AUSUD
BANDIRMA
ONYEDİ EYLÜL
ÜNİVERSİTESİ
JITSA

Research Article

Traffic sign classification for autonomous vehicles using convolutional neural networks

Mehmet Özcan[1,*], Abdurrahim Hüseyin Ezirmik[1]
[1] Department of Computer Engineering, Faculty of Engineering, Balıkesir University, Balıkesir, Turkey
*Correspondence: *mehmet.ozcan@balikesir.edu.tr*

DOI: 10.51513/jitsa.1746494

Abstract: Recognition of traffic signs is one of the key activities in the development of autonomous vehicles for safe navigation on the roads. This work addresses the study of ConvNet in classifying Turkish traffic signs into two classes: hazard-warning signs and regulatory signs. A dataset of 129 traffic sign images, enriched with hue jitter transformations for data augmentation, was utilized to enhance model performance. The ConvNet, based on a three-convolution-layer architecture, four ReLU layers, and two fully connected layers, is trained to classify the two classes of traffic signs. The attained average accuracy was 97.7% ± 5.2% on the training set, 88.8% ± 1.2% on the validation set, and 96.9% ± 7.2% on the test set. These results underscore the capability of ConvNets in identifying and classifying traffic signs, thus proving that they can be utilized in autonomous vehicle technologies. Future research could use real-world photos of traffic signs to test the model's applicability.

Keywords: Machine learning, deep learning, convolutional neural networks, computer vision, image recognition, traffic signs

Otonom araçlar için evrişimsel sinir ağları kullanılarak trafik işareti sınıflandırması

Özet: Trafik işaretlerinin tanınması, yollarda güvenli navigasyon için otonom araçların geliştirilmesinde temel faaliyetlerden biridir. Bu çalışma, ConvNet'in Türk trafik işaretlerini tehlike uyarı işaretleri ve düzenleyici işaretler olmak üzere iki sınıfa ayırma çalışmasını ele almaktadır. Model performansını artırmak için renk tonu titreme dönüşümleriyle zenginleştirilmiş 129 trafik işareti görüntüsünden oluşan bir veri kümesi kullanılmıştır. Üç evrişim katmanlı mimariye, dört ReLU katmanına ve iki tam bağlı katmana dayanan ConvNet, iki trafik işareti sınıfını sınıflandırmak üzere eğitilmiştir. Elde edilen ortalama doğruluk, eğitim setinde %97,7 ± %5,2, doğrulama setinde %88,8 ± %1,2 ve test setinde %96,9 ± %7,2 olmuştur. Bu sonuçlar, ConvNet'lerin trafik işaretlerini tanımlama ve sınıflandırma konusundaki yeteneklerini vurgulayarak, otonom araç teknolojilerinde kullanılabileceğini kanıtlamaktadır. Gelecekteki araştırmalarda, modelin uygulanabilirliğini test etmek için trafik işaretlerinin gerçek dünya fotoğrafları kullanılabilir.

Anahtar Kelimeler: Makine öğrenmesi, derin öğrenme, evrişimsel sinir ağları, bilgisayarla görü, görüntü tanıma, trafik işaretleri

---

[*] Corresponding author.

ORCID: 0000-0001-8250-3143, 0000-0002-1154-1537 (in hierarchical order, added after if manuscript is accepted for publication)
Received 20.07.2025; Received in revised form 10.10.2025; Accepted 14.10.2025.

APA Citation Info: Özcan, M., & Ezirmik, A. H. (2025). Traffic sign classification for autonomous vehicles using convolutional neural networks. Akıllı Ulaşım Sistemleri ve Uygulamaları Dergisi, 8(2), 285-294. https://doi.org/10.51513/jitsa.1746494

1. Introduction

Autonomous vehicles are a type of automobile capable of perceiving their environment and making decisions without human intervention. These vehicles offer numerous benefits that can significantly enhance the quality of life, including a reduction in traffic accidents, lower costs, improved safety, faster traffic flow, enhanced mobility for the elderly and disabled, and increased customer satisfaction (Bartneck et al., 2020). However, the widespread adoption of autonomous vehicles remains limited due to several challenges, including safety concerns, technological barriers, and legal regulations. This paper aims to explore autonomous vehicles from the perspectives of driving safety, pedestrian safety, and technological advancements.

Recent developments in artificial intelligence have accelerated the progress of autonomous vehicle technology, suggesting that such vehicles will become more prevalent in traffic in the near future. The technology behind autonomous vehicles operates by processing real-time sensory data gathered from advanced hardware and software systems. Key hardware components used in autonomous vehicle systems include cameras, computers, LiDAR, radar, and various sensors. These components work together to process sensor data efficiently, allowing for the detection and recognition of objects in the vehicle's surroundings, thereby enabling real-time situational awareness and decision-making.

Over the past few years, there is a remarkable increase in the number of vehicles worldwide. As the number of vehicles in the traffic increases, it becomes more difficult to manage the traffic flow. Therefore, traffic signs are placed on the roads to manage the traffic flow. Traffic signs are used to inform drivers about the condition of the road. Accidents can occur when drivers do not pay attention to or obey traffic signs (Bucsuházy et al., 2020; Kocakanat and Serif, 2021). Further, the effect of traffic signs on the motor behavior of drivers is not yet entirely understood. A better understanding of how individuals process the meaning of traffic signs could contribute to improved reaction times and better decision-making while driving. According to the literature, poorly designed cues can negatively affect driving behavior (Vilchez, 2019). Eradicating such human being perceptional or reactional mistakes, autonomous driving has the potential to significantly reduce the number of traffic accidents. Hereby, autonomous vehicles have been receiving increasing attention from both scientific community and the industrial sector. To ensure their effectiveness, it is essential to continue advancing the software and hardware systems of autonomous vehicles, particularly in areas such as traffic sign recognition.

This study addresses the problem of traffic sign recognition within the scope of Turkish traffic signs. Ensuring driver, pedestrian, and traffic safety, traffic regulatory signs and danger-warning signs are two crucial major groups of traffic signs in Turkiye. Traffic regulatory signs are created for the purpose of providing traffic regulations and they are generally round in shape and express restrictions, prohibitions, and necessities. On the other hand, danger-warning signs indicate problems and dangers that drivers may not easily notice while driving. The effective recognition and understanding of these signs are crucial for maintaining safe and efficient traffic flow.

Conventional machine learning methods have been widely used for traffic sign recognition and other real-world problems (Arora et al., 2022). However, deep learning techniques have become increasingly popular due to their ability to bypass the particularly challenging feature extraction phase. Deep learning methods are especially effective at uncovering complex patterns underlying difficult problems (Arul, 2021; Salvador, 2016; Zou, 2022).

Convolutional Neural Networks (ConvNets) are a prominent deep learning architecture due to their ability to automatically learn from raw data, making them highly effective for tasks such as image detection and recognition (Teuwen and Nikita, 2019). In addition to their success with visual data, ConvNets are also well-suited for classification tasks involving audio, time series, and signal data. Object detection, which involves locating specific objects within an image, is one of the key applications of ConvNets. A specialized variation, Regions with Convolutional Neural Networks (R-ConvNets), enhances detection accuracy by combining rectangular region proposals with ConvNet features (Avci et al., 2023). The R-ConvNet method operates in two phases: first, it identifies potential subregions that may contain an object, and second, it classifies the object within each identified region.

Traffic sign perception is a significant subject in computer vision, with notable research focused on the detection and recognition of traffic signs. While Turkish traffic signs closely resemble European traffic signs, significant efforts have been made to recognize Turkish traffic signs (Arslan et al., 2016; Çetin and Ortataş, 2021; Gezgin and Alkan, 2024; Gündüz et al., 2013; Kilic and Aydin, 2020; Kocakanat and Serif, 2021; Palandız et al., 2021; Uluskan, 2020; Waziry et al., 2024; Yaliç and Can, 2011; Yıldıran, 2019). This study focuses on training a ConvNet model to distinguish between Turkish traffic danger-warning signs and regulatory signs.

The remainder of this paper is organized as follows: Section 2 introduces the methods and data used in this study. Sections 3 and 4 present the experimental results and discussions. Finally, Section 5 provides the conclusions drawn from the study.

2. Materials and methods

In this section, we first explain the data used in the experiments. Afterward, we introduce in detail the deep learning method to be applied to the data. Besides, we introduce the performance criteria to evaluate the trained model.

2.1. Dataset

The highway standard traffic signs data used in this work have been obtained from the Turkish General Directorate of Highways. Turkish traffic signs resemble European traffic signs, but there are some differences as well. The two groups of highway standard traffic signs have been only chosen. These are the traffic danger-warning signs and the regulatory signs. The number of Turkish traffic danger-warning signs is 61 and the number of Turkish regulatory signs is 68. The total number of data is 129. Further, the sizes of the data differ from each other.

To prevent overfitting the model, we augment the collection of images. To this end, we have tried a variety of image warping, cropping, and color transformations, such as rotation, translation, scale, reflection, shear, cropping, hue jitter, saturation jitter, brightness jitter, contrast jitter, grayscale, and synthetic noise. But according to the experimental results, only the hue jitter transformation has improved model performance. Hue defines a color's position on a color wheel and ranges from 0 to 1. Colors can adjust from red to yellow, green, cyan, blue, purple, magenta, and back to red. Hue jitter sets the apparent shade of images. In this study, we have tuned the hue of the input images by a small positive offset selected randomly from the range [0.05, 0.15]. Figure 2 shows the hue-jitter transformations of the original images. Red colors become more orange or yellow. Orange colors seem yellow or green. Consequently, the size of the data set is 258 together with the augmented data.

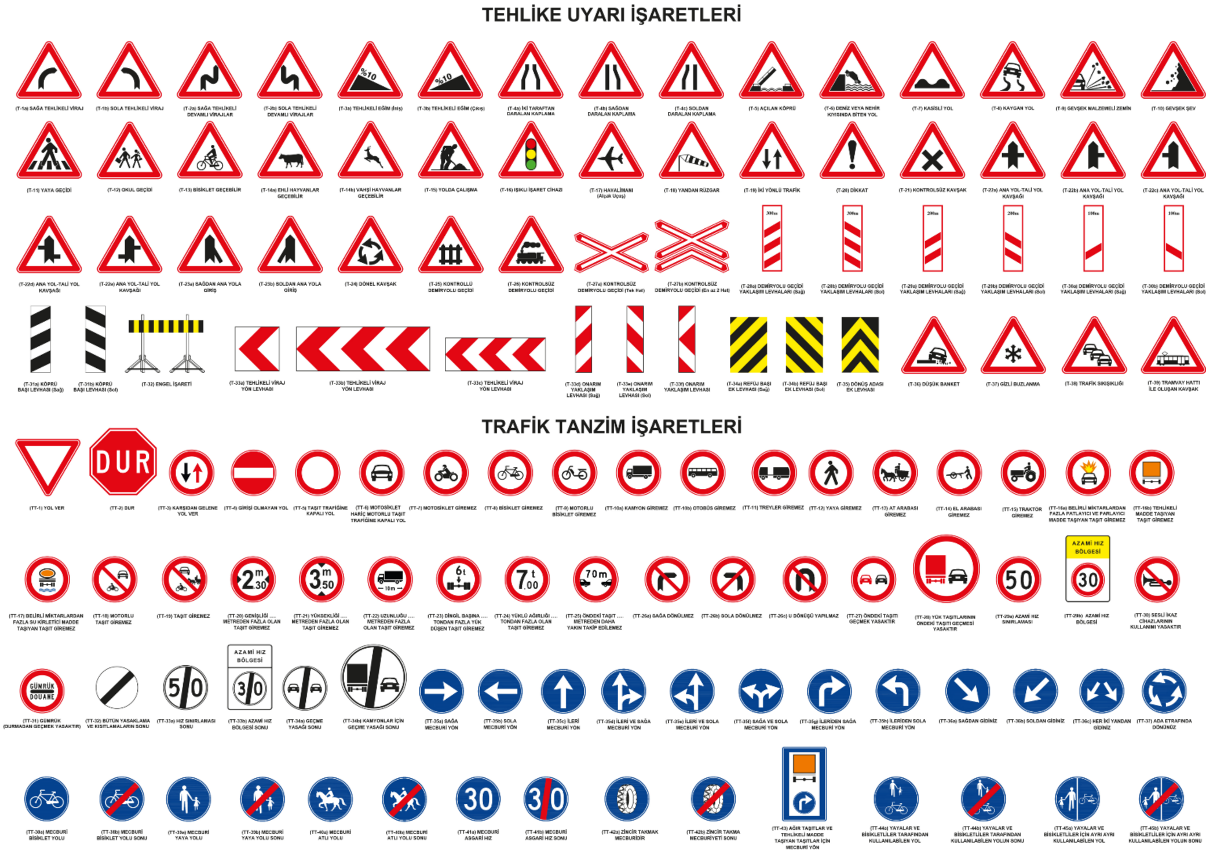

Figure 1. Turkish traffic danger-warning signs and the regulatory signs.

Figure 2. The hue-jitter transformations of the original images (The original pictures are at the top. The hue-jitter transformed pictures are in the middle and at the bottom. The hue ranges are [0.05 0.15] and [0.01 0.30], respectively).

2.2. Convolutional Neural Networks (ConvNets)

ConvNets are a key deep learning model and have the ability to learn from the raw data directly. They excel at detecting spatial patterns in images for objects recognition and show policies for 1-D signal data classification. ConvNets are in essence an input layer, an output layer and a number of hidden layers. Figure 3 shows the illustration of a ConvNet architecture (Wang et al., 2018). The layers fulfill processes that change the data with the intention of learning features regarding data. These layers apply transformations to the data, changing it in ways that are intended to enable them to learn features of the data. The widespread layers to be used in a convolutional neural network are convolutional layers, Rectified Linear Unit (ReLU) layers and pooling layers. ConvNets can exploit numerous layers in order to catch another feature of an image. Filters at different resolutions can be implemented in each image. The output of each convolved image is delivered as the input to the next layer. The filters can be described as features such as edges and brightness and later transformed into features that uniquely characterize the object (Altuntaş et al., 2022; Mahadik et al., 2023; Salman et al., 2022).

The convolution layer arranges the input images via a set of convolutional filters by enabling particular features from the images. The ReLU layer breaks linearity by replacing negative values with zero and keeping positive values. The pooling layer avoids variance by reducing the number of parameters that the network needs and conducting nonlinear subsampling. This process can be iterated along layers until each layer recognizes different features. Unlike a conventional neural network, ConvNets have the same weights and bias values for all hidden neurons in a certain layer. This means that all hidden neurons can obtain the same feature in different regions of the image. A piece of the final layer in the architecture of a ConvNet is a fully connected layer that outputs a vector of the number of classes and calculates the probabilities for each class of an image being classified. The final layer in the architecture of a ConvNet is the classification layer used to output the classes (Abdelkhalek and Mashaly, 2023; Al-Ali et al., 2023).

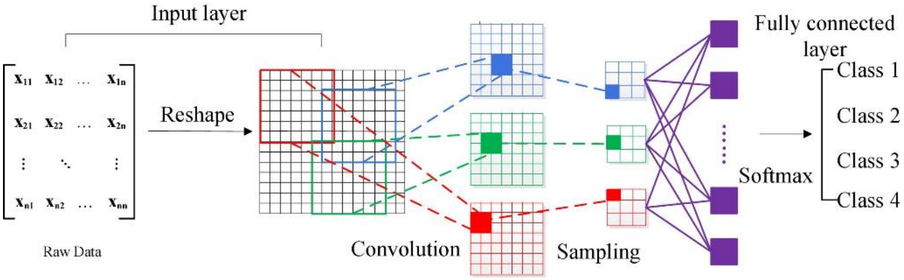

Figure 3. The illustration of a ConvNet architecture (Wang et al., 2018).

2.3. Designed Convolutional Neural Network Architecture

Table 1 shows the architecture of the ConvNet used in this work. The header information in Table 1 denotes the Number (layer order), Process (the process done in a layer), Type (layer type), Activation (activation output size), and Learnable (weights [W] and bias [B]). The ConvNet takes the 32×32 sized images as input. The middle layer of the ConvNet is composed of 3 convolutional layers whose sizes of the filters are 32, 32, and 64, respectively, 3 ReLU layers, and 2 max-pooling layers whose size of the pools is 2×2. The final layer of the ConvNet consists of 2 fully connected layers whose sizes of the outputs are 32 and 2, respectively, 1 ReLU layer, 1 softmax layer, and 1 classification layer whose outcome is with two classes. Furthermore, we initialize the first convolutional layer weights using normally distributed random numbers with a standard deviation of 0.001 so as to improve the convergence of training.

Table 1. The layer architecture of the ConvNet used in this work.

#	**Process**	**Type**	**Activation**	**Learnable**
1	32×32×3 images with zero center normalization	Image input	32×32×3	-
2	32 2×2×3 convolutions with stride [2 2] and padding [0 0 0 0]	Convolution	16×16×32	W: 2×2×3×32 B: 1×1×32
3	ReLU	ReLU	16×16×32	-
4	32 2×2×32 convolutions with stride [2 2] and padding [0 0 0 0]	Convolution	8×8×32	W: 2×2×32×32 B: 1×1×32
5	ReLU	ReLU	8×8×32	-
6	2×2 max pooling with stride [1 1] and padding [0 0 0 0]	Max pooling	7×7×32	-
7	64 2×2×32 convolutions with stride [2 2] and padding [0 0 0 0]	Convolution	3×3×64	W: 2×2×32×64 B: 1×1×64
8	ReLU	ReLU	3×3×64	-
9	2×2 max pooling with stride [1 1] and padding [0 0 0 0]	Max pooling	2×2×64	-
10	32 fully connected layer	Fully connected	1×1×32	W: 32×256 B: 32×1
11	ReLU	ReLU	1×1×32	-
12	2 fully connected layer	Fully connected	1×1×2	W: 2×32 B: 2×1
13	Softmax	Softmax	1×1×2	-
14	Cross entropy loss with classes ‘HazardWarningSigns’ and ‘TrafficRegulationSigns’	Classification Output	1×1×2	-

2.4. Evaluation Criteria

We use the classification accuracy given by Eq. (1) to evaluate the performance of the algorithms. The accuracy rate ranges from 0% to 100%. The larger accuracy rates point out higher performance. Furthermore, the data set has been divided into a training set, a validation set, and a test set as 80%, 10%, and 10%, respectively.

$$ACC = \frac{1}{m}\sum_{i=1}^{m} \delta(actual_i, prediction_i) \tag{1}$$

$$\delta(x, y) = \begin{cases} 1, & x = y \\ 0, & x \neq y \end{cases} \tag{2}$$

where $actual$ denotes the true class labels and $prediction$ indicates the predictions of a classifier.

3. Results

It is crucial to analyze features by monitoring which regions in the layers activate on the images and comparing them with the corresponding regions in the original images. Figure 4 shows the original image and activations of Layer 2, Layer 4, Layer 7, Layer 10, Layer 12, Layer 13, and Layer 14, respectively.

Moreover, it is also critical to discover superior channels by examining channels with big activations. Figure 5 shows the original image and the strongest activation of Layer 2, Layer 4, Layer 7, Layer 10, Layer 12, Layer 13, and Layer 14, respectively.

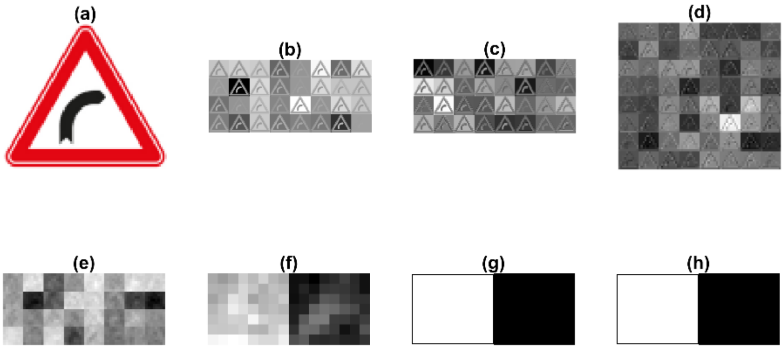

Figure 4. (a) Original image, Activations of (b) layer 2, (c) layer 4, (d) layer 7, (e) layer 10, (f) layer 12, (g) layer 13, (h) layer 14.

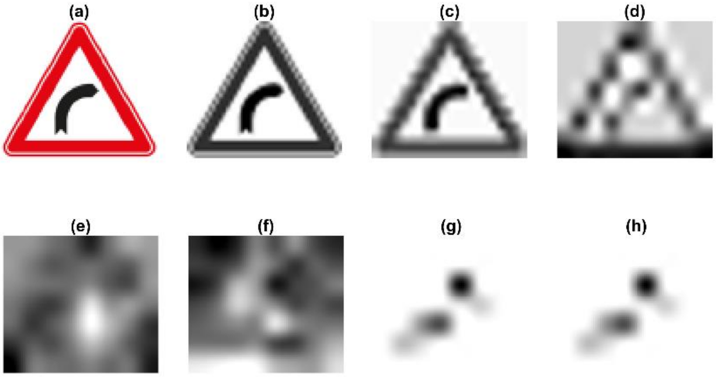

Figure 5. (a) Original image, The strongest activation of (b) layer 2, (c) layer 4, (d) layer 7, (e) layer 10, (f) layer 12, (g) layer 13, (h) layer 14.

Figure 6 shows the accuracy and loss of the ConvNet model on the training set and validation set. After the experiments have been repeated 10 times, the average training, validation, and test accuracy are 97.7% ± 5.2%, 88.8% ± 1.2%, and 96.9% ± 7.2%, respectively. Besides, Figure 7 shows the confusion matrix of the ConvNet model predictions on the test set. According to the results, the ConvNet model slightly misclassifies the 'HazardWarningSigns' class, resulting in a 5.9% error rate.

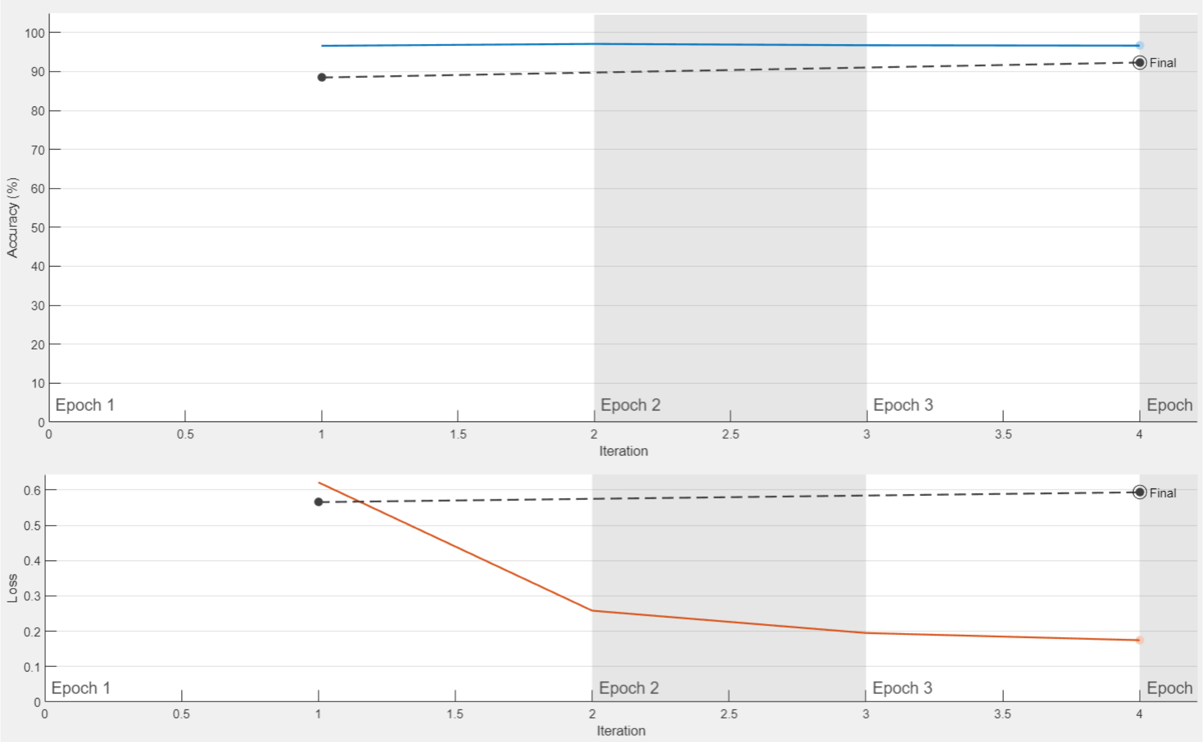

Figure 6. The accuracy and loss of the ConvNet model on the training set and validation set (the blue line denotes the smoothed training accuracy, the light blue line denotes the training accuracy, the red line denotes the smoothed training loss, the light red line denotes the training loss, and the black line denotes the validation accuracy and loss).

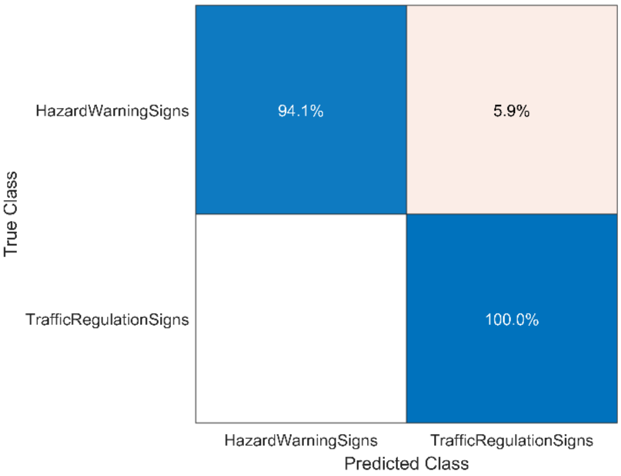

Figure 7. The confusion matrix of the ConvNet model predictions on the test set.

4. Discussion

The average training, validation, and test accuracy of the ConvNet model on the non-augmented data are 68.3% ± 15.4%, 49.2% ± 11.0%, and 51.5% ± 9.6%, respectively. To enhance the representativeness of the original dataset and improve the model's generalization capability, data augmentation was applied. This process increases the diversity of the training data by introducing variations. However, according to the experimental results, the hue jitter transformation contributed significantly to the improvement in model performance. After augmentation, the accuracy results are 97.7% ± 5.2%, 88.8% ± 1.2%, and 96.9% ± 7.2% for training, validation, and test respectively. A considerable difference between the accuracy rates of models trained with and without augmentation was observed, suggesting that the non-augmented model is prone to overfitting As a result, it is apparent that data augmentation alleviates the overfitting problem.

Even after data augmentation, still there exists a noticeable discrepancy between training and validation accuracies. This means that the model may have partially memorized the training data rather than generalizing completely to unseen samples. Hence, data augmentation may reduce overfitting to some extent, but it may not remove it completely in this situation.

There is a key limitation in this study, stemming from the dataset itself. The dataset contains only 61 images of traffic danger-warning signs and 68 images of traffic regulatory signs, which are insufficient for training a deep learning model. This inhibits the power of the model and may limit the generality of the result. Therefore, the performance of the model may not be indicative of its performance on real-world datasets with large volumes and variety of data. In this respect, an open research area for future exploration is the enlargement of the dataset in terms of sample size and variability to validate the results on a wider scale.

Considering the training, validation, and test accuracy of the final ConvNet model, it can be concluded that the model has a significant potential of differentiating between the classes of 'HazardWarningSigns' and 'TrafficRegulationSigns'. Additionally, the model can accurately classify instances with the 'TrafficRegulationSigns' class. Consequently, the experimental results substantiate the validity and effectiveness of the ConvNet model.

5. Conclusion

Traffic sign classification using machine learning has emerged as a widely researched topic within the domain of autonomous driving systems. This study addresses the problem of recognizing the patterns in 129 Turkish traffic signs that were composed of 61 Turkish traffic danger-warning signs and 68 Turkish traffic regulatory signs. To this end, a data augmentation process was first applied, followed by the development of a ConvNet model. The average training, validation, and test accuracies of the ConvNet model were 97.7% ± 5.2%, 88.8% ± 1.2%, and 96.9% ± 7.2%, respectively. Based on the results, the model can effectively detect the correct category of a traffic sign (i.e., 'HazardWarningSigns' and 'TrafficRegulationSigns'). Consequently, the experimental results show the ConvNet model has a remarkable potential in classifying the data. In the next work, this classification task can be extended by using images of road signs on streets and avenues. Leveraging a larger and high variety of real-world data will provide a comprehensive performance assessment of the model, including robustness and generalizability. Furthermore, various ConvNet and deep learning architectures can be systematically applied and evaluated. This approach enables the identification and selection of the model demonstrating the highest performance.

Researchers' Contribution Rate Statement

Authors contributed equally to the work.

Conflict of Interest Statement, if any

The authors certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

References

Abdelkhalek, A., & Mashaly, M. (2023). Addressing the class imbalance problem in network intrusion detection systems using data resampling and deep learning. *The journal of Supercomputing*, *79*(10), 10611-10644.

Al-Ali, E. M., Hajji, Y., Said, Y., Hleili, M., Alanzi, A. M., Laatar, A. H., & Atri, M. (2023). Solar energy production forecasting based on a hybrid CNN-LSTM-transformer model. *Mathematics*, *11*(3), 676.

Altuntaş, Y., Okumuş, F., & Kocamaz, F. (2022). Evrişimsel Sinir Ağları ve Transfer Öğrenme Yaklaşımı Kullanılarak Altın Fiyat Yönünün Tahmini. *Computer Science*, *7*(2), 124-131.

Arora, V., Ng, E. Y.-K., & Singh, A. (2022). Machine learning and its applications. In *Smart Electrical and Mechanical Systems* (pp. 1-37). Elsevier.

Arslan, B., Büyükkaya, T., & Ilgın, H. A. (2016). Real time traffic sign detection and recognition. *Communications Faculty of Sciences University of Ankara Series A2-A3 Physical Sciences and Engineering*, *58*(2), 70-83.

Arul, V. (2021). Deep learning methods for data classification. In *Artificial intelligence in data mining* (pp. 87-108). Elsevier.

Avci, D., Sert, E., Dogantekin, E., Yildirim, O., Tadeusiewicz, R., & Plawiak, P. (2023). A new super resolution Faster R-CNN model based detection and classification of urine sediments. *Biocybernetics and Biomedical Engineering*, *43*(1), 58-68.

Bartneck, C., Lütge, C., Wagner, A., & Welsh, S. (2020). Autonomous vehicles. In *An introduction to ethics in robotics and AI* (pp. 83-92). Springer.

Bucsuházy, K., Matuchová, E., Zůvala, R., Moravcová, P., Kostíková, M., & Mikulec, R. (2020). Human factors contributing to the road traffic accident occurrence. *Transportation research procedia*, *45*, 555-561.

Çetin, E., & Ortataş, F. (2021). Elektrikli ve Otonom Araçlarda Makine Öğrenmesi Kullanarak Trafik Levhaları Tanıma ve Simülasyon Uygulaması. *El-Cezeri*, *8*(3), 1081-1092.

Gezgin, H., & Alkan, R. M. (2024). Traffic sign detection and recognition based on MMS data using YOLOv4-Tiny algorithm. *Neural Computing and Applications*, *36*(33), 20633-20651.

Gündüz, H., Kaplan, S., Günal, S., & Akınlar, C. (2013). Circular traffic sign recognition empowered by circle detection algorithm. 2013 21st Signal Processing and Communications Applications Conference (SIU),

Kilic, I., & Aydin, G. (2020). Traffic sign detection and recognition using tensorflow's object detection API with a new benchmark dataset. 2020 international conference on electrical engineering (ICEE),

Kocakanat, K., & Serif, T. (2021). Turkish traffic sign recognition: comparison of training step numbers and lighting conditions. *Avrupa Bilim ve Teknoloji Dergisi*(28), 1469-1475.

Mahadik, S., Pawar, P. M., & Muthalagu, R. (2023). Efficient intelligent intrusion detection system for heterogeneous internet of things (HetIoT). *Journal of Network and Systems Management*, *31*(1), 2.

Palandız, T., Bayrakçı, H. C., & Özkahraman, M. (2021). Yapay Zekâ Kullanilarak Trafik İşaret Levhalarinin Siniflandirilmasi: Denizli İl Merkezi İçin Örnek Bir Uygulama. *International Journal of 3D Printing Technologies and Digital Industry*, *5*(3), 645-653.

Salman, E. H., Taher, M. A., Hammadi, Y. I., Mahmood, O. A., Muthanna, A., & Koucheryavy, A. (2022). An anomaly intrusion detection for high-density internet of things wireless communication network based deep learning algorithms. *Sensors*, *23*(1), 206.

Salvador, J. (2016). Deep Learning. In *Example-Based super resolution* (pp. 113-127). Elsevier.

Teuwen, J., & Nikita, M. (2019). Convolutional Neural Networks. In *Handbook of medical image computing and computer assisted intervention* (pp. 481-501). Elsevier.

Uluskan, S. (2020). Automatic detection of regulatory traffic signs via circle detection by post edge detection applied to straight line Hough transform. *International Journal of Automotive Science And Technology*, *4*(2), 49-58.

Vilchez, J. L. (2019). Mental representation of traffic signs and their classification: Warning signs. *Transportation research part F: traffic psychology and behaviour*, *64*, 447-462.

Wang, J., Ma, Y., Zhang, L., Gao, R. X., & Wu, D. (2018). Deep learning for smart manufacturing: Methods and applications. *Journal of manufacturing systems*, *48*, 144-156.

Waziry, S., Rasheed, J., Ghabban, F. M., Alsubai, S., Elkiran, H., & Alqahtani, A. (2024). Unveiling interpretability: analyzing transfer learning in deep learning models for traffic sign recognition. *SN Computer Science*, *5*(6), 682.

Yaliç, H. Y., & Can, A. B. (2011). Automatic recognition of traffic signs in Turkey roads. 2011 IEEE 19th Signal Processing and Communications Applications Conference (SIU),

Yıldıran, O. (2019). *Derin Öğrenme Yöntemleri Ile Trafik Işareti Tanıma* Marmara Universitesi (Turkiye)].

Zou, L. (2022). Meta-learning for computer vision. In *Meta-learning: theory, algorithms and applications* (pp. 91-208). Elsevier.