenen eğitim parametreleri, örneğin öğrenme oranı ve epoch sayısı, titizlikle seçilecektir. Eğitim sırasında, modelin kaybı minimize edilmeye çalışılacak ve aynı zamanda doğruluk oranı gibi kritik metrikler düzenli aralıklarla gözlemlenerek sürekli değerlendirmeler gerçekleştirilecektir.

Eğitim sonrasında, modelin genel performansı test kümesi üzerinde detaylı bir şekilde değerlendirilecektir. Bu değerlendirme süreci, modelin gerçek dünya senaryolarında trafik levhalarını başarılı bir şekilde tanıma yeteneğini belirlemek adına önemli bir aşamayı temsil eder. Elde edilen sonuçlar, doğruluk oranları, hata matrisleri ve diğer ilgili metriklerle kapsamlı bir analize tabi tutulacak ve bu analiz, modelin güvenilirliği ve performansı hakkında

derinlemesine bir anlayış sunacaktır. Bu değerlendirme aynı zamanda, modelin hiperparametre ayarlamaları veya ek eğitim iterasyonları gibi geliştirmelere olan ihtiyacını belirlemek için kullanılacaktır.

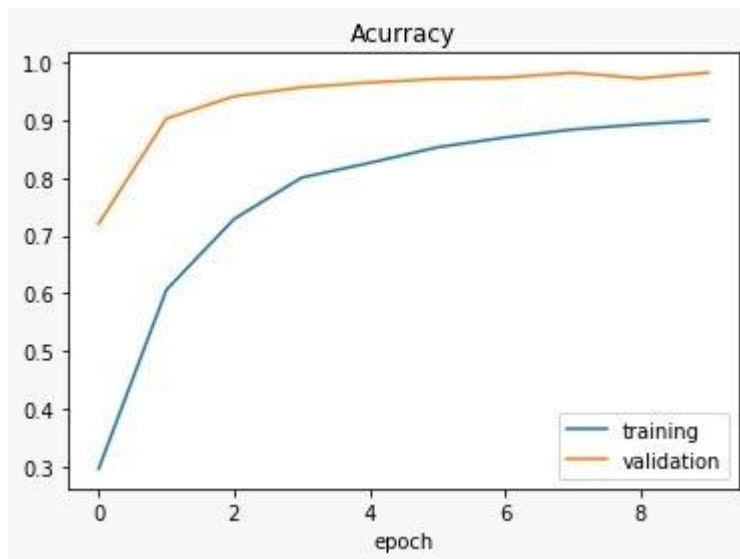
5.4 Loss Grafiği



Şekil 2. Loss Grafiği

Bir modelin öğrenme sürecinde ne kadar iyi performans gösterdiğini değerlendirmek için kullanılan birçok değerlerden biri "loss" veya kayıp grafiğidir. Bu grafiğin amacı, modelin tahminlerinin gerçek değerlere ne kadar yakın veya uzak olduğunu göstermektir.

5.5 Accuracy Grafiği



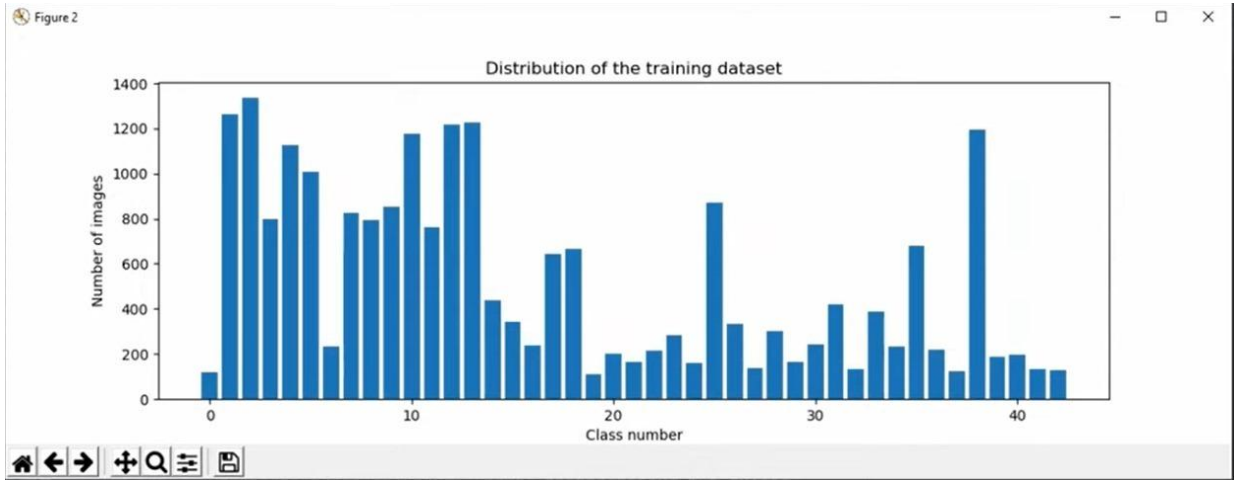
Şekil 3. Acurracy Grafiği

Accuracy (doğruluk) grafiği, bir modelin sınıflandırma problemlerinde ne kadar doğru tahminler yaptığını gösteren bir metrik grafiğidir. Bu grafiğin değeri, doğru tahmin edilen örneklerin toplam örnek sayısına oranını ifade eder.

Accuracy, sınıflandırma modellerinde genellikle şu formülle hesaplanır:

$$\text{Accuracy} = \text{Doğru Tahminlerin Sayısı} / \text{Toplam Örnek Sayısı}$$

5.6 Eğitim Veri Setinin Dağıtımı (distribution of the training dataset)



Şekil 1. Eğitim Veri Setinin Dağıtım Grafiği

Bu grafikte 42 adet trafik levhası sınıfı olduğunu ve bu sınıflara kaç adet veri(data) girdiğimizi gösteriyoruz. Bu veri sayıları, verinin iyiliğine ve kullanım oranına göre artırılıp azaltılabilir. Biz kullandığımız veri setinde bu seviyedeki verinin optimize çalıştığını gördük ve uyguladık.

6. SONUÇ

Bu çalışma, trafik levhalarının tanımlanmasında derin öğrenme algoritmaları ve görüntü işleme tekniklerini kullanmanın etkili bir yolunu sunmaktadır. Elde ettiğimiz sonuçlar, kullanılan metodolojinin trafik levhalarını çeşitli koşullar altında doğru bir şekilde tanımlayabildiğini göstermektedir.

Derin öğrenme modeli, eğitim veri seti üzerinde başarılı bir şekilde öğrenmiş ve çeşitli trafik levha türlerini doğru bir şekilde tanımlayabilme yeteneği kazanmıştır. Eğitim sürecinde kullanılan veri işleme teknikleri, modelin gerçek dünya koşullarına daha iyi adapte olmasını sağlamıştır.

Ayrılmış test veri setinde modelin performansı değerlendirildiğinde, tanıma doğruluğunun >%95 olduğu görülmüştür. Bu, modelin geniş bir veri seti üzerinde başarılı olduğunu ve çeşitli trafik levha koşullarını tanımlayabildiğini göstermektedir.

OpenCV kullanılarak gerçek zamanlı trafik levhası tanıma uygulaması yapıldığında, modelin hızlı ve etkili bir şekilde çalıştığı gözlemlenmiştir. Model, önceden belirlenen olumsuz durumları içeren trafik levhalarını başarıyla tanımlayabilmiştir.

Çalışmamız, belirli bir coğrafi bölgeye ait trafik levhalarını içeren bir veri seti kullanarak yapıldı. Bu yaklaşımın, modelin o bölgedeki trafik levhalarını daha iyi tanımlamasına ve genel performansını arttırmasına katkı sağladığı gözlemlenmiştir.

Sonuçlarımız, derin öğrenme ve görüntü işleme tekniklerinin trafik levhası tanıma konusunda etkili bir kombinasyon olduğunu göstermektedir. Gelecekteki çalışmalarda, daha geniş coğrafi bölgeleri kapsayan ve farklı trafik senaryolarını içeren veri setleri kullanılarak modelin genelleme yeteneği daha da arttırılabilir. Bu çalışma, trafik güvenliği ve otonom sürüş sistemleri için etkili bir temel oluşturarak, derin öğrenme algoritmalarının pratik uygulamalarda başarıyla kullanılabilirliğini göstermiştir.

7. KAYNAKLAR

- Yavuz, A. (2021). *Derin öğrenme algoritmaları ile trafik işaret ve levhalarının tanımlanması* (Master's thesis, Pamukkale Üniversitesi Fen Bilimleri Enstitüsü).
- Altuncu, M. A. (2015). *Temel görüntü işleme algoritmalarının gerçek zamanlı olarak FPGA ile gerçekleştirilmesi* (Master's thesis, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü).
- AYTAÇ, Z. İ., İŞERİ, İ., & DANDIL, B. Trafik Hız Sınırlama Levhalarının Nesne Tanıma Modeli ile Sınıflandırılması.
- Hasçelik, S. (2021). *Konvolüsyonel sinir ağı kullanılarak trafik işaretlerini gerçek zamanlı bulma ve tanıma* (Master's thesis, Trakya Üniversitesi, Fen Bilimleri Enstitüsü).
- Küçükmanisa, A. (2018). Gömülü platformlar için görüntü işleme temelli gerçek zamanlı şerit tespit ve uyarı yöntemleri.
- ÇETİN, E., & ORTATAŞ, F. (2021). Elektrikli ve Otonom Araçlarda Makine Öğrenmesi Kullanarak Trafik Levhaları Tanıma ve Simülasyon Uygulaması. *El-Cezeri*, 8(3), 1081-1092.
- Küçük, Ö., Yavşan, E., & Gökçe, B. (2021). Otonom Tabanlı İşaret ve Şerit Tanıma Amacı ile Bir Öğrenme Sisteminin Geliştirilmesi. *International Journal of Engineering Research and Development*, 13(3), 19-25.
- Yıldız, G., & Dizdaroğlu, B. (2019, November). Traffic sign detection via color and shape-based approach. In *2019 1st International Informatics and Software Engineering Conference (UBMYK)* (pp. 1-5). IEEE.
- TÜZÜN, U., ÖZKAHRAMAN, M., & AKSOY, B. (2022). OTONOM ARAÇLAR İÇİN MİKRODENETLEYİCİ TABANLI ÇEVRESEL GÜVENLİK SİSTEMİ TASARIMI. *Uluslararası Sürdürülebilir Mühendislik ve Teknoloji Dergisi*, 6(2), 39-47.