



Real time classification of traffic sign with deep learning methods

Kemal Usanmaz* , Selda Güney 

Department of Electrical and Electronics Engineering, Faculty of Engineering, Başkent University, 06790, Ankara, Türkiye

Highlights:

- Classification of traffic signs with convolutional neural network
- Design of real time test system
- Finding optimum hyperparameters of convolutional neural network

Keywords:

- Deep learning
- Convolutional neural network
- Traffic sign classification
- Transfer learning
- Hyperparameter

Graphical/Tabular Abstract

In the study, pre-trained Convolutional Neural Networks were used to classify traffic signs in real time. Hyperparameters were found for which pre-trained networks performed best. The best performing algorithm was determined among the AlexNet, GoogleNet, ResNet50 and SqueezeNet models that achieved the best results with the designed real-time user interface. According to the test results, ResNet50 achieved an accuracy rate of 93.49%. The architecture of the developed systems is shown in Figure A.

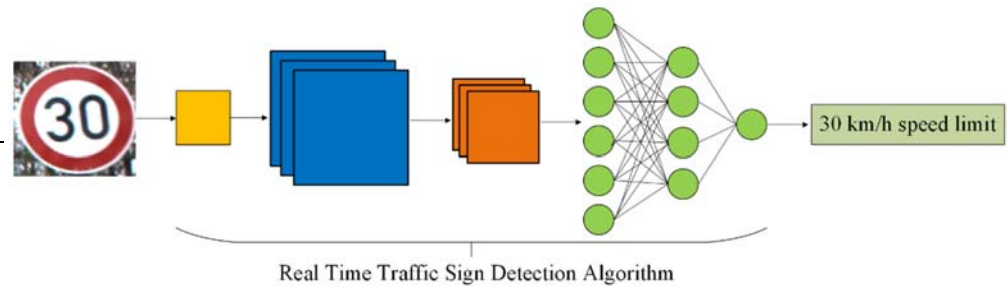


Figure A. Traffic sign detection with convolutional neural networks

Article Info:

Research Article

Received: 08.01.2024

Accepted: 07.10.2024

DOI:

10.17341/gazimmfd.1416186

Correspondence:

Author: Kemal Usanmaz

e-mail:

kemalusanmaz1996@gmail.com

phone: +90 534 365 8593

Purpose: This study focuses on designing and implementing a system capable of classifying traffic signs with high accuracy in real-time scenarios. By leveraging pre-trained Convolutional Neural Networks (CNNs) such as SqueezeNet, GoogleNet, ResNet50, and AlexNet, the research explores the efficacy of the transfer learning method applied to the German Traffic Sign Recognition Benchmark (GTSRB) dataset. The primary objective is to identify the optimal network model and hyperparameter settings that maximize classification performance. The study enhances robustness by addressing challenges from noisy and imbalanced datasets, incorporating artificial noise, and ensuring balanced class distributions.

Theory and Methods: To achieve effective real-time classification of traffic signs, this research adopts a systematic methodology. The study begins with the selection of pre-trained CNN models, namely AlexNet, GoogleNet, ResNet50, and SqueezeNet. Key hyperparameters, such as learning rate, batch size, number of epochs, and optimizer type, were systematically tuned for each network to identify the configurations that yield the best performance. To enhance the robustness of the system, data augmentation techniques were implemented. It simulates real-world challenges, improving the model's ability to generalize across diverse scenarios. A user interface was developed to facilitate the integration of the trained models into a real-time system.

Results: The evaluation of the developed system revealed that ResNet50 outperformed other pre-trained models, achieving an impressive accuracy rate of 93.49% on the test dataset. The addition of noise to the training data significantly improved the system's resilience, particularly in challenging scenarios involving blurred or partially obscured signs.

Conclusion: This research demonstrates the potential of pre-trained CNN models for real-time traffic sign classification. By employing transfer learning and rigorous data augmentation, the study addresses common challenges such as noisy and imbalanced datasets. The real-time user interface effectively integrates the trained models, enabling the deployment of a robust and high-performing system for real-world traffic sign recognition tasks. Future studies may explore further optimizations, including the use of ensemble methods or lightweight architectures for deployment on resource-constrained devices.



Derin öğrenme yöntemleriyle trafik işaretlerinin gerçek zamanlı sınıflandırılması

Kemal Usanmaz*^{ID}, Selda Güney^{ID}

Başkent Üniversitesi, Mühendislik Fakültesi, Elektrik Elektronik Mühendisliği Bölümü, 06790, Ankara, Türkiye

Ö N E Ç I K A N L A R

- Konvolüsyonel sinir ağları ile trafik işaretlerinin sınıflandırılması
- Gerçek zamanlı test sistemi tasarımı
- Konvolüsyonel sinir ağlarının optimum hiper parametrelerinin bulunması

Makale Bilgileri

Araştırma Makalesi
Geliş: 08.01.2024
Kabul: 07.10.2024

DOI:

10.17341/gazimmfd.1416186

Anahtar Kelimeler:

Derin öğrenme,
konvolüsyonel sinir ağı,
trafik işareti sınıflandırma,
transfer öğrenimi,
hiper parametre

ÖZ

Günümüzde trafik güvenliği, gelişmiş sürücü destek sistemlerini (ADAS) önemli ölçüde etkilenmektedir. Sürücüler için önemli bilgiler taşıyan trafik işaretleri çoğu zaman sürücüler tarafından fark edilemez ve bunun sonucunda da çeşitli kazalar meydana gelir. Bu sebeplerle ADAS'ın içinde yer alan trafik işaretlerini algılayan ve sınıflandıran sistemler mevcuttur. Derin öğrenme teknikleri, özellikle de konvolüsyonel sinir ağları üzerinde yapılan çalışmalar, bu alanda önemli ilerlemelere yol açmıştır. Son yıllarda ResNet50, GoogleNet, SqueezeNet ve AlexNet gibi birçok derin öğrenme algoritmaları, trafik işaretlerini daha doğru bir şekilde sınıflandırmaya yardımcı olmak için kullanılmıştır; ancak bu algoritmaların gerçek zamanlı bir sistemde iyi bir performans gösterebileceği konusu tartışmalıdır. Bu çalışmada derin öğrenme algoritmaları transfer öğrenim metodu ile kullanılarak sınıflandırma işlemleri gerçekleştirilmiş ve bu algoritmaların gerçek zamanlı bir sistemde performansları test edilmiştir. Bu çalışmada veri seti olarak, 43 sınıftan oluşan ve 39209 trafik işareti görüntüsü içeren German Traffic Sign Recognition Benchmark (GTSRB) kullanılmıştır. Öncelikle bu algoritmaların hiper parametreleri sistematik bir yöntemle belirlenmiştir. Daha sonra tasarlanan bir kullanıcı arayüzüyle gerçek zamanlı deneyler yapılmıştır. Çalışmada kullanılan mimarilerden ResNet50, %93,49 doğruluk oranı ile gerçek zamanlı bir sistemde en iyi performansı verdiği gözlemlenmiştir. Sonuçta, ResNet50 derin öğrenme algoritması, trafik işaretinin gerçek zamanlı bir sistemde algılanması alanındaki ileri çalışmalar için değerlendirilebilir.

Real time classification of traffic sign with deep learning methods

H I G H L I G H T S

- Classification of traffic signs with convolutional neural networks
- Design of real time test system
- Finding optimum hyperparameters of convolutional neural networks

Article Info

Research Article
Received: 08.01.2024
Accepted: 07.10.2024

DOI:

10.17341/gazimmfd.1416186

Keywords:

Deep learning,
convolutional neural
network,
traffic sign classification,
transfer learning,
hyperparameter

ABSTRACT

Nowadays, traffic safety is significantly affected by advanced driver assistance systems (ADAS). Traffic signs that carry important information for drivers often cannot be noticed by drivers, and as a result, various accidents occur. For these reasons, there are systems that detect and classify traffic signs within ADAS. Studies on deep learning techniques, especially convolutional neural networks, have led to significant advances in this field. In recent years, many deep learning algorithms such as ResNet50, GoogleNet, SqueezeNet, and AlexNet have been used to help classify traffic signs more accurately; However, it is controversial whether these algorithms can perform well in a real-time system. In this study, classification operations were carried out using deep learning algorithms with the transfer learning method, and the performances of these algorithms were tested in a real-time system. In this study, German Traffic Sign Recognition Benchmark (GTSRB), which consists of 43 classes and contains 39209 traffic sign images, was used as the data set. First of all, the hyperparameters of these algorithms were determined by a systematic method. Real-time experiments were performed with a user interface designed later. Among the architectures used in the study, ResNet50 was observed to give the best performance in a real-time system with an accuracy rate of 93.49%. Ultimately, the ResNet50 deep learning algorithm can be evaluated for further studies in the field of traffic sign detection in a real-time system.

1. Giriş (Introduction)

Günümüzde, otomobillerde kullanılan ADAS, sürüş güvenliğini artırmak ve sürücülere daha güvenli bir sürüş deneyimi sunmak amacıyla sürekli olarak gelişmektedir. Bu sistemlerin önemli bir bileşeni, trafik işaretlerini algılama, sınıflandırma ve sürücüye anlamlı bilgiler sunma yeteneği üzerine odaklanmaktadır. Trafik işaretleri, sürücülerin ve yol güvenliğinin en önemli unsurlarından biridir. İstikrarlı bir trafik akışı sağlamak, kaza riskini azaltmak ve genel sürüş güvenliğini sağlamak için bu işaretlere uyulması kritik öneme sahiptir. Ancak, sürücülerin istemli veya istemsiz olarak bu trafik işaretlerine uymadığı durumlar, ciddi kazalara ve güvenlik ihlallerine neden olabilir. Örneğin, bir sürücünün dur işaretine uymaması kavşaklarda çarpışmalara, öncelik verme kurallarına uyulmaması ise kesişen araçlar arasında ölümcül kazalara sebep olabilir. Ayrıca, hız sınırlarına uyulmaması, özellikle yoğun trafikte frenleme mesafesini artırarak arkadan çarpma kazalarına yol açabilir. Bu gibi senaryolar, trafik işaretlerine uyulmamasının sadece sürücünün bireysel güvenliğini değil, aynı zamanda etrafındaki diğer kullanıcıların güvenliğini de tehlikeye attığını göstermektedir. Ayrıca, insansız araçlar açısından bakıldığında, ADAS'ın rolü daha da kritik hale gelmektedir. İnsansız araçlar, trafik işaretlerini algılamak ve doğru bir şekilde yorumlamak için ADAS teknolojisinin güçlü bir şekilde kullanılmasına dayanmaktadır. Bu, insansız araçların trafik kurallarına tam uyum sağlaması ve etkileşimde bulunduğu diğer araçlarla güvenli bir şekilde iletişim kurabilmesi için önemlidir. Dolayısıyla, bu çalışmada hem geleneksel sürücülere hem de insansız araçlara yönelik ADAS teknolojisinin güvenlik ve etkinlik açısından kritik rolü vurgulanmaktadır.

Literatürdeki çalışmalar da göz önüne alındığında trafik işaretlerini sınıflandırmaya olan ilginin ve kullanılan alanlarının giderek arttığı görülebilir. Bu çalışmalarda genellikle kullanılan veri setleri belirli bir bölgeye veya ülkeye özgü trafik işaretlerini içermekte ve modellerin genellikle çeşitli koşullarda nasıl performans gösterdiğini değerlendirmek amacıyla kullanılmaktadır. Bu çalışmaların doğruluk sonuçları, modellerin güvenilirliği ve gerçek dünya uygulamalarında kullanılabilirliği hakkında önemli bilgiler sağlamaktadır. Lorenzo Canese vd. [1] 2022 yılında gerçekleştirdiği çalışmada, ResNet, SqueezeNet ve AlexNet algoritmalarını kullanarak GTSRB veri setini eğiterek trafik işareti görsellerinin sınıflandırılması üzerine odaklanılmıştır. Aynı koşullar altında yapılan bu çalışmada en iyi performansın ResNet50 algoritması tarafından %99,97 başarı oranı olarak elde edildiğini göstermiştir. 2018 yılında Sudha Natarajan vd. [2] tarafından gerçekleştirilen başka bir çalışmada, GTSRB veri setinde bulunan trafik işaretlerini sınıflandırmak için güvenilir bir CNN mimarisi oluşturmayı amaçlamış ve bu oluşturulan mimariyi MobileNet, SqueezeNet, GoogleNet, ResNet50 ve VGG16 CNN algoritmalarıyla eşit koşullarda karşılaştırmıştır.

Çalışmanın sonuçlarına göre, önerilen CNN algoritması %99,59 başarı oranı ile en iyi performansı sergilemiştir. Diğer CNN mimarilerinden SqueezeNet %97,84, GoogleNet %98,74 ve ResNet50 ise %98,49 başarı oranlarına sahip olmuştur. 2021 yılında Soulef Bouaafia vd. [3] tarafından gerçekleştirilen çalışmada, ideal CNN algoritma hiper parametrelerini belirlemek amacıyla VGG16, VGG19, AlexNet ve ResNet50 mimarileriyle GTSRB veri seti eğitilerek deneyler yapılmıştır. Bu çalışmada ise en başarılı algoritmanın %99,74 doğruluk oranına sahip olan AlexNet olduğunu belirlemiştir. Diğer mimariler arasında ise ResNet50, %98,63 doğruluk oranıyla en düşük performansı sergileyen mimari olarak ortaya çıkmıştır. 2019 yılında Wenhui Li vd. tarafından gerçekleştirilen çalışmada [4], 43 sınıfa ayrılmış GTSRB veri setiyle LeNet-5, SqueezeNet, AlexNet, GoogleNet gibi popüler CNN

algoritmaları ile özel olarak oluşturulan bir CNN algoritması arasında performans karşılaştırması yapılmıştır. Eğitim parametrelerini her deney için aynı tutarak doğru bir karşılaştırma sağlanmıştır. Deney sonuçlarına göre, önerilen CNN algoritması %97,40 doğruluk oranıyla popüler CNN algoritmalarını geride bırakmıştır. Diğer kullanılan algoritmalar arasında SqueezeNet %94,20, AlexNet %95.9 ve GoogleNet %96.5 doğruluk oranlarına sahiptir. 2021 yılında Jonah Sokipriala ve Sunny Orike tarafından yapılan çalışmada [5], 50 kat daha az eğitim parametresi kullanılarak ve daha hızlı eğitim süresiyle oluşturulan 8 katmanlı bir CNN modelinin, popüler VGG16 ve ResNet50 CNN algoritmalarına göre daha yüksek doğruluk oranına sahip olabileceğini göstermiştir. GTSRB veri seti, 43 farklı sınıfa sahiptir ve bu çalışmada veri artırma yöntemleri kullanılarak bu veri setine veri eklenmiştir. 10 epok boyunca yapılan eğitim sonucunda, önerilen CNN algoritması %96,0 doğruluk oranına ulaşarak VGG16'nın elde ettiği %95.5 ve ResNet50'nin elde ettiği %95.4 doğruluk oranlarını geçmiştir. Çalışma aynı zamanda uygulanan veri artırma yönteminin aşırı uyum (overfitting) sorununu ortadan kaldırdığını göstermiştir. Anliang Kou tarafından 2020 yılında gerçekleştirilen çalışmada [6], trafik işaretlerini tespit etmek amacıyla görüntü segmentasyonunu kullanarak daha etkili sonuçlar elde etmek için bir CNN modeli tasarlanmış ve bu model AlexNet algoritmasıyla karşılaştırılmıştır. Çalışmada kullanılan veri seti, 43 farklı sınıfa ayrılmış olan GTSRB veri setidir ve toplamda 51,839 trafik işareti görüntüsü içermektedir. Veri setindeki görseller 128x128 boyutlarına ayarlanmış ve deneyler 0.001 öğrenme oranıyla gerçekleştirilmiştir. Önerilen CNN modeli, %96,20 doğruluk oranıyla AlexNet algoritmasının %91.20 doğruluk oranını geride bırakmıştır. Bu çalışmaya için yapılan literatür araştırmaları, özet bir şekilde Tablo 1 ve Tablo 2'de verilmiştir.

Yapılan çalışmalarda en büyük eksiklik eğitimlerde kullanılan trafik işareti veri setlerinde bulunan görüntü verilerinin, arka plan gürültü yoğunluğunun düşük olmasıdır. Bu durum, gerçek zamanlı bir sistemde, sürücülerin karşılaşılabileceği çeşitli zorlukları ve karmaşıklıkları tam olarak yansıtmamaktadır. Bu çalışma, özellikle arka plan gürültü yoğunluğu yüksek, farklı ışık ve hava koşullarında çekilmiş görüntü verileriyle eğitilen ve gerçek zamanlı uygulamalarda etkili bir performans sergileyebilen sınıflandırma modellerinin geliştirilmesini hedeflemektedir. Bu sebeple çalışmada çeşitli CNN mimarileri kullanılarak öncelikle en yüksek doğruluk yüzdesi elde edilecek şekilde optimum hiper parametrelerin bulunması amacıyla sistematik olarak testler yapılmıştır. Bunun yanında kullanılan veri seti gerçek zamanlı bir sisteme uygun olması amacıyla veri artırma yöntemi ile, veri setine gerçek zamanlı görseller eklenerek çeşitli testler yapılmıştır. Bu şekilde literatürdeki eksikliklerin önüne geçilmesi hedeflenmiştir. En uygun yöntemin belirlenmesi aşamasında GTSRB veri seti kullanılarak SqueezeNet, GoogleNet, ResNet50 ve AlexNet CNN mimarileriyle çeşitli testler uygulandı. Yapılan testler sonucunda trafik işaretlerinin sınıflandırılmasında en yüksek doğruluk oranları veren CNN algoritmaları, tasarlanan bir kullanıcı arayüzünde gerçek zamanlı olarak test edilmiştir.

Bu çalışmanın amacı, popüler olarak kullanılan SqueezeNet, GoogleNet, ResNet50 ve AlexNet algoritmalarını GTSRB veri seti ve veri artırma yöntemleri kullanarak oluşturulan veri kümeleri ile eğiterek, trafik işaretlerini etkili bir şekilde algılayabilen, sınıflandırabilen ve gerçek zamanlı uygulamalarda başarılı bir performans sergileyebilen bir sistem için optimum hiper parametreleri belirlemek ve bu algoritmaları gerçek zamanlı test edebilmek amacıyla kullanıcı arayüzü tasarlanmıştır. Elde edilecek sonuçlar, trafik işaretlerini gerçek zamanlı algılama ve sınıflandırma konusundaki etkili modelin belirlenmesine katkı sağlayacaktır. Bu çalışmanın akış diyagramı Şekil 1'de verilmiştir.

Tablo 1. Literatür çalışmalarının özet tablosu (Summary table of literature studies)

Makale	Yazarlar	Yıl	Veri Seti	Veri Sayısı	Sınıf Sayısı	Performans Aracı	Algoritma	Doğruluk (%)
Sensing and Detection of Traffic Signs Using CNNs: An Assessment on Their Performance [1]	Lorenzo Canese	2022	GTSRB	50000	43	Doğruluk	ResNet50 ResNet18 Xception ResNet50 AlexNet SqueezeNet	99.97% 99.95% 99.89% 99.87% 99.63%
Evaluation of Vision Transformers for Traffic Sign Classification [7]	Yuping Zheng	2022	GTSRB	51839 51839	43	Doğruluk	Vgg16 DenseNet ResNet SqueezeNet MobileNet ShuffleNet MnasNet	99.94% 99.90% 99.82% 99.56% 99.56% 98.96% 98.96%
Traffic Sign Recognition Using Weighted Multi-Convolutional Neural Network [2]	Sudha Natarajan	2018	GTSRB	51839	43	Doğruluk	Tasarlanan CNN GoogleNet ResNet50 Vgg16 SqueezeNet MobileNet	99.59% 98.74% 98.49% 98.46% 97.84% 97.39%
Deep Pre-trained Models for Computer Vision Applications: Traffic Sign Recognition [3]	Soulef Bouaafia	2021	GTSRB	39209	43	Doğruluk Kesinlik Duyarlılık F1-Skor	Vgg19 Vgg16 AlexNet ResNet50	99.45% 99.43% 99.74% 98.63%
Detection and Recognition of Traffic Signs based on Improved Deep Learning [6]	Anliang Kou	2020	GTSRB	51839	43	Doğruluk	Tasarlanan CNN AlexNet	96.20% 91.20%
Traffic Sign Classification Comparison Between Various Convolution Neural Network Models [5]	Jonah Sokipriala	2021	GTSRB	39209	43	Doğruluk	Tasarlanan CNN Vgg16 ResNet50	96.00% 95.50% 95.40%
Comparing Performance for Myanmar Road Signs Recognition [8]	Tin Zar Htun	2023	Myanmar Sign Recognition Dataset	2678	25	Doğruluk Kesinlik Duyarlılık	GoogleNet AlexNet HOG-SVM	98.32% 97.95% 97.57%
Traffic Sign Recognition with a small convolutional neural network [4]	Wenhui Li	2019	GTSRB	51831	43	Doğruluk	Tasarlanan CNN GoogleNet AlexNet SqueezeNet LeNet-5	97.40% 96.50% 95.90% 94.20% 87.00%
Traffic Sign Recognition by Integrating Convolutional Neural Network and Support Vector Machine [9]	Shawkh Ibne Rashid	2019	GTSRB	51839	43	Doğruluk Kesinlik Duyarlılık F1- Skor	ResNet50+SVM ResNet50 AlexNet AlexNet+SVM	97.55% 96.41% 96.19% 95.67%

Bu makale şu şekilde organize edilmiştir: Bölüm 1, bu çalışmada kullanılan materyal ve metotları anlatmaktadır. Bu bölümün içinde GTSRB veri setini, bilgisayarlı görme uygulamaları için kullanılan,

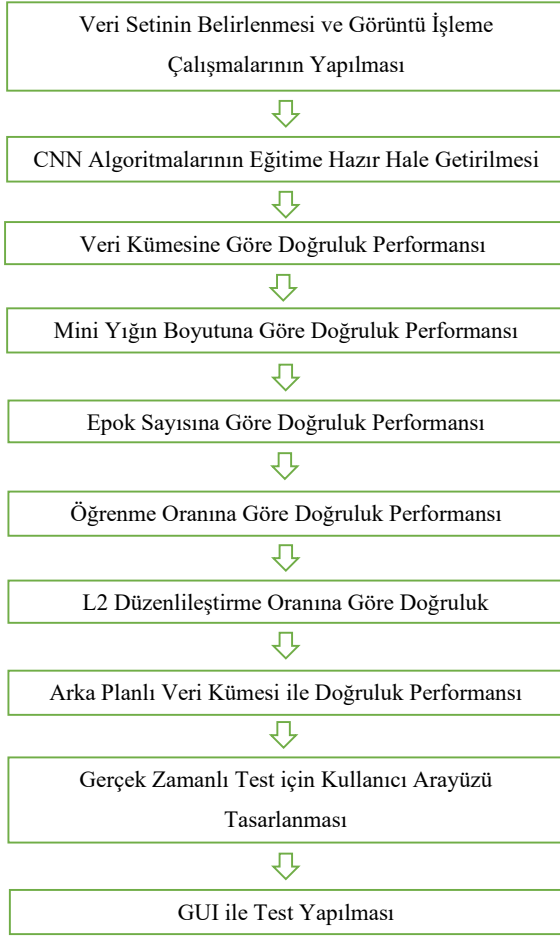
önceden eğitilmiş derin CNN'lere odaklanan önceki yaklaşımları, veri setinin zenginleştirilmesi ve eğitimler için daha uygun hale getirilmesi amacıyla veri artırma yöntemleri, eğitim sonuçlarını değerlendirmek

amacıyla kullanılan performans değerlendirme metriği ve gerçek zamanlı testlerin yapılabilmesi için oluşturulan grafiksel kullanıcı arayüzü tasarımı gözden geçiriliyor.

Bölüm 3, elde edilen deneysel performans sonuçlarını analiz etmektedir. Bu makalenin sonuçları bölüm 4'te tartışılmaktadır.

Tablo 2. Literatür çalışmalarının özet tablosu devamı (Summary table of literature studies continued)

Makale	Yazarlar	Yıl	Veri Seti	Veri Sayısı	Sınıf Sayısı	Performans Aracı	Algoritma	Doğruluk (%)
A Comparative Study of Machine Learning Methods for Traffic Sign Recognition [10]	Ioan Cristian Schuszter	2017	BTSD	9095	62	Doğruluk	LeNet ResNet18 ResNet50 HOG + SVM	95.83% 93.69% 89.92% 86.00%
Traffic Signs Classification with Transfer Learning of CNN Based Models and Performance Comparison [11]	Mohammed Ghazi	2023	GTSRB	26442	44	Doğruluk Kesinlik Duyarlılık F1- Skor	XceptionNet ResNet50 MobileNet VGG16 VGG19 DenseNet121 EfficientNet	97.90% 96.80% 92.70% 88.80% 88.15% 85.60% 71.90%
Otonom Araçlar İçin Trafik İşaretlerini Tanıma Sistemi [12]	Emre Balkan	2022	GTSRB	50000	43	Doğruluk	Tasarlanan CNN	95.00%
Trafik İşaretlerini Tanıyan Bir Sürücü Güvenlik Destek Sistemi [13]	Mehmet Zam	2019	GTSRB	156692	18	Doğruluk	ResNet50 GoogleNet AlexNet LeNet-5	98.10% 97.40% 95.20% 94.40%
Real-Time Application of Traffic Sign Recognition Algorithm with Deep Learning [14]	Faruk Emre Aysal	2022	Afyon karahisar	1249	26	F1-Skor	YOLO v5s YOLO v5m	98.00% 99.00%
An effective automatic traffic sign classification and recognition deep convolutional networks [15]	Jayant Mishra	2022	GTSRB GTSDB BTSC TSRD+ GTSRB	51839 900 7171 58003	43 43 62 101	Doğruluk	Tasarlanan CNN	99.76% 99.81% 99.79% 98.37%
A real-time and high-precision method for small traffic-signs Recognition [16]	Junzhou Chen	2021	Tencent Street View Map	30000	45	mAP	Tasarlanan CNN Tasarlanan CNN (RFB-c ile)	89.9% 90.2%
A real-time simulation environment architecture for autonomous vehicle design [17]	Yusuf Özçevik	2023	GTSRB	51839	8	Doğruluk	Faster R-CNN Mask R-CNN YOLO-v4	91% 93% 95%



Şekil 1. Trafik işaretleri sınıflandırma süreci akış diyagramı (Flow diagram of Traffic signs classification process)

2. Deneysel Metot (Experimental Method)

2.1. Veri Seti (Dataset)

Bu çalışmada, GTSRB ve bu veri setinden türetilen üç farklı veri seti kullanıldı [18]. GTSRB veri setine Orijinal Veri Seti adı verildi. Veri artırma teknikleri kullanılarak üç farklı veri seti oluşturuldu ve toplamda dört farklı veri seti eğitimlerde kullanıldı. Orijinal Veri Seti'nde bulunan trafik işaretleri, farklı hava ve parlaklık koşullarında renkli görüntülerden oluşur ve bu görsellerin piksel boyutları 15 x 15 ile 250 x 250 piksel arasında değişmektedir. Toplamda 43 farklı trafik işareti sınıfını içerir ve eğitim ve test amaçları için 39,209 trafik işareti görüntüsü içerir. Şekil 2'de veri setinden örnek görseller gösterilmektedir. Test verisi, eğitim verisinin %20'sini oluşturan 7,842 trafik işareti görüntüsünden oluşmaktadır. Ancak her sınıfa ait görüntü sayısı homojen bir şekilde dağılmamakta ve yeterli gürültü içermemektedir. Bu nedenle, Orijinal Veri Seti'nden türetilen veri artırma teknikleri kullanılarak üç farklı veri seti oluşturulmuştur. Dengeli Veri Seti, veri setindeki verilerin sınıflardaki dağılımının etkisini incelemek için oluşturulmuştur. Bu veri seti, Orijinal Veri Seti'nde bulunan her sınıfa ait görüntülerin en fazla görüntü sayısına sahip sınıfa, yani 2250'ye eşitlenmesiyle oluşturulmuştur. Bu veri seti oluşturulurken, Orijinal Veri Seti'nden rastgele trafik işareti görüntüleri seçilmiş, saat yönünün tersine doğru 0-15 derece arasında rastgele döndürülmüş ve veri setinde bulunan her sınıf, 2250 trafik işareti görüntüsüne tamamlanmıştır. Dengeli Veri Seti'nde bulunan trafik işaretleri, farklı hava ve parlaklık koşullarında renkli

görüntülerden oluşmaktadır ve bu görsellerin piksel boyutları 15 x 15 ile 250 x 250 piksel arasında değişmektedir. Dengeli Veri Seti'nde toplamda 96,750 trafik işareti görüntüsü bulunmaktadır. Test verisi, eğitim verisinin %20'si seçilerek 19,350 trafik işareti görüntüsü olarak belirlenmiştir. Arttırılmış Veri Seti, veri setinin daha fazla ve daha çeşitli veriye sahip olduğu durumun etkisini incelemek için oluşturulmuştur. Bu veri seti, Orijinal Veri Seti'ndeki her sınıfa ait rastgele seçilen görüntülere, görüntü artırma teknikleri uygulanarak çoğaltma işlemi yapılmıştır. İlk aşamada, Orijinal Veri Seti'nde bulunan görüntüler, gerçek zamanlı bir sistemde kameranın açısının veya tabelanın açısının değişebileceği öngörüsüyle 20 derece saat yönünün tersinde döndürülmüştür. İkinci aşamada, Orijinal Veri Seti'nde bulunan görüntüler, gerçek zamanlı bir sistemde tabelanın güneş veya farklı bir ışık kaynağı sebebiyle yansıma durumu göz önüne alınarak x ekseninde, yani yatayda yansımaları alınmıştır. Üçüncü aşamada ise Orijinal Veri Seti ile ilk iki aşamada yapılan çalışma birleştirilmiş ve toplamda 117,627 görüntü içeren Arttırılmış Veri Seti elde edilmiştir. Test verisi, eğitim verisinin %20'si seçilerek 23,525 trafik işareti görüntüsü olarak belirlenmiştir. Arka Planlı Veri Seti, veri setinde bulunan verilerin eğitim sırasında arka plan yoğunluğundan ne kadar etkilendiğini incelemek için oluşturulmuştur. Bu veri seti oluşturulurken, Dengeli Veri Seti'ne ait rastgele seçilmiş 50 görüntü, sınıflardan silinmiş ve yerine arka planında trafik akışı, yol, ağaç vb. gibi materyaller bulunan 50 trafik işareti görüntüsü eklenmiştir. Arka Planlı Veri Seti'nde toplamda 96,750 trafik işareti görüntüsü bulunmaktadır. Test verisi, eğitim verisinin %20'si seçilerek 19,350 trafik işareti görüntüsü olarak belirlenmiştir.

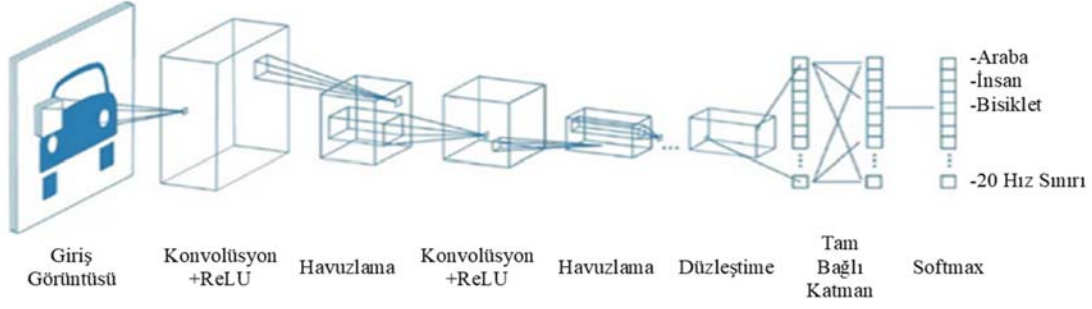


Şekil 2. GTSRB örnek görseller (GTSRB sample pictures)

2.2. Konvolüsyonel Sinir Ağları (Convolutinal Neural Network)

CNN ileri beslemeli çok katmanlı bir algılayıcıdır [19]. Özellikle bilgisayarla görme görevlerinde derin öğrenme topluluğu içinde en başarılı ve yaygın olarak tercih edilen mimarilerden biridir [20]. CNN, çoklu katmana sahip olup tanıma, kümeleme ve sınıflandırma için kullanılır [21].

1998 yılında Yann LeCun vd. tarafından LeNet-5 mimarisi ile kullanılan CNN, konvolüsyonel ağların ilk başarılı uygulamalarından biri olarak kabul edilir. CNN, beyin görsel korteksinin bir özelliği olan reseptif alandan ilham alarak tasarlanmıştır [19]. Gelişmiş ağ yapısı, hesaplama karmaşıklığını azaltırken bellek kullanımında tasarruf sağlar. Görüntüden öznitelik çıkarma ve bu öznitelik çoğaltma yeteneği, CNN'i oldukça yüksek sınıflandırma performansına sahip kılar [22, 23]. Şekil 3'te CNN'nin gelen mimarisi gösterilmektedir.



Şekil 3. Konvolüsyonel sinir ağı mimarisi (Structure of convolutional neural network)

Eğitilebilir çok katmanlı CNN'ler genellikle giriş, konvolüsyon, havuzlama, sınıflandırma katmanlarından oluşur. Konvolüsyon katmanında, giriş katmanından aldığı renkli (3 kanallı) veya tek renkli bir görüntü verilerini ardışık katmanlarda filtreleri kullanarak işler [21]. Bu filtreler, eğitim sürecinde kendi değerlerini öğrenir ve verilerdeki belirli desenleri ortaya çıkarır. Havuzlama katmanı, konvolüsyon katmanlarından gelen verilerin boyutunu azaltmak için belirli yöntemleri kullanarak verileri işler. Sınıflandırma katmanında, elde edilen veriler vektör haline getirilir ve çok katmanlı algılayıcılar kullanılarak nihai sonuç elde edilir. Elde edilen sonuç ile istenen sonuç arasındaki farka göre bir hata oluşur. Bu hatanın minimum olması hedeflenir. Geri yayılım algoritması kullanılarak ağırlıkların güncellenmesiyle hata, tüm ağırlıklara iletilir. Her iterasyonda ağırlıkların güncellenmesiyle hata azaltılarak süreç devam eder [24].

2.3. Transfer Öğrenimi (Transfer Learning)

Bir derin ağı sıfırdan eğitmek istenildiğinde ve yeterince geniş bir veri setine sahip olunmadığında, transfer öğrenimi kullanarak önceden eğitilmiş bir ağı temel almak, bu zorluğun üstesinden gelmek için etkili bir stratejidir [24]. Transfer öğrenimi yaklaşımı, özellikle yeterli sayıda veri olmadığı durumlarda, önceden eğitilmiş parametrelerin transferi ile spesifik problemlere etkili çözümler sağlar. Belirli bir görev için eğitilmiş derin yapılar, modifikasyonlarla ilişkilendirilerek başka bir görev için uyarlanabilir. Dolayısıyla, derin bir modeli sıfırdan eğitmek için yeterli örnek olmadığında, transfer öğrenimi kullanışlı bir alternatiftir [20]. Transfer öğrenimi, ağıdaki parametreleri uygun başlangıç değerleriyle başlatır ve küçük modifikasyonlarla yeni probleme uyarlar. Ayrıca, parametrelerin ayarlanması, kabul edilebilir bir başarı elde etmek için daha kısa bir eğitim süresi gerektirecektir [25].

Transfer öğrenimi yönteminin en önemli avantajlarından biri, önceden eğitilmiş bir derin ağı zengin öznitelik kümesini öğrenmiş olmasıdır. Bu öznitelikler, farklı problemlerde de etkili olabilir. Örneğin, milyonlarca görüntü üzerinde eğitilmiş bir derin ağı öğrendiği zengin öznitelik çıkarımları, yalnızca yüzlerce görüntü içeren yeni bir veri kümesinde sınıflandırma yapmak için tekrar eğitilebilir. Transfer öğrenimi için kullanılan iki temel strateji bulunmaktadır.

Önceden Eğitilmiş Modellerin Öznitelik Çıkarıcı Olarak Kullanılması: Derin öğrenme modelleri, katmanlı mimarilerinde farklı seviyelerde öznitelikleri öğrenen yapılar içerir. Bu strateji, önceden eğitilmiş modellerin, örneğin AlexNet, GoogleNet gibi ağırlıklı katmanlarının sadece softmax çıkışı, yeni görevin sınıf sayısına göre ayarlamak amacıyla kullanılır. Bu yöntem, transfer öğrenimi uygulamanın en yaygın ve etkili yollarından biridir.

Önceden Eğitilmiş Modellerde İnce Ayar Yapılması: Bu strateji, yalnızca son katmanın değiştirilmediği, aynı zamanda önceki katmanlardan bazılarının seçilip yeniden eğitildiği bir tekniktir. Derin sinir ağlarının yapılandırılabilir mimarileri, genel özellikleri öğrenen ilk katmanlara ve daha spesifik özellikleri öğrenen sonraki katmanlara

sahiptir. Bu strateji, belirli katmanların dondurulup yani ağırlıklarının sabitlenmesiyle, geri kalan katmanlarda ise ihtiyaca bağlı ince ayarlamaların yapıldığı bir yaklaşımı içerir. Böylece, dondurulan ağırlıklar, yeniden eğitim aşamasında bir başlangıç noktası olarak kullanılır ve daha kısa sürede daha iyi performans elde edilmesine olanak tanır [24].

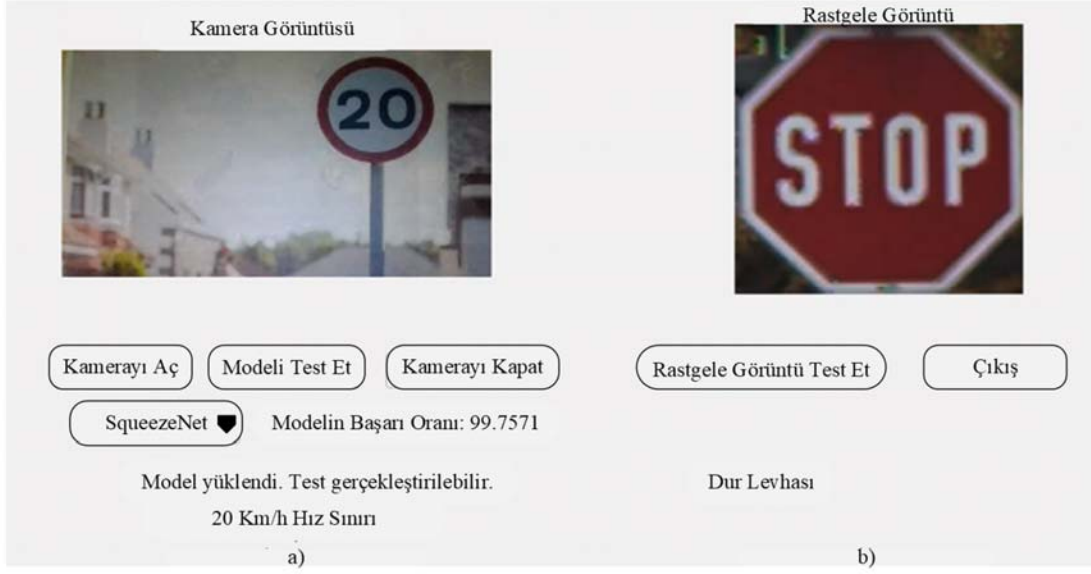
Bu çalışmada, kullanılan derin öğrenme mimarileri ile transfer öğrenme yapılırken Önceden Eğitilmiş Modellerde İnce Ayar Yapılması metodu kullanılmıştır. Son katman olan sınıflandırma katmanında, çıkış boyutu (output size) varsayılan değer yerine veri setinin sınıf sayısına göre değiştirilmiştir. Tam bağlantılı katmanında çıkış boyutu yine varsayılan değer yerine veri setinin sınıf sayısına göre değiştirilmiştir. Eğitimleri hızlandırmak adına ağırlık öğrenme faktörü (weight learning factor) ve bias öğrenme faktörü (bias learning factor) varsayılan değerler yerine 20'ye ayarlanmıştır. İlk 10 katman ise dondurulup ağırlıklarını sabitlenmiştir.

2.4. Çalışmada Kullanılan Algoritmaların Yapıları (Structures of the Algorithms Used in the Study)

AlexNet: 2012 yılında, Alex Krizhevsky vd. [26], derin ve geniş bir CNN modeli olan AlexNet'i geliştirdi ve ImageNet Büyük Ölçekli Görsel Tanıma Yarışması'nı (ILSVRC) kazandı. AlexNet mimarisindeki diğer yeni bir yaklaşım ise ReLu aktivasyonunun ilk kez bu mimaride standart olan tanh yerine kullanılması olmuştur [27]. 60 milyon parametre ve 650.000 nöron içeren sinir ağı; 5 evrişim katmanı bunların birçoğunu takip eden maksimum havuzlama katmanı ve 3 tam bağlantılı katmandan oluşmaktadır. [26].

GoogleNet: Christian Szegedy [28] tarafından geliştirilen %5,7 hata oranı ile 2014 ILSVRC yarışmasını kazanarak ortaya çıkan bir modeldir. Bu mimari genel olarak, ardışık bir yapıda konvolüsyon ve havuzlama katmanlarını üst üste koymaktan uzaklaşan ilk CNN mimarilerinden biridir. [29]. Temel hedefi, geleneksel CNN'lerle karşılaştırıldığında hesaplama karmaşıklığını azaltmaktır. Bu amaçla inception katmanlarını içeren bir yöntem önerilmiştir; bu katmanlar değişen çekirdek boyutlarıyla değişken reseptif alanlar oluşturarak yeni özellik haritası yığımında seyrek korelasyon desenlerini yakalamak için operasyonlar gerçekleştirir. GoogleNet, bir dizi inception katmanını kullanarak tanıma doğruluğunu artırmıştır. GoogleNet toplamda 22 katmandan oluşur, bu, AlexNet'ten çok daha fazladır. Ancak, GoogleNet'in kullandığı ağ parametre sayısı, AlexNet gibi önceki modellere kıyasla çok daha düşüktür [27].

ResNet50: Residual Network, %3,6 hata oranıyla ILSVRC 2015 yarışmasında birinci olan ve Kaiming He [30] tarafından geliştirilen şimdiye kadar tasarlanan tüm mimarilerden daha derin bir yapıya sahip olan bir mimaridir. ResNet, önceki modellerin yaşadığı vanishing gradient sorununu çözmek amacıyla ultra-derin ağlar ile oluşturulmuştur. Başarı oranını artırmak amacıyla, her katmandaki



Şekil 4. Trafik işaretlerini gerçek zamanlı sınıflandıran kullanıcı arayüzü tasarımı. a) gerçek zamanlı test bölümü. b) gerçek zamanlı olmayan test bölümü

(User interface design that classifies traffic signs in real time. a) real-time testing section. b) non-real-time testing part)

değişiklikleri dikkate almadan bir sonraki alt katmana geçişin daha sorunsuz olmasını sağlamak üzere bir artık blok ile tasarlanmıştır. ResNet, farklı katman sayıları ile geliştirilmiştir; 34, 50, 101, 152 ve 1202. Aralarında popüler olan ResNet50, 49 evrişim katmanı ve ağır sonunda bir tam bağlantılı katman toplamda ise 177 katmandan oluşan bir mimariden oluşur.

SqueezeNet: SqueezeNet 2016'da Iandola vd. tarafından geliştirilmiş olup bu mimari ile görüntü işleme problemlerinde büyük derin mimarilerin sağladığı başarıyı korurken daha küçük boyutlu alternatif bir mimari geliştirmek amaçlanmıştır. [31]. Bu evrişim ağı, AlexNet'e kıyasla 50 kat daha az parametre ile daha iyi performans gösterdiği gözlemlenmiştir [32]. Ayrıca, modelin boyutu sıkıştırma teknikleri kullanılarak 0.5 MB'den daha küçük bir bellek kapasitesinde tutulmuştur. Bu özellikler, SqueezeNet modelini eğitim sırasında daha az iletişim, daha küçük bant genişliği ve sınırlı depolama alanına sahip donanımlara entegre edilebilme yeteneği ile popüler kılmıştır [32]. SqueezeNet mimarisi, beş farklı katmana sahip toplamda on beş katmandan oluşur [33]. SqueezeNet mimarisi, daha az parametre ile yüksek doğruluk elde etme konusunda etkileyici bir başarı sergiler [34]. SqueezeNet mimarisi, önerilen ateşleme modüllerinden oluşmaktadır. Her ateşleme modülü, sıkıştırılmış ve genişletilmiş evrişim katmanlarından meydana gelir [32].

2.5. Veri Arttırma (Data Augmentation)

Küçük bir veri kümesiyle modellere, eğitim yapılırken aşırı uyum (overfitting) sorununa sebep olabilir. Bu durumda, model yalnızca eğitim kümesinin detaylarını hatırlar ve sınıflandırma problemi ortaya çıkar, bu nedenle aşırı uyum sorununu azaltmak için veri artırma tekniği kullanılır. Veri artırımı birçok derin öğrenme çalışmasında performans artışında önemli katkılar sağlamaktadır. Veri artırımı yapılarak performans sonuçları üzerinde kıyaslama yapılabilir. Veri arttırmada kullanılan Geometrik yöntemler Krizhevsky ve arkadaşları [26] ile popüler hale getirildikten sonra, en çok kullanılan veri artırma yöntemlerinden biridir.

Çevirme (Flipping): Yatay eksen çevirme, dikey eksen çevirmekten çok daha yaygındır. Bu artırma, uygulanması en kolay olanlardan

birdir ve CIFAR-10 ve ImageNet gibi veri setlerinde etkili olduğu kanıtlanmıştır.

Renk uzayı (Color space): Dijital görüntü verisi genellikle (yükseklik \times genişlik \times renk kanalları) boyutundaki bir tensör olarak kodlanır. Renk kanalları uzayında yapılan artışlar, uygulaması oldukça pratik olan başka bir stratejidir. Çok basit renk artışları, R (Kırmızı), G (Yeşil) veya B (Mavi) gibi tek bir renk kanalını izole etmeyi içerir. Bir görüntü, o matrisi izole ederek ve diğer renk kanallarından gelen 2 sıfır matris ekleyerek hızlı bir şekilde tek bir renk kanalındaki temsiline dönüştürülebilir. Ayrıca, RGB değerleri, görüntünün parlaklığını artırmak veya azaltmak için basit matris işlemleri kullanılarak kolayca değiştirilebilir.

Kesme (Cropping): Görüntülerin kırılması, her bir görüntünün merkezi bir parçasını kırma yoluyla karışık yükseklik ve genişlik boyutlarına sahip görüntü verileri için pratik bir işleme adımı olarak kullanılabilir. Ayrıca, rastgele kırma, translasyonlarla oldukça benzer bir etki sağlamak için de kullanılabilir.

Döndürme (Rotation): Döndürme artırımları, görüntüyü bir eksende sağa veya sola 1° ila 359° arasında döndürerek gerçekleştirilir. Döndürme artırımlarının güvenliği, döndürme derecesi parametresi tarafından büyük ölçüde belirlenir.

Kaydırma (Translation): Görüntüleri sola, sağa, yukarı veya aşağı kaydırmak, verideki konumsal önyargıyı önlemek için çok kullanılan bir dönüşüm olabilir. Orijinal görüntü bir yönde kaydırıldığında, kalan boşluklar 0 veya 255 gibi sabit bir değerle doldurulabilir ya da rastgele veya Gauss görüntüsüyle doldurulabilir.

2.6. Kullanıcı Arayüzü Tasarımı (User Interface Design)

Algoritmaların gerçek zamanlı bir sistemde test edilebilmesi için bir kullanıcı arayüzü tasarlanmıştır. Bu test için oluşturulan kullanıcı arayüzü, MATLAB yazılımının App Designer özelliğiyle yapılmıştır. Şekil 4'te trafik işaretlerinin çalışmada kullanılan dört algoritma üzerinde gerçek zamanlı test imkanı sağlayan kullanıcı arayüzünden bahsedilmiştir.

Tasarlanan kullanıcı arayüzü iki temel bölüme ayrılmıştır. Şekil 4'teki "a" bölümü, gerçek zamanlı test imkânı sunarken, "b" bölümü lokalde bulunan verilerle gerçek zamanlı olmayan bir test fırsatı sunmaktadır. "a" bölümünde yer alan 'Webcam Image' sekmesi, gerçek zamanlı kameraya yansıyan görüntüyü içermektedir. Gerçek zamanlı testlerde kullanılan kamera, 30 FPS (frame per second) hızında çalışmaktadır. 30 FPS, gerçek zamanlı uygulamalar için yaygın olarak kullanılan bir değerdir ve yeterli akıcılık ile performans sağlar. "OPEN CAMERA" butonu bilgisayarda gömülü olarak bulunan kamerayı açar, "CLOSE CAMERA" butonu ise kamerayı kapatır. "MODEL" seçim butonu ile daha önce Arka Planlı Veri Kümesi ile eğitilmiş SqueezeNet, GoogleNet, ResNet50 veya AlexNet algoritmalarından biri seçilebilir ve seçilen algoritma yüklenir. Algoritmanın başarıyla yüklendiğinde, altındaki etiket alanında "Model uploaded. The test can be performed." yazısı görüntülenir. Ayrıca algoritma seçildiğinde, seçilen algoritmanın başarı oranını gösteren bir etiket bulunmaktadır. "TEST MODEL" butonu ile seçilen algoritma kullanılarak kameraya yansıyan görüntü gerçek zamanlı olarak test edilir. Testin sonucu, "a" bölümünün en altında bulunan etikette anlık olarak görüntülenir. "b" bölümünde ise, bu çalışmada kullanılan bilgisayar ortamında bulunan 12630 adet test görüntüsü içeren Test Veri Kümesi ile gerçek zamanlı olmayan bir test yapılabilmektedir. Algoritma seçimi, "a" bölümündeki "MODEL" seçim butonu ile gerçekleştirilir. "TEST RANDOM IMAGE" butonu ile lokalde bulunan bu test veri kümesinden rastgele bir trafik işareti görseli kullanıcı arayüzüne yüklenir. Altındaki etikette ise trafik işareti görselinin sınıfı görüntülenir.

3. Sonuçlar ve Tartışmalar (Results and Discussions)

Bu bölümde, algoritmalar için yapılan test sonuçlarına yer verilmiştir. Yapılan testlerdeki amaç algoritmaların en iyi performansı gösterdiği

hiper parametreleri bulup, gerçek zamanlı testlerde başarı durumunu gözlemlemektir.

3.1. Veri Seti Karşılaştırması (Comparison of Dataset)

İlk aşamada testlerin devamında kullanılacak en verimli veri setini seçmektir. Bu sebeple veri artırma yöntemiyle oluşturulmuş veri setleri tüm algoritmalar ile eğitime girmiştir. Aynı koşullar altında kıyaslamak amacıyla mimarilerin eğitim hiper parametreleri için başlangıçta varsayılan değerlerler verilmiştir. Bu değerler, Mini yığın boyutu için 32, epok sayısı için 5 öğrenme oranı için 0,001 ve L2 Düzenleme oranı için 0,0005 olarak belirlenmiştir. Sonuç olarak her algoritma varsayılan hiper parametreler ayarlanarak Orijinal Veri Kümesi, Dengeli Veri Kümesi ve Arttırılmış Veri Kümesi ile eğitim yapılmıştır. Eğitim sonuçları Tablo 3'te paylaşılmıştır.

Bu deneyde tüm CNN mimarileri en iyi performansı Dengeli Veri Kümesi'nde göstermiştir. Sonuç olarak veri setindeki verilerin sınıflara homojen olarak dağılımı CNN performansını arttırdığını göstermektedir. Bu deneyde en iyi performansı gösteren algoritma %99,98 sınıflandırma oranı ile ResNet50 çıkış da algoritmaların genel başarı oranları birbirlerine oldukça yakın çıkmıştır.

3.2. Mini Yığın Boyutu Kıyaslaması (Comparison of Mini Batch Size)

Algoritmaların en iyi performansı gösterdiği hiper parametreleri bulmak adına ilk yapılan deney mini batch size kıyaslaması olmuştur. Bu aşamada yapılan deneylere Dengeli Veri Kümesi ile devam edilmiş olup, epok sayısı 32, öğrenme oranı 0,001 ve L2 düzenleme oranı 0,005 olarak tüm mimariler için ayarlanmıştır. Mini yığın boyutu ise 16, 32 ve 64 olacak şekilde üç seviyede incelenmiştir. Eğitim sonuçları Tablo 4'te paylaşılmıştır.

Tablo 3. Veri setine göre doğruluk oranı (Accuracy rate according to data set)

Algoritma	Veri Seti	Veri Sayısı	Zaman (min)	Doğruluk (%)
SqueezeNet	Orijinal	39209	18,36	99,71
	Dengeli	96750	103,44	99,93
	Arttırılmış	117627	150,23	99,62
GoogleNet	Orijinal	39209	50,49	99,78
	Dengeli	96750	121,33	99,96
	Arttırılmış	117627	148,51	99,80
ResNet50	Orijinal	39209	280,23	99,90
	Dengeli	96750	812,20	99,98
	Arttırılmış	117627	931,43	99,93
AlexNet	Orijinal	39209	23,38	99,77
	Dengeli	96750	57,35	99,94
	Arttırılmış	117627	988,02	99,89

Tablo 4. Mini yığın boyutuna göre doğruluk oranı (Accuracy rate according to mini batch size)

Algoritma	Veri Seti	Mini Yığın Boyutu	Zaman (min)	Doğruluk (%)
SqueezeNet	Dengeli	16	214,25	99,96
		32	103,44	99,93
		64	65,35	99,93
GoogleNet	Dengeli	16	244,13	99,97
		32	121,33	99,96
		64	141,24	99,94
ResNet50	Dengeli	16	461,02	99,99
		32	812,20	99,98
		64	1169,08	99,94
AlexNet	Dengeli	16	1857,55	99,95
		32	57,35	99,94
		64	42,52	99,95

Bu deneyde eşit koşullarda mini yığın boyutunun düşürülmesi performansı olumlu yönde etkilerken eğitim süresinin arttığı gözlemlenmiştir. Eğitim esnasında veri kümesinin 16'ya bölünmesi gradyan hesaplaması ve ağırlık değerlerinin güncellenme işlemlerini daha fazla veriyle yapması anlamına gelir. Bu durum işlem oranını arttırdığı için eğitim süresinin artmasını fakat kayıp değerinin düşmesini ve doğruluk oranında artışın meydana gelmesini sağlar. Mini yığın boyutu artırıldığında ise eğitim süresinin ciddi bir oranda azaldığı fakat AlexNet mimarisi dışında performansın küçük bir şekilde düştüğü gözlemlenmiştir. Eğitim esnasında veri kümesinin 64'e bölünmesi gradyan hesaplaması ve ağırlık değerlerinin güncellenme işlemlerini 16'ya ve 32'ye bölünmesine göre daha az veriyle yapması anlamına gelir. Bu durum işlem oranını azalttığı için eğitim süresinin azalmasını fakat kayıp değerinde artışa ve doğruluk oranında ise düşüşe sebep olur.

3.3. Epok Karşılaştırması (Comparison of Epoch)

Algoritmaların en iyi performansı gösterdiği hiper parametreleri bulmak adına ikinci yapılan deney epok kıyaslaması olmuştur. Bu aşamada yapılan deneylere Dengeli Veri Kümesi ile devam edilmiş olup, mini yığın boyutu her CNN mimarisi için bir önceki deneyde en iyi performansı veren değere ayarlanmıştır. Öğrenme oranı 0,001 ve L2 düzenleme oranı 0,005 olarak tüm mimariler için ayarlanmıştır. Epok sayısı ise 3, 5 ve 10 olacak şekilde üç seviyede incelenmiştir. Eğitim sonuçları Tablo 5'te paylaşılmıştır.

Bu deneyde sonucunda tüm algoritmalar en iyi performansı için 5 epokta göstermiştir. Genel olarak epok sayısının artırılması performansı etkilememişken eğitim süresini ciddi oranlarda arttırarak zaman kaybına neden olduğu tespit edilmiştir. Daha hızlı bir eğitim ihtiyacı olunan durumlarda ise belli miktarda performans kaybı kabul edilerek 3 epok sayısının da yeterli olabileceği gözlemlenmiştir.

3.4. Öğrenme Oranı Karşılaştırması (Comparison of Learning Rate)

Algoritmaların en iyi performansı gösterdiği hiper parametreleri bulmak adına üçüncü yapılan deney öğrenme oranı kıyaslaması olmuştur. Bu aşamada yapılan deneylere Dengeli Veri Kümesi ile devam edilmiş olup, eğitimler 5 epok yapılmıştır. Mini yığın boyutu her CNN mimarisi için bir önceki deneyde en iyi performansı veren değere ayarlanmıştır. L2 düzenleme oranı tüm mimariler için 0,005 olarak ayarlanmıştır. Öğrenme oranı ise 0.0001, 0.001 ve 0.01 olacak şekilde üç seviyede incelenmiştir. Eğitim sonuçları Tablo 6'da paylaşılmıştır. Bu deneyde optimum öğrenme oranının CNN mimarilerine göre değişkenlik gösterdiği gözlemlenmiştir. Öğrenme oranı, bir modelin eğitimi sırasında her ağırlık güncellemesinin model tarafından ne kadar öğrenilmiş olacağını belirleyen bir faktördür. Bu oran, her epokta ağırlıkların ne kadar güncelleneceğini kontrol eder. Yani, bir ağırlık güncelleme adımının boyutunu belirler. GoogleNet mimarisinde öğrenme oranını arttırmak ağırlıkların daha hızlı bir şekilde değişmesine rağmen sınıflandırma doğruluğunu arttırırken diğer üç mimaride öğrenme oranını arttırmak ve azaltmanın performansı düşürdüğü gözlemlenmiştir.

3.5. L2 Düzenleme Oranı Karşılaştırması (Comparison of L2 Regularization Rate)

Algoritmaların en iyi performansı gösterdiği hiper parametreleri bulmak adına dördüncü yapılan deney L2 düzenleme oranı kıyaslaması olmuştur. Bu aşamada yapılan deneylere Dengeli Veri Kümesi ile devam edilmiş olup, eğitimler 5 epok süresi boyunca yapılmıştır. Mini yığın boyutu ve öğrenme oranı, her CNN mimarisi için bir önceki deneyde en iyi performansı veren değerlere ayarlanmıştır. L2 düzenleme oranı ise 0.0001, 0.0005 ve 0.0009 olacak şekilde üç seviyede incelenmiştir. Eğitim sonuçları Tablo 7'de paylaşılmıştır.

Tablo 5. Epok sayısına göre doğruluk oranı (Accuracy rate according to epoch)

Algoritma	Veri Seti	Mini Yığın Boyutu	Epok	Zaman (min)	Doğruluk (%)
SqueezeNet	Dengeli	16	3	97,55	99,90
		16	5	214,25	99,96
		16	10	383,15	99,96
GoogleNet	Dengeli	16	3	152,10	99,94
		16	5	244,13	99,97
		16	10	476,42	99,97
ResNet50	Dengeli	16	3	280,58	99,98
		16	5	461,02	99,99
		16	10	911,42	99,99
AlexNet	Dengeli	64	3	26,40	99,88
		64	5	42,52	99,95
		64	10	84,42	99,95

Tablo 6. Öğrenme oranına göre doğruluk oranı (Accuracy rate according to learning rate)

Algoritma	Veri Seti	Mini Yığın Boyutu	Epok	Öğrenme Oranı	Zaman (min)	Doğruluk (%)
SqueezeNet	Dengeli	16	5	0,0001	197,30	99,91
		16	5	0,001	214,25	99,96
		16	5	0,01	201,43	2,33
GoogleNet	Dengeli	16	5	0,0001	336,57	99,86
		16	5	0,001	244,13	99,97
		16	5	0,01	214,46	99,98
ResNet50	Dengeli	16	5	0,0001	458,23	99,93
		16	5	0,001	461,02	99,99
		16	5	0,01	462,52	99,98
AlexNet	Dengeli	64	5	0,0001	39,35	99,67
		64	5	0,001	42,52	99,95
		64	5	0,01	38,11	92,36

Tablo 7. L2 Düzenleştirme oranına göre doğruluk oranı (Accuracy rate according to L2 regularization rate)

Algoritma	Veri Seti	Mini Yığın Boyutu	Öğrenme Oranı	L2 Düzenleştirme Oranı	Zaman (min)	Doğruluk (%)
SqueezeNet	Dengeli	16	0,001	0,0001	179,28	99,96
		16	0,001	0,0005	214,25	99,96
		16	0,001	0,0009	177,44	99,94
GoogleNet	Dengeli	16	0,01	0,0001	203,38	99,95
		16	0,01	0,0005	214,46	99,98
		16	0,01	0,0009	210,21	99,90
ResNet50	Dengeli	16	0,001	0,0001	464,01	99,98
		16	0,001	0,0005	461,02	99,99
		16	0,001	0,0009	463,45	99,99
AlexNet	Dengeli	64	0,001	0,0001	39,24	99,92
		64	0,001	0,0005	42,52	99,95
		64	0,001	0,0009	38,07	99,94

Tablo 8. Önerilen sistemin doğruluk oranı (Accuracy rate of proposed system)

Algoritma	Mini Yığın Boyutu	Epok	Öğrenme Oranı	L2 Düzenleştirme Oranı	Zaman (min)	Doğruluk (%)	Duyarlılık (Recall)	Kesinlik (Precision)	F1 Score
SqueezeNet	16	5	0,001	0,0001	179,28	99,9638	99,9638	99,964	99,9639
GoogleNet	16	5	0,01	0,0005	214,46	99,9845	99,9845	99,98452	99,98451
ResNet50	16	5	0,001	0,0005	461,02	99,99483	99,99483	99,99484	99,994837
AlexNet	64	5	0,001	0,0005	42,52	99,9483	99,9485	99,9483	99,9484

Tablo 8. Önerilen sistemin doğruluk oranı (Accuracy rate of proposed system)

Algoritma	Mini Yığın Boyutu	Epok	Öğrenme Oranı	L2 Düzenleştirme Oranı	Zaman (min)	Doğruluk (%)	Duyarlılık (Recall)	Kesinlik (Precision)	F1 Score
SqueezeNet	16	5	0,001	0,0001	179,28	99,9638	99,9638	99,964	99,9639
GoogleNet	16	5	0,01	0,0005	214,46	99,9845	99,9845	99,98452	99,98451
ResNet50	16	5	0,001	0,0005	461,02	99,99483	99,99483	99,99484	99,994837
AlexNet	64	5	0,001	0,0005	42,52	99,9483	99,9485	99,9483	99,9484

L2 düzenleştirme, ağır ağırlıklarının büyüklüğünü kontrol etmek ve aşırı öğrenmeyi engellemek amacıyla kullanılan bir yöntemdir. L2 düzenleştirme oranı arttıkça ağırlıkları daha fazla küçültür bu durum basit öznitelikleri öğrenmesi anlamına geldiği için performansın düştüğü gözlemlenmiştir. Oranın artışı ise ağırlıkları büyüttüğü için daha karmaşık özniteliklerin öğrenilmesini sağlar fakat genelleme yeteneği düşeceği için SqueezeNet mimarisi dışında sınıflandırma performansını olumsuz yönde etkilemiştir.

Şu ana kadar yapılan testlerin doğruluk oranı ve zaman kriterlerine göre GTSRB veri seti kullanılarak dört algoritmaya ait optimum hiper parametreler bulunmuştur. Çalışmada kullanılan mimariler arasında ise en başarılı performansa sahip CNN mimarisi %99,99 sınıflandırma doğruluğuna sahip ResNet50 olarak belirlenmiştir. Önerilen sistemler Tablo 8’de paylaşılmıştır.

3.6. Arka Planlı Veri Kümesi ile Yapılan Eğitimlerin Sonuçları (Results of Training with Background Dataset)

En iyi performansı gösterdiği hiper parametreleri bulunan algoritmalar için gürültü yoğunluğu yüksek görseller ile eğitim yapıldığında etkisinin incelenmesi amacıyla optimum hiper parametreler değiştirilmeden dengeli veri kümesinden 50 adet görsel rastgele çıkarılmış ve yerlerine gerçek zamanlı çekilmiş arka planlı gürültü yoğunluğu yüksek 50 adet görsel eklenmiştir. Bu 50 görselden 48 adedi eğitim sırasında, 2 adedi ise test sırasında kullanılmak üzere

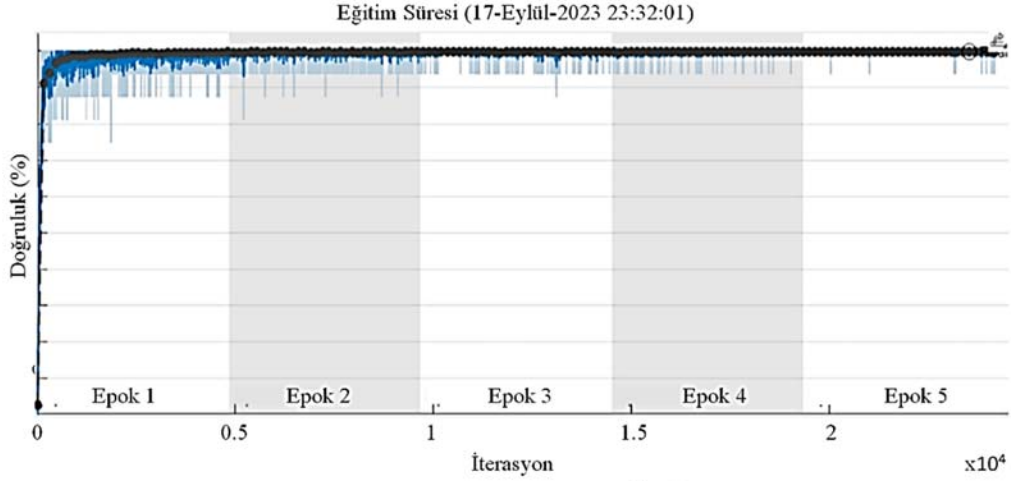
ayarlanmıştır. Algoritmaların bu deneyde elde ettiği sonuçlar Tablo 9’da verilmiştir.

Bu deneyin sonucunda ise gürültü yoğunluğu yüksek görsellerin veri setinde bulunması durumunda en az etkilenen CNN mimarisi ResNet50 olurken, en çok etkilenen ise SqueezeNet’tir. Bu deneyde en başarılı sonuçlara ulaşan ResNet50’nin öğrenim süreci Şekil 5’te, karışıklık matrisi ise Şekil 6’da verilmiştir.

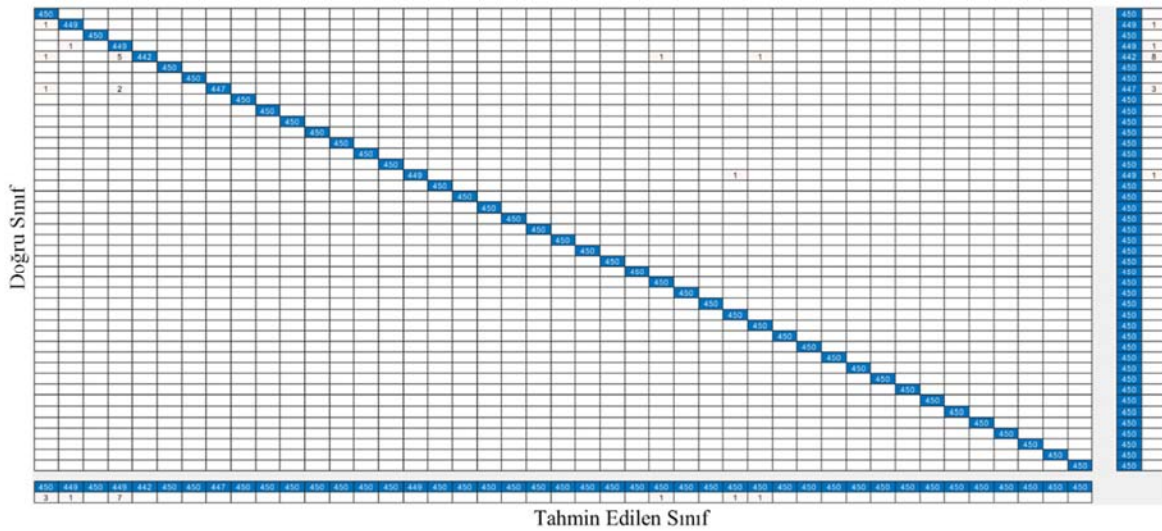
Şekil 6’da verilen karışıklık matrisi; Doğru Sınıf için yukarıdan aşağıya doğru, Tahmin Edilen Sınıf için soldan sağa doğru sırasıyla ‘maksimum 100 km/h hız limiti’, ‘maksimum 120 km/h hız limiti’, ‘maksimum 20 km/h hız limiti’, ‘maksimum 30 km/h hız limiti’, ‘maksimum 50 km/h hız limiti’, ‘maksimum 60 km/h hız limiti’, ‘maksimum 70 km/h hız limiti’, ‘maksimum 80 km/h hız limiti’, ‘80 km/h hız limiti sonu’, ‘sadece ileri’, ‘ileri veya sola dön’, ‘ileri veya sağa dön’, ‘bisiklet yolu’, ‘sol viraj’, ‘sağ viraj’, ‘girilmez’, ‘çift viraj’, ‘geçilemeyen bölgenin sonu’, ‘turlar için geçilemeyen bölgenin sonu’, ‘önceki sınırlamaların sonu’, ‘genel tehlike’, ‘buzlanma’, ‘geçiş yok’, ‘3.5 ton ve üzeri araçlar için geçiş yok’, ‘soldan geçiş’, ‘sağdan geçiş’, ‘yayaların geçişi yasak’, ‘öncelik’, ‘öncelikli yol’, ‘yol kapalı’, ‘yol daralır’, ‘yol çalışması’, ‘dönel kavşak’, ‘ıslak veya kirli kaygan’, ‘dur’, ‘ileride trafik lambası’, ‘ileriden sola dön’, ‘ileriden sağa dön’, ‘engebeli yol’, ‘3.5 ton ve üzeri araçlar yasak’, ‘çocuklara dikkat’, ‘vahşi hayvan geçişi’ ve ‘yol ver’ trafik işaretlerini temsil eden sınıflardır.

Tablo 9. Arka planlı veri kümesi doğruluk oranı (Accuracy rate of background dataset)

Algoritma	Mini Yığın Boyutu	Epok	Öğrenme Oranı	L2 Düzenleme Oranı	Zaman (min)	Doğruluk (%)	Duyarlılık (Recall)	Keskinlik (Precision)	F1 Score
SqueezeNet	16	5	0,001	0,0001	163,58	99,7571	99,7571	99,7583	99,8349
GoogleNet	16	5	0,01	0,0005	217,33	99,8036	99,8036	99,8042	99,8039
ResNet50	16	5	0,001	0,0005	469,23	99,9276	99,9276	99,9282	99,9279
AlexNet	64	5	0,001	0,0009	40,07	99,8346	99,8346	99,8353	99,8349



Sonuçlar		Eğitim Döngüsü	
Validasyon Doğruluğu	%99.93	Epok	5
Eğitim Tamamlanması	Tüm epoklar tamamlandı	İterasyon	24185
Eğitim Zamanı		Her epoktaki iterasyon	4837
Başlangıç zamanı	17-Eylül-2023 23:32:01	Maksimum iterasyon	24185
Geçen zaman	469 min 23 s	Diğer Bilgiler	
Doğrulama Sıklık	150 iterasyon	Donanım kaynağı	Sadece GPU
		Öğrenme oranı planı	Sabit
		Öğrenme Oranı	0.001

Şekil 5. ResNet50 öğrenim süreci (Training progress of ResNet50)**Şekil 6.** ResNet50 karışıklık matrisi (Confusion Matrix of ResNet50)

Tablo 10. Gerçek zamanlı test sonuçları (Real time test results)

Algoritma	Test Verisi Doğruluk Oranı (%)					Toplam
	1. Görüntü	2. Görüntü	3. Görüntü	4. Görüntü	5. Görüntü	
SqueezeNet	90,70	93,02	93,02	93,02	90,70	92,09
GoogleNet	83,72	88,37	95,35	90,70	95,35	90,70
ResNet50	93,02	95,35	95,35	90,70	93,02	93,49
AlexNet	79,07	81,40	74,42	76,74	79,07	78,14

Bu bölümde algoritmalar, her sınıf beş adet test trafik işareti görseli olacak şekilde toplamda 215 adet trafik işareti görseli, bölüm 2.6.'da detaylarının anlatıldığı kullanıcı arayüzü ile gerçek zamanlı olarak test edilmiştir. Kullanılan kullanıcı arayüzü ve test düzeneği, algoritmaların görselleri canlı bir ortamda işlermiş gibi sunulmasını sağlar. Bu düzenek, algoritmaların anlık karar verme süreçlerini değerlendirmek için tasarlanmıştır. Her trafik işareti görseli sırasıyla kullanıcı arayüzüne sunulmuştur ve her algoritma bu görselleri sınıflandırmıştır. Görseller, gerçek zamanlı karşılaştıra durumunu performanslarının değerlendirilmesine olanak tanımaktadır. Arka plan yoğunluğunun algoritmaların performansına etkisini gözlemlemek amacıyla bu çeşitlilik özellikle seçilmiştir. Gerçek zamanlı testler sırasında, algoritmaların her bir görseli doğru sınıflandırma oranları ayrı ayrı kaydedilmiş ve Tablo 10'da sunulmuştur. Tablo 10, her bir algoritmanın her sınıfta bulunan beş farklı görsel üzerindeki doğruluk oranlarının ortalamasını göstermektedir.

Yapılan bu çalışmanın sonuçlarına göre, gerçek zamanlı bir ortamda en başarılı algoritmanın ResNet50, en az başarılı algoritmanın ise AlexNet olduğu belirlenmiştir. Bu farklılığın temel sebeplerinden biri, ResNet50'nin gürültü yoğunluğu yüksek verilere uyum sağlama yeteneğinin AlexNet'e kıyasla daha güçlü olmasıdır. ResNet50, katmanlı yapısı ve derin öğrenme kapasitesi sayesinde, karmaşık arka planlar ve yüksek gürültü içeren görüntülerde bile yüksek doğruluk oranları sağlayabilmektedir. AlexNet ise daha basit bir mimariye sahip olduğu için, bu tür karmaşık ve gürültülü verilerde performans kaybı yaşamaktadır. Ayrıca, önceki çalışmalara dayanarak; aynı şartlarda ve aynı veri setleri üzerinde yapılan eğitim süreçlerinde, ResNet50'nin genellikle AlexNet'e göre daha başarılı sonuçlar verdiği gözlemlenmiştir. Bazı kısıtlar altında bu gerçek zamanlı testler gerçekleştirilmiştir. Araç hızının ve işaretçinin bulunduğu konumun yaratacağı etkiler dikkate alınmamıştır. Bu çalışma, gerçek zamanlı benzetimi sağlamak için sabit hızda ve belirli konumlarda görsellerin uygulanmasını içermektedir. Gelecek çalışmalarda bu etkiler de dikkate alınarak çalışmalar gerçekleştirilebilir.

4. Sonuçlar (Conclusions)

Bu makalede, 4 farklı CNN algoritması kullanılmıştır ve bu algoritmalar GTSRB veri setiyle eğitilmiştir. Daha sonrasında geliştirilen kullanıcı arayüzüyle gerçek zamanlı bir sistemde tahmin başarısı değerlendirilmiştir. Algoritmaların gerçek zamanlı bir sistemde tahmin başarısını değerlendirmeden önce, yapılan deneylerde algoritmaların en iyi performans gösterdiği hiper parametreler belirlenmiştir. Aynı zamanda veri artırma teknikleri kullanılarak oluşturulan veri kümeleri arasında, en iyi performansı gösterdiği veri kümesi tespit edilmiştir. Yapılan deneyler sonucunda algoritmalar, en iyi performansını veri kümesi bazında her sınıfa görüntü verisinin homojen olarak dağıldığı Dengeli Veri Kümesi ile yapılan eğitimlerde göstermiştir. ResNet50, %99,99 oranı ile en iyi sınıflandırma performansına ulaşmıştır. Bu sonuçlara göre CNN ile

görüntü verilerinin sınıflandırmasında veri setinde bulunan sınıflarda verilerin dengeli dağılımı performansı olumlu yönde etkilediği gözlemlenmiştir. Aynı zaman da her algoritma için kullanılan hiper parametre değerleri değişiklik göstermekte ve performans etkilenmektedir. Bu sebeple klasik CNN algoritmaları ile yapılacak gerçek zamanlı bir testten önce algoritmalar için en uygun hiper parametreler belirlenmelidir. GTSRB veri setinde bulunan görüntülerin gürültü yoğunluğu düşük olduğu tespit edilerek, gerçek zamanlı bir uygulamada, bu veri setiyle eğitilmiş algoritmaların başarılı sınıflandırma yapamayacağı öngörülmüştür. Bu nedenle veri kümesine arka plan gürültü yoğunluğu yüksek trafik işareti görüntüleri eklenerek Arka Planlı Veri Kümesi oluşturulmuş, algoritmalar bu veri kümesiyle eğitilmiş ve gürültüden en az etkilenen algoritma tespit edilmiştir. Yapılan deneyler sonucunda, en iyi sınıflandırma performansını %99,93 oranı ile ResNet50 göstermiştir. Bu sayede ResNet50, arka plan yoğunluğu yüksek yani gürültülü görüntü verilerini öğrenen en iyi algoritma olarak belirlenmiştir. Bu sonuçlara göre veri setine arka plan gürültü yoğunluğu yüksek verilerin eklenmesi performansı küçük bir oranda düşürmesine rağmen gürültüye uyumluluğu arttırdığı gözlemlenmiştir. Gerçek zamanlı bir test sistemi için oluşturulan kullanıcı arayüzünde, Arka Planlı Veri Kümesi ile eğitilmiş 4 algoritma, her sınıfa ait 5 trafik işareti görüntüsüyle test edilmiş ve en iyi sınıflandırma performansını gösteren algoritma belirlenmiştir. Oluşturulan test ortamında, gerçek zamanlı bir sisteme yakın olacak şekilde ışık koşulları değişkenlik göstermektedir. Yapılan deneyler sonucunda en iyi tahmin başarısını %93,49 oranla ile ResNet50 göstermiştir. Bu sayede ResNet50, gerçek zamanlı bir sistemde gürültü yoğunluğu yüksek görüntülerin derin öğrenme metoduyla öğrenilmesi için en iyi algoritma olarak belirlenmiştir. Çıkan bu sonuçlarla birlikte araçlarda kullanacak trafik işaretini tespit etme probleminde çözüm sunacak bir gelişmiş sürücü destek sisteminin sınıflandırma algoritması olarak en uygun ResNet50 algoritması olarak belirlenmiştir. Eğitim süreleri göz önüne alındığında ise ResNet50 algoritmasından çok daha az işlem yoğunluğuna sahip olan SqueezeNet algoritması gerçek zamanlı yapılan teste elde ettiği %92,09 sınıflandırma doğruluğu ile alternatif olarak kullanılabilirliği öngörülmektedir. İlerleyen dönemlerde yapılacak olan çalışmalarda, bu tez çalışmasında eğitilen ResNet50 algoritması kullanılarak bir sürücü destek sistemi oluşturulması planlanmaktadır. Bu sistemin, araç içerisine yerleştirilen araç içi kamerasından alınan görüntüleri kullanarak, bu tezde önerilen ve en yüksek sınıflandırma doğruluğuna sahip ResNet50 algoritmasını temel olarak yol taraması gerçekleştirmesi hedeflenmektedir. Bu tarama sonucunda elde edilen trafik işareti bilgileri, sürücüye, dikkat dağıtıcı olmayan bir şekilde yansıtılacaktır. Bu çalışmada gece trafiğine ait görüntüler ele alınmamıştır ve bu durum, çalışmanın kısıtları arasında yer almaktadır. Bu sebeple gelecek çalışmalarda, gece trafiğinde elde edilen görüntülerin de veri setine eklenerek, sistemdeki performansın daha da iyileştirilebileceği planlanmaktadır.

Teşekkür (Acknowledgement)

Bu kapsamlı çalışmanın ışığında, her adımda sağlam bir rehber ve destek olan kıymetli hocam Doç. Dr. Selda Güney'e en içten teşekkürlerimi sunarım. Yol gösterici bilgelikleriniz, sabrınız ve samimiyetiniz, bu zorlu süreçte benim için bir ışık kaynağı oldu.

Kaynaklar (References)

1. Canese, L., Cardarilli, G.C., Di Nunzio, L., Fazzolari, R., Famil Ghadakchi, H., Re, M., Spanò, S., Sensing and Detection of Traffic Signs Using CNNs: An Assessment on Their Performance, *Sensors*, 22 (8830), 1-29, 2022.
2. Natarajan, S., Annamraju, A.K., Baradkar, C.S., Traffic Sign Recognition Using Weighted Multiconvolutional, *IET Intelligent Transportation Systems*, 12 (10), 1396-1405, 2018.
3. Bouaafia, S., Messaoud, S., Maraoui, A., Deep Pre-trained Models for Computer Vision Applications: Traffic Sign Recognition, 18th International Multi-Conference on Systems, Signals & Devices (SSD), Monastir, Tunisia, 24-27, 2021.
4. Li, W., Li, D., Zeng, S., Traffic Sign Recognition with a Small Convolutional Neural Network, 2019 IOP Conf. Ser.: Mater. Sci. Eng., 688, 1-7, 2019.
5. Sokipriala, J., Orike, S., Traffic Sign Classification Comparison Between Various Convolution Neural Network Models, *Int. J. Sci. Eng. Res.*, 12 (7), 165-171, 2021.
6. Kou, A., Detection and Recognition of Traffic Signs Based on Improved Deep Learning, *International Core Journal of Engineering*, 6 (12), 208-213, 2020.
7. Zheng, Y., Jiang, W., Evaluation of Vision Transformers for Traffic Sign Classification, *Wireless Communications and Mobile Computing*, 2022 (1), 3041117, 2022.
8. Htun, T. Z., Oo, A. N., Winn, K. Z. N., Myint, M. T., Comparing Performance for Myanmar Road Signs Recognition, 2023 IEEE Conference on Computer Applications (ICCA), Egypt, 27-32, 2023.
9. Rashid, S. I., Islam, A., Hasan, A. M., Traffic Sign Recognition by Integrating Convolutional Neural Network and Support Vector Machine, *International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2)*, Rajshahi, Bangladesh, 2019.
10. Schuszter, I. C., A Comparative Study of Machine Learning Methods for Traffic Sign Recognition, *Proc. 2017 19th Int. Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, Timisoara, Romania, 390-392, 2017.
11. Alshami, M., Traffic Signs Classification with Transfer Learning of CNN Based Models and Performance Comparison, *Yüksek Lisans Tezi, Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Kayseri*, 2023.
12. Balkan, E., Traffic Signs Recognition System for Autonomous Vehicles, *Yüksek Lisans Tezi, İstanbul Arel Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul*, 2022.
13. Zam, M., Trafik İşaretlerini Tanıyan Bir Sürücü Güvenlik Destek Sistemi, *Yüksek Lisans Tezi, Bursa Uludağ Üniversitesi, Fen Bilimleri Enstitüsü, Bursa*, 2019.
14. Aysal, F. E., Yıldırım, K., & Cengiz, E., Real-Time Application of Traffic Sign Recognition Algorithm with Deep Learning, *Journal of Materials and Mechatronics: A*, 3 (2), 275-289, 2022.
15. Mishra, J., Goyal, S., An Effective Automatic Traffic Sign Classification and Recognition with Deep Convolutional Networks, *Multimedia Tools and Applications*, 81 (13), 18915-18934, 2022.
16. Chen, J., Jia, K., Chen, W., Lv, Z., & Zhang, R. A real-time and high-precision method for small traffic-signs recognition. *Neural Computing and Applications*, 34, 2233-2245, 2021.
17. Özçevik Y., Solmaz Ö., Baysal E., Ökten M., A real-time simulation environment architecture for autonomous vehicle design, *Journal of the Faculty of Engineering and Architecture of Gazi University* 38 (3), 1867-1878, 2023.
18. Rho E.J., Kaggle, German Traffic Sign Recognition Benchmark (Cropped), <https://www.kaggle.com/datasets/eunjurho/german-traffic-sign-recognition-benchmark-cropped>, Yayın Tarihi: 2019, Erişim Tarihi: 2023.
19. Kocakanat, K., Serif, T., Turkish Traffic Sign Recognition: Comparison of Training Step Numbers and Lighting Conditions, *Avrupa Bilim ve Teknoloji Dergisi*, 28, 1469-1475, 2021.
20. Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., Terzopoulos, D., Image Segmentation Using Deep Learning: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44 (7), 3523-3542, 2022.
21. Veličković, N., Stojković, Z., Dimić, G., Miletić, D., Vasiljević, J., Nagamalai, D., Traffic Sign Classification Using Convolutional Neural Network, *Figshare*, doi: 10.6084/m9.figshare.6652256.v1, 2018.
22. Tümen, V., Yıldırım, Ö., Ergen, B., A Convolutional Neural Network Model for Road Flow Direction Detection, *Journal of Intelligent Systems with Applications (Akıllı Sistemler ve Uygulamaları Dergisi)*, 2 (2), 94-99, 2019.
23. Üzen H., Türkoğlu M., Arı A., Hanbay D., InceptionV3 based enriched feature integration network architecture for pixel-level surface defect detection, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 38 (2), 721-732, 2023.
24. Öztürk, G., Derin Evrimsel Sinir Ağları Kullanılarak Araç, İnsan ve Trafik İşaretlerinin Tanınması, *Mechatronic Eng. M.Sc. Thesis, Sakarya University, Sakarya, Turkey*, 2020.
25. Xiao, T., Liu, L., Li, K., Qin, W., Yu, S., Li, Z., Comparison of Transferred Deep Neural Networks in Ultrasonic Breast Masses Discrimination, *BioMed Research International*, 2018.
26. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., Dropout: A Simple Way to Prevent Neural Networks from Overfitting, *Journal of Machine Learning Research*, 15, 1929-1958, 2014.
27. Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Hasan, M., Van Essen, B. C., Awwal, A. A. S., & Asari, V. K., The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches, *arXiv preprint arXiv:1803.01164*, 2018.
28. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., Going Deeper with Convolutions, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-9, 2015.
29. İnik, Ö., Ülker, E., Derin Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri, *GBAD (Görüntü Bilimi ve Analizi Dergisi)*, 6 (3), 85-104, 2017.
30. He, K., Zhang, X., Ren, S., Sun, J., Deep Residual Learning for Image Recognition, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 770-778, 2016.
31. Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., Keutzer, K., SqueezeNet: AlexNet-Level Accuracy with 50x Fewer Parameters and <0.5MB Model Size, *arXiv preprint arXiv:1602.07360*, 2016.
32. Polat, H., Özerdem, M. S., Derin Transfer Öğrenimi Yaklaşımı ile Kamusal Alanda Medikal Maske Kullanımının Otomatik Kontrolü, *Türk Doğa ve Fen Dergisi*, 10 (2), 191-198, 2021.
33. Ucar, F., Korkmaz, D., COVIDiagnosis-Net: Deep Bayes-SqueezeNet Based Diagnosis of the Coronavirus Disease 2019 (COVID-19) from X-ray Images, *Medical Hypotheses*, 140, 109761, 2020.
34. Sayed, G. I., Soliman, M. M., & Hassani, A. E., A Novel Melanoma Prediction Model for Imbalanced Data Using Optimized SqueezeNet by Bald Eagle Search Optimization, *Computers in Biology and Medicine*, 136, 104712, 2021.