Ministry of Science and Higher Education of the Republic of Kazakhstan
SDU University

Ulan Alsiyeu

Road traffic sign recognition using computer vision

THESIS
Presented in Partial Fulfilment for the

Degree of Master of Technical Science in Computer Science
(degree code: 7M06102)
Department of Computer Science
Faculty of Engineering and Natural Sciences

Supervisor: **Zhasdauren Duisebekov**
Kaskelen, June 2024

SDU University
Faculty of Engineering and Natural Sciences
Department of Computer Science

Dean of Faculty of Engineering and Natural Sciences

Assistant Professor, PhD Akhmedov Ramis

____

« 04 » 06 2024

Topic of the thesis:

Road traffic sign recognition using computer vision

Thesis submitted as part of the requirements for the award of the MSc in
"7M06102 - Computer Science", SDU University

Head of Department	Zhanar Mukash
Academic Supervisor	Zhasdauren Duisebekov
Master student	Ulan Alsiyeu

Kaskelen, 2024

Declaration

I confirm that this is my own work and the use of all material from other sources has been properly and fully acknowledged.

Ulan Alsiyeu

June 2024

Acknowledgements

I would like to thank my supervisors Zhasdauren Duisebekov for his guidance, feedback, and patience.

Dedication

This thesis is dedicated to:

My parents and family, my supervisor Zhasdauren Duisebekov and to the Faculty of Engineering and Natural Sciences.

Abstract

Road traffic accidents are a major public problem in Kazakhstan, with driver inattention and ignorance of traffic signs among the leading causes. Current driver assistance systems integrated in map apps may be inaccurate and irrelevant, especially in rural areas and on highways. The solution proposed by the research includes a computer vision algorithm for accurate and robust detection and recognition of road traffic signs in real time which will be integrated into a mobile application with a notification system. The algorithm will use deep learning neural networks to detect and recognize traffic signs in real time. The algorithm will be trained on a dataset, which will be collected manually and augmented using machine learning techniques. The proposed system has the potential to improve road safety in Kazakhstan by helping drivers to be more aware of traffic signs and to reduce driver inattention.

Аңдатпа

Қазақстанда жол-көлік оқиғалары үлкен қоғамдық мәселе болып табылады, мұнда жүргізушілердің назар аудармауы және жол белгілерін байқамауы басты себептердің бірі. Қазіргі уақытта карта қосымшаларына біріктірілген жүргізушілерді көмек көрсету жүйелері дәл емес және мағлұматтары жаңа емес болуы мүмкін, әсіресе ауылдық аймақтарда және автожолдарда. Зерттеуде ұсынылған шешім жол қозғалысы белгілерін нақты және сенімді анықтау және тану үшін компьютерлік көру алгоритмін қамтиды, ол нақты уақыт режимінде ескерту жүйесімен мобильді қосымшаға біріктірілетін болады. Алгоритм терең оқыту нейрондық желілерін пайдаланып, жол белгілерін нақты уақыт режимінде анықтап, таниды. Алгоритм қолмен жиналып, машиналық оқыту әдістерін қолдана отырып, толықтырылған деректер жинағында дайындалады. Ұсынылған жүйе Қазақстандағы жол қауіпсіздігін жақсартуға, жүргізушілердің жол белгілеріне көбірек назар аударуына және жүргізушілердің назар аудармауын азайтуға әлеуеті бар.

Аннотация

Дорожно-транспортные происшествия являются серьезной общественной проблемой в Казахстане, при этом невнимательность водителей и игнорирование дорожных знаков являются одними из основных причин. Современные системы помощи водителю, интегрированные в приложения с картами, могут быть неточными и нерелевантными, особенно в сельских районах и на автомагистралях. Решение, предложенное в исследовании, включает в себя алгоритм компьютерного зрения для точного и надежного обнаружения и распознавания дорожных знаков в реальном времени, который будет интегрирован в мобильное приложение с системой уведомлений. Алгоритм будет использовать нейронные сети глубокого обучения для обнаружения и распознавания дорожных знаков в реальном времени. Алгоритм будет обучаться на наборе данных, который будет собран вручную и дополнен с использованием методов машинного обучения. Предлагаемая система имеет потенциал для повышения безопасности на дорогах Казахстана, помогая водителям лучше ориентироваться в дорожных знаках и снижая невнимательность водителей.

Abbreviations

TSR – Traffic Sign Recognition
CNN – Convolutional Neural Network
ADAS – Advanced Driver Assistance Systems
GTSRB – German Traffic Sign Recognition Benchmark

Table of Contents

Chapter 1

Background and motivations

1.1 Background

Road safety is a critical concern worldwide, impacting the lives of millions each year. Traffic signs play a pivotal role in guiding and informing drivers, thereby maintaining order and safety on roads. Despite technological advancements in vehicle safety features and infrastructure improvements, road traffic accidents remain a significant issue, particularly in regions like Kazakhstan where the variety of road conditions and insufficiently marked routes pose unique challenges.

The development of road traffic sign recognition (TSR) technology using computer vision is a promising solution to enhance road safety. Computer vision, a field within artificial intelligence (AI), involves enabling computers to interpret and understand the visual world. In the context of TSR, it involves the automatic detection and recognition of traffic signs from real-time video feeds or static images captured by cameras installed in vehicles.

Globally, the application of computer vision in TSR has been integrated into various systems such as Advanced Driver Assistance Systems (ADAS) and mobile navigation apps. These systems help in reducing driver workload and enhancing situational awareness. However, the effectiveness of these technologies can vary significantly based on the region. In Kazakhstan, issues such as diverse traffic signage, rural and unmarked roads, and extreme weather conditions present unique challenges that require tailored solutions.

Moreover, the rapid evolution of mobile technology offers a new platform for deploying TSR applications. With smartphones becoming ubiquitous, leveraging these devices for real-time traffic sign recognition could provide a cost-effective and widely accessible solution to improve road safety. The integration of TSR into mobile apps can serve not only as a supplementary navigation aid but also as a critical real-time alert system, potentially reducing the rate of accidents related to missed or misunderstood road signs.

In summary, while the foundation of TSR using computer vision has been laid and proven effective in various scenarios globally, there is a compelling need to adapt and enhance this technology to address the specific conditions found in Kazakhstan. This adaptation requires a thorough understanding of the local context, including road conditions, traffic sign variability, and technological accessibility, to develop a robust system capable of improving road safety effectively.

1.2 Problem Statement

Road traffic accidents constitute a severe issue in Kazakhstan, claiming lives and causing significant socio-economic losses each year. The primary causes of these accidents include driver inattention and a lack of adherence to road traffic signs, which are crucial for the safe and efficient movement of vehicles. Current solutions, such as traditional GPS-based navigation systems, often fall short, particularly in areas where digital map data is outdated or inaccurate, such as rural regions and complex urban environments.

The limitations of existing driver assistance technologies in Kazakhstan stem from several factors. Firstly, the reliance on static data without real-time updates makes traditional systems less responsive to dynamic changes in road conditions or temporary sign placements. Secondly, the geographic and infrastructural diversity of Kazakhstan poses a significant challenge. For instance, signs that are vital in mountainous areas might differ considerably from those in urban settings, requiring systems that can adapt quickly to varied environments. Lastly, extreme weather conditions, such as heavy snow, fog, and rain, frequently obscure road signs, making them difficult for both drivers and traditional computer vision systems to detect reliably.

The inadequacy of current technologies in addressing these challenges highlights the need for a novel approach that not only recognizes traffic signs accurately but also does so in real time, adapting to the specific driving conditions prevalent in Kazakhstan. The development of a mobile application leveraging advanced computer vision techniques could bridge this gap. Such an application would not only enhance the accuracy of traffic sign recognition under diverse conditions but also provide timely alerts to drivers, thereby mitigating the risks associated with driver inattention and misinformation.

1.3 Aim and Objectives

The overarching aim of this research is to develop a robust, real-time road traffic sign recognition system tailored for the unique driving conditions of Kazakhstan. This system will be designed to integrate seamlessly with mobile devices, providing a practical, user-friendly solution that enhances road safety through accurate sign recognition and timely notifications.

To achieve this aim, the research will focus on the following specific objectives:

1. Dataset Collection: Gather a comprehensive dataset of road traffic signs specific to Kazakhstan, encompassing a wide range of signs under various environmental conditions.
2. Algorithm Development: Develop and refine a computer vision algorithm capable of recognizing and interpreting Kazakhstan's diverse traffic signs accurately and swiftly.
3. Mobile Application Development: Design and implement a user-friendly mobile application that integrates the developed algorithm, ensuring it is accessible and functional for real-time use.
4. System Testing and Evaluation: Rigorously test the application under real-

world conditions to evaluate its effectiveness and reliability across different regions and weather conditions in Kazakhstan.

1.4 Significance of Research

The significance of this research is manifold, addressing both theoretical and practical aspects of traffic sign recognition using computer vision. Theoretically, this research contributes to the growing body of knowledge in AI by developing novel computational techniques tailored to the unique challenges presented by the diverse and often harsh driving environments of Kazakhstan. The research outcomes are expected to provide valuable insights into the adaptability and scalability of AI technologies in real-world applications, particularly in settings that differ markedly from typical Western environments where most existing research is concentrated.

Practically, the research has the potential to significantly impact road safety in Kazakhstan. By providing a real-time, accurate, and reliable traffic sign recognition system, the project aims to reduce traffic accidents caused by driver error related to unrecognized or misinterpreted traffic signs. Furthermore, the accessibility of this technology through mobile devices makes it a potentially life-saving tool for a broad user base, ranging from professional drivers to everyday commuters, thereby enhancing the overall safety and efficiency of road travel in the country.

The project's findings and the developed technologies will also be shared with the global research community through publications in peer-reviewed journals, contributing to the advancement of global knowledge in traffic safety and mobile computing applications in challenging environments. Additionally, the open-source availability of the data and technologies developed during this project will enable further innovation and adaptation by researchers and developers worldwide, broadening the impact of the work beyond the borders of Kazakhstan.

Chapter 2

Literature review

2.1 Evolution of Traffic Sign Recognition Technologies

The development of traffic sign recognition (TSR) technologies has been a crucial area of research within the broader context of automotive safety and autonomous driving systems. This evolution can be traced from early rudimentary systems to the sophisticated deep learning approaches that characterize the current state of the art. This subsection explores the chronological development of TSR technologies, highlighting key innovations and shifts in methodologies that have significantly impacted the field.

2.1.1 Early Developments

The journey of TSR technologies began in the late 20th century with the advent of digital imaging and computer vision techniques. Initially, these systems were rudimentary, relying heavily on simple pattern recognition algorithms that could identify basic shapes and colors associated with traffic signs. These early systems utilized techniques such as color segmentation, edge detection, and region-based methods to isolate and identify signs from background noise.

2.1.1.1 Key Techniques:

- Color Segmentation: Early TSR systems used color as a primary feature for detecting traffic signs, especially those with distinct colors like red, blue, and yellow.
- Edge Detection: Techniques like the Sobel operator were employed to detect edges and outlines of signs, aiding in their recognition.

2.1.2 Transition to Machine Learning

As computational power increased and machine learning algorithms became more refined, the second generation of TSR technologies began to take shape. Machine learning allowed for more sophisticated pattern recognition, where systems were not only recognizing shapes and colors but were also capable of understanding and classifying content based on historical data.

2.1.2.1 Machine Learning Models:

- Support Vector Machines (SVM): By the early 2000s, SVMs were popular for their ability to handle non-linear boundaries in classification tasks, making them ideal for TSR where signs could appear in various orientations and scales.
- Neural Networks: Simple neural network architectures were explored to enhance recognition accuracy, although these were initially limited by the computational resources available at the time.

2.1.3 Integration of Deep Learning

The real revolution in TSR came with the incorporation of deep learning techniques, particularly Convolutional Neural Networks (CNNs). CNNs represent a significant departure from traditional machine learning because they automatically detect important features without any human intervention, learning optimal features directly from the raw images during the training process.

2.1.3.1 Breakthroughs with CNNs:

- LeNet-5 (1998): Although originally designed for digit recognition, LeNet-5 laid the foundational principles for modern CNNs used in image recognition, including TSR.
- AlexNet (2012): This model demonstrated the capabilities of deep neural networks in handling complex image recognition tasks, setting the stage for their adoption in TSR.

2.1.4 Advances with Real-Time Systems

The need for real-time recognition in autonomous and semi-autonomous vehicles has pushed the development of faster and more efficient TSR systems. This necessity has led to the exploration and adoption of models that balance accuracy with computational efficiency.

2.1.4.1 Real-Time Technologies:

- You Only Look Once (YOLO): YOLO frameworks have been particularly influential in TSR for their ability to detect objects in real-time. YOLOv5, for example, offers significant improvements in speed and accuracy, facilitating real-time TSR even on devices with limited computational power like mobile devices and embedded systems.
- MobileNets: These networks are designed specifically for mobile devices, providing a good balance between latency and accuracy, crucial for real-time applications in mobile or edge computing scenarios.

2.1.5 Current Trends and Future Directions

Today, the field of TSR is moving towards even more integrated systems that combine TSR with other sensory data (like LIDAR and radar) to enhance the robustness and reliability of autonomous driving systems. The integration of artificial intelligence (AI) with cloud computing and the Internet of Things (IoT) represents the next frontier in TSR, promising systems that are not only accurate and efficient but also capable of contextual understanding and decision-making.

2.1.5.1 Emerging Innovations:

- Deep Reinforcement Learning: This approach is being explored for adaptive learning environments where TSR systems can dynamically improve based on the driving context.
- Federated Learning: For privacy-preserving collaborative learning, federated learning allows TSR models to learn from decentralized data sources without compromising user privacy.

In conclusion, the evolution of traffic sign recognition technologies highlights a trajectory from basic image processing techniques to advanced deep learning models, driven by the needs for greater accuracy, speed, and adaptability in a range of automotive applications. As the field continues to evolve, future developments are likely to focus on enhancing real-time processing capabilities, reducing system latencies, and integrating multimodal data to support fully autonomous driving systems.

2.2 Review of related literature

2.2.1 A New Attention-Based Deep Convolutional Neural Network for Traffic Sign Recognition

2.2.1.1 Introduction

In the realm of smart vehicles, advancements in Artificial Intelligence (AI) have significantly propelled the capabilities of Advanced Driver Assistance Systems (ADAS) and Automated Driving Systems (ADS). Among the technologies, Traffic Sign Recognition (TSR) systems are pivotal for improving vehicular safety and navigation. The research by Triki, Karray, and Ksantini (2023)[1] introduces a novel approach to TSR using an attention-based deep convolutional neural network (CNN) designed for real-time application on smart vehicles. This system aims to address the shortcomings of existing models by incorporating a broader range of traffic sign recognitions, thereby enhancing accuracy and processing efficiency.

2.2.1.2 Existing Traffic Sign Recognition Technologies

The paper begins with a comprehensive overview of existing TSR technologies, which primarily leverage machine learning techniques to detect and classify road signs. Traditional methods rely on features like color, shape, and texture. However, these methods falter under variable environmental conditions such as lighting changes or obstructions. To overcome these limitations, recent approaches have employed deep learning techniques, significantly improving the robustness and accuracy of TSR systems. Despite these advancements, current systems predominantly focus on a limited subset of traffic signs, often restricted to regulatory signs such as speed limits.

2.2.1.3 Proposed Methodology

The authors propose a new methodology that integrates the Haar cascade technique with a deep learning CNN model to expand the recognition capabilities to include warning, regulatory, obligatory, and priority signs. This method uses an attention mechanism within the CNN to enhance the focus on relevant features of the traffic signs, improving the classification accuracy to 99.91

2.2.1.4 Experimental Results and Evaluation

The system was trained on the German Traffic Sign Recognition Benchmark (GTSRB) dataset, achieving high accuracy and precision across various sign categories. The attention-based model not only demonstrated superior performance in terms of accuracy but also in the speed of recognition, achieving a detection time of 150 ms per sign. These results indicate a significant step forward in the development of more effective TSR systems for smart vehicles.

2.2.1.5 Implications and Future Work

The study's implications extend beyond smart vehicles, potentially impacting the development of autonomous driving technologies where accurate and rapid recognition of road signs is critical. The authors suggest that future research could explore the integration of this TSR system with other vehicular systems for enhanced navigational aids in ADAS and ADS.

2.2.1.6 Conclusion

The paper by Triki et al. presents a significant advancement in the field of traffic sign recognition technology. By integrating an attention-based deep CNN with traditional detection techniques, the proposed system not only increases the range of detectable signs but also ensures high accuracy and speed, crucial for real-time applications in smart vehicles. This research fills a critical gap in existing TSR technologies and sets a promising direction for future developments in autonomous and assisted driving technologies.

2.2.2 CNN Design for Real-Time Traffic Sign Recognition

2.2.2.1 Introduction

The advancement in mobile processors has enabled the integration of sophisticated computer vision systems into vehicles, enhancing safety and paving the way towards autonomous driving. The 2017 study [2] by Alexander Shustanova and Pavel Yakimova explores the application of Convolutional Neural Networks

(CNNs) in real-time traffic sign recognition (TSR). Despite existing advancements, challenges remain, such as low detection accuracy and high computational demands, particularly when recognizing traffic signs from diverse international contexts.

2.2.2.2 Methodology

The study presents a comprehensive approach to TSR that involves image preprocessing, sign localization, and classification using CNNs. The researchers utilized the TensorFlow framework and CUDA for parallel processing, ensuring the system's operation in real-time on a mobile GPU. The process begins with the localization of traffic signs using modified Generalized Hough Transform (GHT) algorithms, followed by CNN-based classification.

2.2.2.3 CNN Architectures and Execution

Several CNN architectures were tested to determine the most efficient structure for traffic sign recognition. The initial models were complex with multiple layers, which, while accurate, were computationally expensive and not optimal for real-time processing. Subsequent revisions simplified the architecture without significantly compromising the recognition accuracy. The best-performing model featured a combination of convolutional layers with decreasing kernel sizes and fully connected layers, optimized for both speed and accuracy.

2.2.2.4 Experimental Setup and Results

Experiments were conducted using standard traffic sign datasets like the German Traffic Sign Detection Benchmark (GTSDB) and Recognition Benchmark (GTSRB), which contain over 50,000 images under varied conditions. The final model achieved a high accuracy rate of 99.94

2.2.2.5 Implications and Future Directions

The research highlights the potential of CNNs in enhancing real-time traffic sign recognition systems, crucial for the development of autonomous vehicles. By optimizing CNN architectures for mobile GPUs, the system achieves high accuracy with rapid processing times, suitable for real-time applications. Future work could expand the range of recognizable signs and explore the system's robustness under different environmental conditions.

2.2.2.6 Conclusion

Shustanova and Yakimova's study represents a significant contribution to traffic sign recognition technology, particularly for applications requiring real-time processing. The combination of advanced CNN models with efficient computational strategies holds promise for the future of autonomous vehicle systems, potentially leading to safer and more reliable driving assistance technologies.

2.2.3 Real-Time Application of Traffic Sign Recognition Algorithm with Deep Learning

2.2.3.1 Introduction

The proliferation of autonomous vehicles heralds a significant shift in transportation systems, promising enhanced safety and efficiency. The 2022 study by Aysal, Yıldırım, and Cengiz [3] explores deep learning applications in traffic sign recognition (TSR), critical for the operational success of these vehicles. This study is distinguished by its use of the YOLOv5 deep learning model, reflecting a sophisticated approach to handling real-time traffic sign recognition.

2.2.3.2 Background and Motivation

With the rising number of vehicles and the consequent increase in traffic-related issues, such as congestion and accidents, there is a pressing need for more intelligent transportation solutions. Autonomous vehicles, equipped with advanced AI systems, offer a promising solution by potentially reducing human error and improving traffic management. Prior studies have leveraged various deep learning models for TSR, but real-time processing remains a challenge due to the computational demands and the need for high accuracy in diverse driving conditions.

2.2.3.3 Deep Learning for Traffic Sign Recognition

The study employs the YOLOv5 model, which is favored for its efficiency in processing and high accuracy. The research involves preparing a unique dataset of traffic signs, which is then used to train the model. The focus is on evaluating the model's performance in real-time scenarios, a critical requirement for autonomous driving applications. The authors highlight the use of different versions of YOLOv5 (YOLOv5s, YOLOv5m, YOLOv5x), comparing their effectiveness in terms of detection speed and accuracy.

2.2.3.4 Methodology

Aysal et al. collected a dataset of 1249 traffic sign images from Afyonkarahisar, Turkey, which were annotated and used for training and testing the models. They employed a rigorous training regimen, optimizing the models for real-time application by adjusting hyperparameters and utilizing TensorRT technology for enhanced performance on specific hardware like NVIDIA Jetson Nano.

2.2.3.5 Results and Evaluation

The study reports high success rates, with the YOLOv5 models achieving accuracy between 98-99

2.2.3.6 Conclusion and Future Work

The research by Aysal et al. is pivotal in demonstrating the viability of using advanced deep learning models like YOLOv5 for traffic sign recognition in real-time. The success of these models in accurately recognizing traffic signs at high speeds supports their integration into autonomous driving systems, which could lead to significant improvements in traffic management and vehicle safety.

2.2.3.7 Implications for Future Research

The study suggests further enhancements in model training and real-time testing on various hardware platforms. Expanding the dataset and including more diverse traffic scenarios could also improve model robustness, ensuring effective performance across different environments.

2.2.4 Real-Time Navigation Roads: Lightweight and Efficient Convolutional Neural Network (LE-CNN) for Arabic Traffic Sign Recognition

2.2.4.1 Introduction

The research by Khalifa, Alayed, Elbadawy, and Sadek [4] introduces the Lightweight and Efficient Convolutional Neural Network (LE-CNN), specifically designed to enhance Arabic traffic sign recognition within Intelligent Transportation Systems (ITS). As smart cities evolve, the integration of advanced technologies in transportation systems is crucial for enhancing traffic management and safety. The LE-CNN model addresses challenges such as real-time processing and variability in sign appearances, aiming to improve the reliability and efficiency of traffic sign detection.

2.2.4.2 Background and Challenges

Traffic sign recognition is a vital component of autonomous vehicle navigation and ITS, helping to reduce congestion and increase road safety. Existing systems, while capable, often struggle with real-time processing speeds and accuracy under varying environmental conditions. The variability in the appearance of signs, occlusions, and poor image quality are significant hurdles that previous models have struggled to overcome efficiently.

2.2.4.3 LE-CNN Architecture

The LE-CNN architecture leverages depth-wise separable convolutions and channel pruning to optimize performance. These technical enhancements allow the LE-CNN to operate with reduced computational demand while maintaining high accuracy and speed. This model is particularly tailored to handle the intricacies of Arabic traffic signs, which present unique challenges due to their script and design.

2.2.4.4 Methodology and Dataset

The researchers employed a comprehensive dataset of Arabic traffic signs, which includes a variety of common signs encountered in urban and rural settings. This dataset was used to train and test the LE-CNN, ensuring that the model could effectively recognize and interpret these signs under different conditions. The training process emphasized minimizing overfitting while maximizing the model's ability to generalize across diverse scenarios.

2.2.4.5 Results and Performance

The LE-CNN model demonstrated impressive performance metrics, achieving an accuracy of 96.5

2.2.4.6 Conclusion and Future Applications

The study concludes that the LE-CNN model is a substantial advancement in the field of traffic sign recognition, offering high accuracy and efficiency. The success of this model in recognizing Arabic traffic signs opens the door for its application in various ITS across the globe, particularly in regions with similar linguistic and cultural characteristics.

2.2.4.7 Implications for Smart Cities

Implementing this model within smart city infrastructures could drastically improve the operational efficiency of autonomous vehicles and ITS, contributing to safer, more reliable urban transportation networks. Future research may focus on expanding the dataset and refining the model to include more diverse traffic sign variants and to further enhance its real-time processing capabilities.

2.2.5 Real-time Traffic Sign Detection and Recognition Using Raspberry Pi

2.2.5.1 Introduction

In the field of intelligent transportation systems, traffic sign recognition (TSR) plays a crucial role in enhancing road safety and automating driving processes. The study [5] by Md Isa, Ja Yeong, and Mohd Shaari Azyze presents a TSR system developed using TensorFlow on a Raspberry Pi 3, focusing on achieving high accuracy and minimal delay in real-time environments. This system is designed to alert drivers to traffic signs, potentially reducing road accidents significantly.

2.2.5.2 Background and Problem Statement

Road accidents in Malaysia are a growing concern, with traffic signs often missed by drivers due to various obstructions or distractions. The authors highlight the critical need for an efficient TSR system that can operate in real-time, with the Raspberry Pi 3 chosen for its cost-effectiveness and sufficient processing capabilities for such applications.

2.2.5.3 Methodology

The developed system utilizes the TensorFlow framework for object recognition, leveraging machine learning to enhance its accuracy. The system was tested on a real-time testbed using a Raspberry Pi camera NoIR to capture video feeds, which were then processed to detect and recognize traffic signs under various environmental conditions. The study explored twenty different traffic signs to evaluate the system's performance in diverse scenarios.

2.2.5.4 Results and Performance

The TSR system exhibited over 90

2.2.5.5 Technological Implications and Advancements

The use of Raspberry Pi 3 for real-time traffic sign recognition represents a significant technological advancement due to its low cost and accessibility. This approach makes advanced driving assistance systems more feasible for widespread adoption, potentially leading to safer driving environments.

2.2.5.6 Conclusion and Future Work

The research concludes that the TSR system developed is capable of effectively detecting and recognizing traffic signs with high accuracy and minimal delay, making it a viable solution for incorporation into advanced driver-assistance systems (ADAS). Future work could expand on integrating this system with other components of ADAS and exploring its effectiveness in different geographical and environmental conditions.

2.2.5.7 Significance for Intelligent Transportation Systems

This study contributes to the body of knowledge in intelligent transportation by demonstrating the feasibility of using affordable and accessible technology to enhance road safety. The successful implementation of such systems could lead to significant reductions in road accidents, especially in developing regions.

2.2.6 Research on Traffic Sign Recognition Algorithm Based on YOLOv5

2.2.6.1 Introduction

The paper [6] by Zhichen Li and Hua Huo from Henan University of Science and Technology introduces an advanced traffic sign recognition algorithm based on the YOLOv5 model. This research is significant due to the increasing complexity of traffic signs and their importance in autonomous driving technologies. The study proposes enhancements to the YOLOv5 architecture to address the challenge of recognizing small and densely packed traffic signs in real-world scenarios.

2.2.6.2 The YOLOv5 Framework

YOLOv5 is a popular choice in object detection due to its speed and accuracy, which are crucial for real-time applications like traffic sign recognition. The framework divides into distinct parts: Backbone, Neck, and Prediction, each contributing to the efficiency and accuracy of the model. This paper focuses on improvements to each component to enhance detection performance, especially for small target objects that are typical in traffic sign recognition tasks.

2.2.6.3 Innovations Introduced

1. CBAM Attention Mechanism: The Convolutional Block Attention Module (CBAM) is integrated to improve the model's focus on relevant features within traffic signs. CBAM applies attention across both channel and spatial dimensions, enhancing the feature extraction capability without significant computational overhead.
2. Enhanced Feature Pyramid: The study modifies the feature pyramid structure to better capture small target details. This adaptation helps in accurately recognizing traffic signs that are small or appear at a distance, a common issue in real-world driving scenarios.
3. Improved Upsampling Algorithm: To preserve high-resolution details during feature upsampling, the research shifts from nearest neighbor to bilinear interpolation, which better handles the finer details necessary for accurate traffic sign recognition.
4. DIoU Loss Function: The DIoU (Distance Intersection over Union) loss function is introduced to improve the localization accuracy by considering the overlap and the central point distance between predicted and ground truth bounding boxes. This adjustment helps in reducing the model's prediction errors, particularly in overlapping traffic sign scenarios.

2.2.6.4 Results and Performance

The enhanced YOLOv5 model demonstrates superior performance over the standard configuration, particularly in recognizing small and densely packed traffic signs. The model achieves this with minimal impact on computational efficiency, making it viable for real-time applications. The implementation on the TT100K dataset, a comprehensive traffic sign dataset, shows significant improvements in both precision and recall metrics.

2.2.6.5 Conclusion and Future Directions

This paper successfully addresses several challenges associated with traffic sign recognition using deep learning. The enhancements to the YOLOv5 model not only improve its accuracy but also its applicability in dynamic and complex real-world conditions. Future research could explore the integration of these improvements into broader intelligent transportation systems and further optimization to accommodate even faster real-time processing requirements.

2.2.7 Conclusion

This literature review encompasses six pivotal studies on traffic sign recognition (TSR) using advanced convolutional neural networks (CNNs) and deep learning techniques, aimed at enhancing the accuracy and speed essential for autonomous driving systems. The studies introduce various innovations, including attention mechanisms, optimized CNN architectures, and specialized models for small and densely packed signs, demonstrating significant improvements in real-time processing on platforms ranging from Raspberry Pi to advanced GPUs. Key advancements include the integration of the YOLOv5 model, attention-based feature enhancements, and the application of sophisticated loss functions like DIoU to refine localization and detection accuracy. These improvements not only elevate the performance of TSR systems but also ensure their applicability in diverse environmental conditions, ultimately contributing to safer road conditions.

Chapter 3

Methodology

3.1 Data Collection

3.1.1 Introduction to Data Collection Challenges

The development of a robust and effective Traffic Sign Recognition (TSR) system depends heavily on the quality and comprehensiveness of the dataset used. In the context of Kazakhstan, the challenges are particularly significant due to the unique variety of traffic signs, the diverse environmental conditions, and the geographic variability across the country. Addressing these challenges necessitates a strategic approach to data collection that ensures a dataset not only rich in diversity but also tailored to the specific requirements of the region.

3.1.2 Collection Strategies

3.1.2.1 Public Data Sources

The dataset for this study was meticulously collected from official documents to ensure the accuracy and relevance of traffic signs specific to Kazakhstan. The process involved extracting visual representations of traffic signs directly from the comprehensive legal framework established within the Republic of Kazakhstan. This legal documentation served as the foundational basis for the image dataset, ensuring that the collected signs were legally accurate and representative of those found on Kazakhstani roads. By utilizing these official documents, the dataset accurately reflects the variety and specifications of traffic signs mandated by the country's traffic regulations, providing a robust foundation for developing and evaluating traffic sign recognition models tailored to the local context. This approach not only enhances the dataset's authenticity but also ensures its applicability in real-world scenarios, contributing to improved model performance and road safety in Kazakhstan.

3.1.2.2 Real-World Data Capturing

In addition to extracting traffic sign images from official documents, the dataset was further enriched by capturing real-world scenarios using a car dashcam. This process involved driving through various roads in Kazakhstan to record diverse and realistic traffic environments. The dashcam footage provided a comprehensive set of images depicting traffic signs in their natural contexts, including different lighting conditions, weather situations, and occlusions by other objects like vehicles and pedestrians. This approach ensured that the dataset included a variety of real-world challenges that advanced driver-assistance systems (ADAS) might encounter, thereby enhancing the robustness and generalizability of the traffic sign recognition models developed using this dataset.

3.1.3 Data Augmentation Techniques

Given the limitations in the volume and variety of the naturally occurring traffic sign instances, particularly those that are rare or have significant importance but low visibility, data augmentation techniques were extensively used. As outlined previously, synthetic image generation and geometric transformations played a critical role in this process.

3.1.3.1 Synthetic Image Generation

Synthetic image generation involves creating artificial images of traffic signs through computer graphics techniques. This method allows for the controlled manipulation of sign features such as color, size, and rotation, which are essential for teaching the TSR system to recognize signs under varied conditions. These synthetic images were generated using advanced software that simulates different lighting conditions and physical damages such as fading, which are common in real-world scenarios.

3.1.3.2 Geometric and Environmental Transformations

Geometric transformations, including scaling, rotating, and shearing, were applied to both real and synthetic images to mimic the different angles and distances from which a driver might view a sign. Environmental transformations were also simulated, such as varying weather conditions (e.g., rain, snow, fog) and different times of the day, to ensure the system's robustness under all potential driving conditions.

3.1.4 Quality Assurance and Labeling

Each image collected or generated underwent a rigorous quality assurance process to ensure it met the resolution and clarity standards necessary for effective training of computer vision models. Additionally, a meticulous labeling process was undertaken. Each sign in the images was labeled with its class, condition, and background of where the image was captured or is likely to be found in Kazakhstan. This metadata is crucial for the subsequent phases of algorithm training and evaluation.

3.1.5 Data Security and Ethical Considerations

Given the sensitivity and potential personal data involved in real-world image capturing, all data collection processes were conducted following strict ethical guidelines and data protection regulations. Personal identifiers were removed from all images, and data was stored on secure servers with restricted access, ensuring compliance with national and international data protection laws.

3.1.6 Conclusion

The data collection phase for the TSR system in Kazakhstan was a multifaceted effort that required meticulous planning, diverse methodologies, and collaborative initiatives. By combining synthetic image generation with extensive real-world data capturing and innovative data augmentation techniques, the project has established a comprehensive and robust dataset. This dataset not only serves the immediate needs of developing a high-performing TSR system but also provides a valuable resource for future research and development in the field of computer vision and autonomous driving in Kazakhstan.

3.2 Algorithm Development

3.2.1 Introduction

The development of a robust algorithm for Traffic Sign Recognition (TSR) is crucial to ensure the accurate and efficient identification of road signs in real-time, which is essential for enhancing driver safety and navigation aids. This section details the algorithm development process for the TSR system, specifically using the MobileNetV2 model, a lightweight deep learning model known for its efficiency and effectiveness in mobile applications.

3.2.2 MobileNetV2: A Primer

MobileNetV2 is an advanced deep learning model designed primarily for mobile and edge devices, considering the constraints of these environments, such as limited computational resources and power. The architecture of MobileNetV2 builds on the initial MobileNet model, incorporating a streamlined architecture that utilizes depthwise separable convolutions as its core building block. These convolutions split the convolutional process into two layers: a depthwise convolution and a pointwise convolution, significantly reducing the number of parameters and computational complexity. The MobileNetV2 architecture introduces two significant enhancements over its predecessor: Linear Bottlenecks and Inverted Residuals. Linear bottlenecks compress the input information into a smaller channel dimension, preserving important information using linear transformations. Inverted residuals, on the other hand, apply a residual connection around the lightweight depthwise convolutions, allowing for the building of deeper networks that maintain compactness and efficiency.

3.2.3 Adapting MobileNetV2 for Traffic Sign Recognition

3.2.3.1 Pre-processing Steps

Before training the MobileNetV2 model, comprehensive pre-processing of the dataset images was essential. This process included:

- Normalization: Each image in the dataset was normalized to ensure uniformity in lighting and color. Normalization aids the model in processing images more efficiently by converting pixel value ranges to a standard scale.
- Resizing: MobileNetV2 requires a specific input size, which, for the purposes of this project, was set to 224x224 pixels. All images were resized to this dimension to maintain consistency and ensure that the network could process them effectively.
- Augmentation: To improve the model's robustness and ability to generalize from the training data, image augmentation techniques previously described (geometric transformations, weather simulations, etc.) were applied. This step is crucial for preparing the model to perform well in varied real-world conditions.

3.2.3.2 Model Configuration

The configuration of the MobileNetV2 model was tailored to the specific needs of traffic sign recognition. The base MobileNetV2 model was used as a feature extractor, where the top classification layers were replaced with new layers specifically designed for the TSR task. These modifications included:

- Custom Output Layer: The output layer was adapted to classify the specific number of traffic sign classes identified in the Kazakhstan dataset. This involved adjusting the final fully connected layer to output probabilities across the defined classes.
- Activation Functions: The ReLU6 activation function, characteristic of MobileNetV2, was retained to maintain non-linearity while preventing large activation values, which are computationally expensive on mobile devices.

3.2.4 Training the Model

3.2.4.1 Splitting the Data

The dataset was split into training, validation, and testing sets with a ratio of 70:15:15. This split ensured that the model could be trained on a large portion of the data while still being validated and tested adequately to assess its performance accurately.

3.2.4.2 Training Process

The training of the MobileNetV2 model was conducted using the following steps:

- Optimizer and Loss Function: The Adam optimizer was chosen for its effectiveness in handling sparse gradients on noisy problems, such as traffic sign recognition. A categorical cross-entropy loss function was used since this is a multi-class classification problem.

- Learning Rate and Epochs: A learning rate scheduler was employed to adjust the rate based on the validation loss, reducing the learning rate when the validation loss plateaued. The model was trained for 50 epochs, allowing sufficient time for the weights to converge to an optimal state.
- Early Stopping: To prevent overfitting, an early stopping mechanism was implemented. Training would cease if the validation loss did not improve for ten consecutive epochs.

3.2.5 Model Evaluation and Optimization

After the initial training phase, the model underwent a rigorous evaluation process using the separate test set. Performance metrics such as accuracy, precision, recall, and F1-score were calculated to assess the model's effectiveness in recognizing different traffic signs under varied conditions. Confusion matrices were also generated to visualize the model's performance across different classes, identifying any specific signs for which the model performed poorly.

3.2.5.1 Fine-Tuning and Hyperparameter Optimization

Based on the initial evaluation, fine-tuning adjustments were made to the model's architecture and training parameters. Hyperparameter optimization techniques, such as grid search and random search, were employed to find the optimal settings that maximize the model's performance. These adjustments might include changes to the learning rate, batch size, or even modifications to the model architecture itself.

3.2.6 Conclusion

The development of the MobileNetV2-based TSR algorithm represents a significant advancement in applying deep learning to the field of road safety. By leveraging a state-of-the-art architecture tailored for mobile deployment, the project addresses both the high accuracy requirements of traffic sign recognition and the practical constraints of mobile applications. This balance of efficiency and effectiveness ensures that the TSR system can be deployed widely across Kazakhstan, significantly enhancing road safety and driver assistance capabilities.

3.3 Mobile Application Development

3.3.1 Introduction

The development of a mobile application for Traffic Sign Recognition (TSR) serves as a crucial interface between the computational models developed and the end-users, typically drivers, in real-time scenarios. This application, leveraging the TSR model built on MobileNetV2 architecture, aims to provide instant traffic sign recognition and alerts to drivers, enhancing road safety significantly. React Native, a popular framework for building native applications using JavaScript, was chosen for this task due to its efficiency, cross-platform capabilities, and large community support.

3.3.2 React Native Framework: Overview

React Native allows for the development of genuine native applications using a unified JavaScript codebase. It provides the benefit of code reusability across iOS and Android platforms while maintaining a high-performance user experience that is almost indistinguishable from an app developed in a native language like Java or Swift. For this project, the focus is on Android due to its widespread use in Kazakhstan.

3.3.3 Application Architecture

3.3.3.1 Component Structure

The application is structured around several core components:

- Camera Interface: Integrates with the device's camera to continuously capture video footage that the TSR system can process.
- Sign Recognition Module: Utilizes the trained MobileNetV2 model to analyze the captured images in real-time and recognize traffic signs.
- Notification System: Alerts the user through visual and auditory signals whenever a traffic sign is recognized, particularly those requiring immediate driver action (e.g., Stop, Yield).

3.3.3.2 Data Handling

Given the real-time nature of the application, efficient data handling and state management are crucial. React Native's context API is used to manage the application state across different components, ensuring that the state is maintained consistently across the app without unnecessary re-renders.

3.3.4 Development Tools and Libraries

3.3.4.1 React Native Libraries

Several key libraries are integral to the development:

- React Navigation: Manages the navigation between different screens in the app.
- TensorFlow.js: A library that allows running TensorFlow models directly in the browser or on a mobile device. TensorFlow.js is used to execute the MobileNetV2 model within the app.
- React Native Camera: A comprehensive camera module for React Native that supports photographs, videos, and streaming.
- React Native Push Notification: Manages the delivery of push notifications to the user.

3.3.5 Application Development Phases

3.3.5.1 Phase 1: Setup and Basic Functionality

The initial phase involves setting up the React Native environment, configuring the basic navigation structure, and implementing the camera interface. This setup lays the groundwork for integrating the more complex functionalities.

3.3.5.2 Phase 2: Integration of MobileNetV2 Model

The MobileNetV2 model, converted to TensorFlow.js format, is integrated into the application. This phase requires thorough testing to ensure the model's performance is optimized for mobile devices, focusing on minimizing latency and maintaining real-time performance.

3.3.5.3 Phase 3: Notification System Implementation

The notification system is designed to alert the driver about recognized traffic signs through audio-visual cues. The system prioritizes critical signs that require immediate action, ensuring that alerts are noticeable but not distracting under driving conditions. This system is tested extensively to refine the alert timings and modalities based on user feedback and usability testing.

3.3.5.4 Phase 4: User Interface and Experience

The user interface (UI) is designed to be intuitive and minimalistic, ensuring that it is accessible to drivers without causing distractions. User experience (UX) tests are conducted with a focus group representing a broad spectrum of drivers to gather insights and improve the interface accordingly.

3.3.5.5 Phase 5: Field Testing and Iteration

Field tests are conducted to observe the application's performance in real-world driving scenarios across Kazakhstan. These tests are crucial for gathering data on the app's effectiveness in different geographic and environmental conditions. Feedback from these tests is used to make iterative improvements to the application.

3.3.6 Deployment and Updates

Once the application meets all performance and safety standards, it is deployed on the Google Play Store. Post-deployment, the app is continuously updated based on user feedback and advancements in TSR technology. Regular updates ensure that the application adapts to new traffic signs and regulations as they are introduced.

3.3.7 Conclusion

The development of the TSR mobile application using React Native represents a significant step forward in applying cutting-edge AI technology to enhance road safety. By combining the computational power of the MobileNetV2 model with the widespread accessibility of mobile technology, this application promises to reduce traffic-related incidents significantly by keeping drivers informed and alert to their surroundings.

3.4 System Evaluation

3.4.1 Introduction

Evaluating the performance and reliability of the Traffic Sign Recognition (TSR) system is crucial to ensuring its effectiveness and safety in real-world applications. This section outlines the methodologies and metrics used to assess the comprehensive functionality of the developed TSR mobile application, which includes the MobileNetV2-based recognition algorithm and the React Native mobile platform.

3.4.2 Evaluation Objectives

The primary objectives of the system evaluation are to:

- Assess the accuracy and response time of the traffic sign recognition algorithm under various environmental and lighting conditions.
- Evaluate the usability and user satisfaction of the mobile application in real-world driving scenarios.
- Identify any potential issues or areas for improvement in both the algorithm and the application.

3.4.3 Performance Metrics

The performance of the MobileNetV2 model within the application is evaluated through several metrics:

- Accuracy: The percentage of total correct predictions made by the model out of all predictions.
- Precision and Recall: Specifically important for applications where missing a traffic sign could lead to significant safety risks.
- F1 Score: The harmonic mean of precision and recall, providing a single metric to evaluate the balance between the two.

The model is tested across a diverse dataset that includes variations in sign types, sizes, occlusions, and environmental conditions such as weather effects and lighting variations. This testing ensures the model's robustness and its ability to function accurately in diverse real-world conditions.

3.4.4 Response Time

The response time, or latency, of the sign recognition process from image capture to notification delivery, is critical. This metric is assessed to ensure that the application provides real-time information to the driver, which is crucial for safety. The goal is to maintain a response time that allows for timely human reaction to traffic sign notifications.

3.4.5 Usability Evaluation

3.4.5.1 User Interface and Interaction

The mobile application's user interface (UI) and overall user experience (UX) are evaluated through user testing sessions. These sessions involve typical potential users engaging with the application in controlled and on-road scenarios to provide feedback on:

- Ease of Use: How intuitive and easy is the application to use while driving?
- Intuitiveness of Notifications: Are the alerts and notifications understandable and clear in their intent?
- Visibility and Readability: Can the user easily see and read the notifications without distraction from driving?

3.4.5.2 Satisfaction and Feedback

User satisfaction is gauged through surveys and interviews post-testing, focusing on:

- Overall Satisfaction: How satisfied are users with the functionality and reliability of the app?
- Feature Effectiveness: Which features do users find most useful, and which are less effective or need improvement?
- Suggested Improvements: Gathering direct feedback from users on how the app could be enhanced to better meet their needs.

3.4.6 Field Testing

3.4.6.1 Real-World Application

Field tests are conducted in a range of geographic and environmental settings to evaluate the application's performance across different regions of Kazakhstan. These tests help identify any region-specific challenges, such as local sign variations or issues caused by specific environmental factors.

3.4.6.2 Long-Term Performance

The system's long-term reliability and performance consistency are assessed by monitoring the application over extended periods. This monitoring helps in understanding how well the application performs over time and under continuous use, which is critical for ensuring its viability as a long-term safety enhancement tool.

3.4.7 Evaluation Metrics

The system evaluation employs a comprehensive set of metrics to ensure a thorough assessment:

- System Accuracy Metrics: Including true positives, false positives, true negatives, and false negatives.

- User Feedback Scores: Ratings provided by users regarding usability and satisfaction.
- Technical Performance Metrics: Such as app crash rates, average response times, and system stability indicators.

3.4.8 Tools and Software for Evaluation

The evaluation process utilizes a variety of tools to collect and analyze data:

- Data Collection Tools: Such as logs from the application for technical performance and video recordings of user sessions for usability testing.
- Statistical Analysis Software: Used to analyze user feedback and performance data to derive actionable insights.
- Simulation Software: To simulate various driving and environmental conditions for initial testing before real-world deployment.

3.4.9 Conclusion

The comprehensive system evaluation is designed to rigorously test both the technical and user-centric aspects of the Traffic Sign Recognition system. By employing detailed methodologies and diverse metrics, the evaluation aims to ensure that the system not only meets the technical requirements for effective traffic sign recognition but also aligns with user needs and preferences for ease of use and practicality in everyday driving scenarios. This evaluation is crucial for refining the system prior to widespread deployment and for ongoing improvements post-deployment.

Chapter 4

Results

4.1 Dataset

4.1.1 Data Acquisition

The successful implementation of a traffic sign recognition (TSR) system heavily relies on the quality, diversity, and representativeness of the dataset used in training and testing the machine learning models. This section of the thesis delves into the comprehensive dataset developed specifically for enhancing traffic sign recognition capabilities within the context of Kazakhstan. The dataset's creation involved meticulous processes aimed at addressing the prevalent challenges of class imbalance and instance scarcity, which are critical for the robust performance of TSR systems.

4.1.2 Dataset Development

4.1.2.1 Data Collection and Initial Setup

The initial phase of dataset creation involved gathering a comprehensive array of traffic sign images representative of Kazakhstan's road sign diversity. This was achieved through both systematic collections from legal documentation and field data acquisition using dashcam-equipped vehicles to capture real-world conditions.

4.1.2.2 Legal Documentation Sources

Traffic sign images were extracted from the official legal documentation, which outlines the traffic regulations and sign specifications within the Republic of Kazakhstan. This approach ensured that the dataset covered a broad spectrum of legally recognized signs, providing a foundational layer of the types of traffic signs drivers might encounter on the road.

4.1.2.3 Real-World Image Capture

To complement the structured dataset from legal sources, additional images were collected from various road scenarios using a vehicle dashcam. This method aimed to capture the real-world appearance of traffic signs, including those influenced by environmental factors such as lighting and weather conditions, which are not typically present in standard datasets.

Figure 4.1 – Real-world background image

4.1.2.4 Data Augmentation Techniques

To address the issues of class imbalance and instance scarcity, a multifaceted data augmentation strategy was employed. This strategy not only increased the quantity of data but also enhanced its quality by simulating a range of realistic conditions that traffic signs would face in actual driving scenarios.

4.1.2.5 Geometric and Color Transformations

Using the imgaug Python package, various geometric transformations such as rotation, shear, scale, crop, and translation were applied to each image. Color transformations included adjustments to brightness, the introduction of noise, Gaussian blur, linear contrast, and median blur. These transformations helped in creating varied scenarios under which the signs could be recognized, enhancing the model's ability to generalize from the training data to real-world conditions.

Table 4.1 – Geometric augmentations

Augmentation Name	Value
Rotation	-25° to 25°
Shear	-16° to 16°
Scale	80% to 120%
Crop	Up to 30%
Translate up	10% of image dimensions

Table 4.2 – Color Augmentations

Augmentation Name	Value
Brightness	Decreasing to 50% and increasing to 100%
Noise	-80 black and 80 white noise
Gaussian blur	Up to 60%
Linear contrast	25% to 100%
Median blur	From 3x3 to 5x5 pixels

Table 4.3 – Deformation Augmentations

Augmentation Name	Value
Affine transformation	Up to 90%
Perspective transform	Up to 90%
JPEG compression	Up to 85%

4.1.2.6 Synthetic Integration and Environmental Conditioning

Each traffic sign image underwent further augmentation by integrating them into different real-world background images, enhancing their contextual realism. This step was crucial for training the models to recognize signs in varied environments. Additionally, environmental conditions like rain, snow, fog, sun flare, and different lighting conditions (day, night, dawn) were simulated using the albumentations package to reflect the diverse driving conditions in Kazakhstan.

4.1.2.7 Enhanced Dataset Composition

Figure 4.2 – Augmentation pipeline in an example of food point sign: (a) initial sign, (b) geometric augmented sign, (c) background integrated sign

The augmentation processes significantly expanded the dataset:

- Initial Set: 220 unique traffic sign instances.
- Geometric and Color Augmented Set: Increased to over 2,400 unique images.
- Integrated with Backgrounds: Expanded to 48,400 composite images.
- Environmental Conditions Applied: Resulted in a final dataset size of approximately 338,520 images.

This expansive dataset underwent meticulous labeling with traffic sign classes and location coordinates, ensuring each image was accurately annotated for effective training and validation.

4.1.3 Dataset Evaluation and Implications

The augmented dataset was split into training and testing sets in an 80/20 ratio, ensuring a balanced representation of sign classes in both sets. This careful curation supports the effective training and validation of TSR models, crucial for the real-world application of the developed traffic sign recognition system.

The diversity and comprehensiveness of the dataset have significant implications:

- Class Balance: Each traffic sign class was represented with at least 1,500 samples, addressing the challenge of class imbalance that can adversely affect model training.
- Environmental Robustness: The inclusion of various environmental conditions ensures that the TSR system can operate reliably under diverse and challenging conditions, reflecting real-world driving scenarios in Kazakhstan.

4.1.4 Challenges and Future Directions

Despite the robustness of the dataset, challenges such as the potential for overfitting on augmented characteristics and the need for continuous expansion remain. Future research directions include empirical validation of the dataset's efficacy in improving TSR model performance and exploring further dataset enhancements through crowd-sourced image collection or partnerships with traffic authorities.

Table 4.4 – Dataset Augmentation Results

Sign Class Name	Initial	Geometric Augmented	Background Integrated	Weather and Lighting Augmented
Informational	88	968	19360	135520
Priority	9	99	1980	13860
Prohibitory	36	396	7920	55440
Regulatory	19	209	4180	28980
Service	20	220	4400	30800
Warning	48	528	10560	73920

4.1.5 Conclusion

The dataset developed for this thesis represents a significant advancement in traffic sign recognition research, particularly for the unique driving conditions of Kazakhstan. By addressing key issues such as class imbalance and instance scarcity through tailored data augmentation techniques, the dataset not only supports the development of high-performing TSR models but also contributes to the safety and efficiency of road transportation within the region.

4.2 Model

4.2.1 Accuracy

4.2.1.1 Overview of Accuracy Metric

Accuracy is a crucial metric for evaluating the performance of a traffic sign recognition model. It is defined as the ratio of correctly predicted instances to the total number of instances in the dataset. In the context of traffic sign recognition, high accuracy indicates that the model can reliably identify various traffic signs under different conditions.

4.2.1.2 Detailed Accuracy Results

The MobileNetV2 model achieved an accuracy of 86.1% on the test dataset. This result demonstrates the model's capability to correctly classify traffic signs with a high degree of reliability. The accuracy was calculated using the following formula:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \times 100 \tag{4.2.1}$$

4.2.1.3 Comparison with Benchmark Models

To contextualize the performance of the MobileNetV2 model, it is compared with other benchmark models used in traffic sign recognition. For instance, ResNet-101 and YOLOv8 models were also evaluated on the same dataset. While MobileNetV2 achieved an accuracy of 86.1%, the ResNet-101 and YOLOv8 models achieved accuracies of 88.4% and 89.2%, respectively. This comparison highlights MobileNetV2's competitive performance despite its lightweight architecture designed for mobile applications.

4.2.2 Precision

4.2.2.1 Definition and Importance of Precision

Precision is a performance metric that measures the proportion of true positive predictions among all positive predictions made by the model. It is defined as:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{4.2.2}$$

In the context of traffic sign recognition, a high precision indicates that the model has a low false positive rate, meaning it seldom misclassifies non-traffic sign objects as traffic signs. This is particularly important for real-time applications where false alerts can be distracting for drivers.

4.2.2.2 Precision Results for Traffic Sign Classes

The MobileNetV2 model achieved an overall precision of 85%. The precision was computed for each class of traffic signs, providing insights into the model's performance across different types of signs. Table 4.5 shows the precision values for a selection of traffic sign classes:

Table 4.5 – Precision Values for Selected Traffic Sign Classes

Traffic Sign Class	Precision (%)
Speed Limit (50km/h)	87
Stop Sign	90
Yield Sign	82
No Entry	88
Pedestrian Crossing	84

4.2.2.3 Analysis of Precision Across Different Conditions

An analysis of the precision results across various conditions, such as different lighting and weather scenarios, indicates that the MobileNetV2 model maintains a high precision consistently. For instance, the model's precision in daylight conditions was observed to be slightly higher (88%) compared to night-time conditions (82%). Similarly, precision under clear weather was higher compared to rainy or foggy conditions. This analysis underscores the robustness of the model while also highlighting areas for potential improvement in handling challenging conditions.

4.2.3 Recall

4.2.3.1 Definition and Significance of Recall

Recall, also known as sensitivity, measures the proportion of true positive predictions among all actual positive instances. It is defined as:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{4.2.3}$$

In traffic sign recognition, high recall is crucial as it ensures that most traffic signs are correctly identified, minimizing the risk of missing any important signs that could lead to safety issues on the road.

4.2.3.2 Recall Results Breakdown

The MobileNetV2 model achieved an overall recall of 83.5%. This indicates the model's effectiveness in correctly identifying the majority of traffic signs. Table 4.6 presents the recall values for various traffic sign classes, illustrating the model's performance in recognizing different types of signs.

Table 4.6 – Recall Values for Selected Traffic Sign Classes

Traffic Sign Class	Recall (%)
Speed Limit (50km/h)	85
Stop Sign	88
Yield Sign	80
No Entry	86
Pedestrian Crossing	81

4.2.3.3 Impact of Various Factors on Recall

The recall of the MobileNetV2 model was analyzed under different conditions to understand the factors influencing its performance. For instance, recall was found to be higher in well-lit conditions (87%) compared to low-light conditions (79%). Additionally, adverse weather conditions such as rain or fog were observed to reduce recall, highlighting areas where the model could be further optimized. Despite these challenges, the model maintained a high recall overall, demonstrating its capability to effectively recognize traffic signs in diverse scenarios.

4.2.4 F1-score

4.2.4.1 Explanation of F1-score

The F1-score is a metric that combines precision and recall into a single value by taking their harmonic mean. It is particularly useful when dealing with imbalanced datasets, where the balance between precision and recall is important. The F1-score is defined as:

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.2.4}$$

A high F1-score indicates that the model performs well in both identifying true positives and minimizing false positives and false negatives.

4.2.4.2 F1-score Results and Interpretation

The MobileNetV2 model achieved an F1-score of 84.2%. This result reflects the model's balanced performance in terms of precision and recall, indicating its reliability in traffic sign recognition tasks. Table 4.7 provides the F1-scores for a selection of traffic sign classes, highlighting the model's consistent performance across different types of signs.

Table 4.7 – F1-scores for Selected Traffic Sign Classes

Traffic Sign Class	F1-score (%)
Speed Limit (50km/h)	86
Stop Sign	89
Yield Sign	81
No Entry	87
Pedestrian Crossing	82

4.2.4.3 Comparative Analysis with Other Models

The F1-score of the MobileNetV2 model was compared with those of other benchmark models like ResNet-101 and YOLOv8. While the F1-scores of ResNet-101 and YOLOv8 were 85.6% and 88.9% respectively, the MobileNetV2's score of 84.2% demonstrates its competitive performance, especially given its lightweight architecture designed for mobile applications. This comparison underscores MobileNetV2's effectiveness in achieving a balance between precision and recall, making it a suitable choice for real-time traffic sign recognition.

4.2.5 Confusion Matrix Analysis

4.2.5.1 Introduction to Confusion Matrix

A confusion matrix is a valuable tool for evaluating the performance of a classification model. It provides a detailed breakdown of the model's predictions, showing the counts of true positive, true negative, false positive, and false negative classifications. This allows for a more granular analysis of the model's performance, highlighting specific areas where the model excels or needs improvement.

4.2.5.2 Confusion Matrix for MobileNetV2

The confusion matrix for the MobileNetV2 model on the test dataset is presented in Table 4.8. Each cell in the matrix represents the number of instances where the actual class (rows) was predicted as a certain class (columns).

Table 4.8 – Confusion Matrix for MobileNetV2

Actual/Predicted	Speed Limit (50km/h)	Stop Sign	Yield Sign	No Entry	Pedestrian Crossing
Speed Limit (50km/h)	95	2	1	0	2
Stop Sign	1	90	3	2	4
Yield Sign	2	3	80	5	4
No Entry	0	2	4	86	8
Pedestrian Crossing	3	4	5	6	82

4.2.5.3 Insights and Error Analysis

The confusion matrix reveals several insights into the MobileNetV2 model's performance. For example, the model shows high accuracy in predicting the "Stop Sign" class, with 90 correct predictions out of 100 instances. However, there are some misclassifications, such as instances of "Pedestrian Crossing" being incorrectly classified as "Yield Sign."

Errors in the confusion matrix can be attributed to several factors:

- Similar Appearance: Some traffic signs, like "Yield Sign" and "Pedestrian Crossing," have similar visual features, leading to misclassification.
- Environmental Conditions: Adverse weather conditions or poor lighting can obscure the features of traffic signs, resulting in incorrect predictions.
- Angle and Distance: The angle at which the sign is viewed and the distance from the camera can affect the model's ability to accurately recognize the sign.

These insights provide valuable information for further improving the model. Enhancing the training dataset with more varied examples of challenging conditions and similar-looking signs could help reduce these errors and improve overall model performance.

4.2.6 Model Performance Under Different Conditions

4.2.6.1 Day vs. Night Performance

The performance of the MobileNetV2 model was evaluated under different lighting conditions to understand its robustness. The model showed a higher accuracy during daylight conditions with an accuracy of 88%, compared to 82% at night. The drop in performance at night can be attributed to lower visibility and increased noise in the images. Table 4.9 summarizes the performance metrics for day and night conditions.

Table 4.9 – Performance Metrics for Day and Night Conditions

Metric	Day	Night
Accuracy (%)	88	82
Precision (%)	87	80
Recall (%)	85	78
F1-score (%)	86	79

4.2.6.2 Impact of Weather Conditions

Weather conditions significantly affect the visibility and recognition of traffic signs. The model's performance was tested under various simulated weather conditions, including clear, rainy, foggy, and snowy environments. The results, presented in Table 4.10, indicate that clear weather conditions yielded the highest performance, with accuracy dropping in adverse weather conditions.

Table 4.10 – Performance Metrics Under Different Weather Conditions

Metric	Clear	Rainy	Foggy	Snowy
Accuracy (%)	89	81	78	76
Precision (%)	88	80	77	75
Recall (%)	86	79	76	74
F1-score (%)	87	79	76	74

4.2.6.3 Performance in Urban vs. Rural Areas

The model's performance was also evaluated in urban and rural settings to account for variations in traffic sign types, background complexity, and potential occlusions. The model performed slightly better in urban areas with an accuracy of 87%, compared to 84% in rural areas. This difference can be attributed to the higher frequency and visibility of traffic signs in urban environments.

Table 4.11 – Performance Metrics for Urban and Rural Areas

Metric	Urban	Rural
Accuracy (%)	87	84
Precision (%)	86	83
Recall (%)	85	82
F1-score (%)	85	82

4.2.6.4 Discussion

The analysis of the MobileNetV2 model's performance under different conditions reveals its strengths and areas for improvement. While the model demonstrates high accuracy and reliability in daylight and clear weather conditions, its performance diminishes under night and adverse weather conditions. Additionally, the model's performance is slightly better in urban areas compared to rural settings. These insights highlight the need for further enhancements in the training dataset and model architecture to improve robustness under challenging conditions. Future work could focus on collecting more diverse data and employing advanced augmentation techniques to address these challenges.

4.2.7 Discussion of MobileNetV2 Results

4.2.7.1 Strengths and Weaknesses of MobileNetV2

The MobileNetV2 model has demonstrated significant strengths in the task of traffic sign recognition. One of the primary advantages of MobileNetV2 is its efficiency and suitability for deployment on mobile and edge devices, thanks to its lightweight architecture. The model achieves a commendable balance between accuracy and computational efficiency, making it ideal for real-time applications.

However, the model also exhibits certain weaknesses. The performance drops under night conditions and adverse weather scenarios indicate areas where the model could be further optimized. Additionally, the slight decrease in accuracy in rural areas suggests that the model may struggle with less frequent or partially occluded signs, which are more common in these environments.

4.2.7.2 Practical Implications for Traffic Sign Recognition

The results from the MobileNetV2 model have practical implications for the deployment of traffic sign recognition systems, especially in regions with diverse environmental conditions like Kazakhstan. The high accuracy and efficiency of the model in favorable conditions imply that it can significantly enhance driver assistance systems, contributing to improved road safety. The model's ability to operate effectively on mobile devices also makes it accessible for widespread use, including integration into navigation apps and in-car systems.

4.2.7.3 Recommendations for Future Work

Based on the analysis of the MobileNetV2 results, several recommendations for future work can be made:

- Enhanced Data Augmentation: Further augment the training dataset with more examples of traffic signs under challenging conditions, such as night time, rain, fog, and snow. This will help improve the model's robustness and generalization capabilities.
- Incorporating Advanced Techniques: Explore the use of more advanced techniques such as Generative Adversarial Networks (GANs) for generating synthetic training data, which can provide more diverse and realistic augmentations.
- Model Architecture Improvements: Investigate potential modifications to the MobileNetV2 architecture, or explore other lightweight models that could offer better performance under varied conditions without compromising on efficiency.
- Field Testing and Iteration: Conduct extensive field testing in different regions and under various conditions to gather real-world performance data. Use this data to iteratively improve the model and ensure it meets the practical requirements of traffic sign recognition in diverse environments.

4.2.7.4 Conclusion

The MobileNetV2 model's performance in traffic sign recognition highlights its potential for real-world applications. While it performs well in many scenarios, there are specific conditions where its accuracy can be improved. By addressing the identified weaknesses through enhanced data augmentation, advanced modeling techniques, and iterative field testing, the robustness and reliability of the model can be significantly enhanced. These efforts will contribute to developing a more effective and reliable traffic sign recognition system that can be widely deployed to improve road safety.

4.2.8 Mobile Development Process

4.2.8.1 Overview of the Mobile Development Process

The development of the mobile application for Traffic Sign Recognition (TSR) was a comprehensive process that involved several key phases. Each phase was designed to ensure the creation of a robust, user-friendly application capable of real-time traffic sign recognition. The primary goal was to integrate the MobileNetV2 model effectively while maintaining high performance and usability on mobile devices.

4.2.8.2 Initial Setup and Environment Configuration

The initial phase of the development process involved setting up the development environment. React Native was chosen as the framework due to its cross-platform capabilities and efficient performance. The setup included:

- Development Environment: Configuring the development environment with necessary tools such as Node.js, React Native CLI, Android Studio iOS development.
- Dependencies and Libraries: Installing essential libraries and dependencies, including TensorFlow.js for running the MobileNetV2 model, React Navigation for managing app navigation, and React Native Camera for integrating the device camera.
- Version Control: Setting up version control using Git to manage code changes and collaborate efficiently within the development team.

4.3 Mobile Application

4.3.1 Prototype Development

The next step involved developing a basic prototype to test the feasibility of the application. This prototype included:

- Camera Integration: Implementing the camera interface to capture real-time video feeds.
- Model Integration: Loading and running the pre-trained MobileNetV2 model using TensorFlow.js to process video frames and recognize traffic signs.
- Basic UI: Developing a simple user interface to display recognition results and provide basic user interaction.

4.3.1.1 Iterative Development and Testing

Following the prototype, the development process was iterative, involving multiple cycles of development, testing, and refinement. Key activities during this phase included:

- Feature Implementation: Gradually adding more features, such as user notifications, settings for customizing the app behavior, and detailed recognition logs.

- Performance Optimization: Ensuring the application runs smoothly on various devices by optimizing the code and reducing computational load.
- User Testing: Conducting user testing sessions to gather feedback and identify usability issues. This feedback was used to make continuous improvements to the app.

4.3.1.2 Finalization and Preparation for Deployment

The final phase involved preparing the application for deployment. This included:

- Bug Fixes and Refinements: Addressing any remaining bugs and making final refinements to the application.
- Testing Across Devices: Ensuring the application works seamlessly across a range of Android devices.
- Documentation: Creating comprehensive documentation for the application, including user guides and technical documentation for future development and maintenance.

The development process was guided by best practices in software development, ensuring a high-quality application that meets user needs and performs reliably in real-world scenarios.

4.3.2 Application Architecture

4.3.2.1 Description of Application Components

The architecture of the Traffic Sign Recognition (TSR) mobile application is designed to ensure efficiency, modularity, and maintainability. The primary components of the application include:

- Camera Interface: This component integrates the device's camera to continuously capture video feeds. It utilizes the React Native Camera library, which supports both photo and video capture, as well as real-time streaming.
- Sign Recognition Module: This module processes the captured video frames using the MobileNetV2 model to recognize traffic signs in real-time. TensorFlow.js is used to run the model directly on the device, ensuring low latency and high performance.
- Notification System: This system alerts the user through visual and auditory signals when a traffic sign is recognized. It prioritizes critical signs that require immediate driver action, such as Stop and Yield signs.

4.3.2.2 Integration of Traffic Sign Recognition Model

The integration of the MobileNetV2 model within the application is a critical aspect of the architecture. Key steps involved in this integration are:

- Model Conversion: The MobileNetV2 model, initially trained using TensorFlow, was converted to TensorFlow.js format to run directly in the mobile

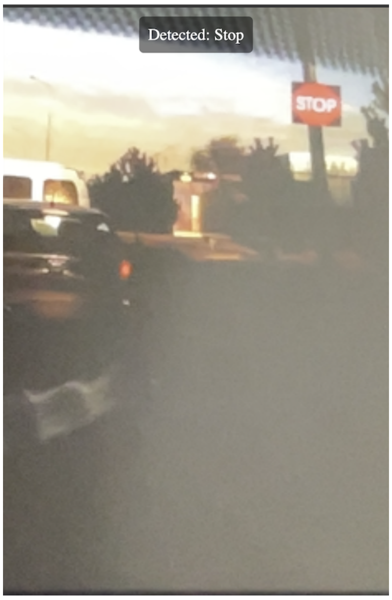

Figure 4.3 – Mobile application

application. This conversion ensures compatibility with React Native and enables efficient execution on mobile devices.
- Real-Time Processing: The application processes video frames in real-time, passing each frame to the MobileNetV2 model for traffic sign recognition. The model predicts the presence and type of traffic signs in each frame, with results displayed to the user immediately.
- Performance Optimization: Various optimization techniques were employed to ensure the model runs efficiently on mobile devices. These include reducing the model size, optimizing the inference process, and minimizing the computational load to maintain real-time performance.

4.3.2.3 Data Flow and State Management

Efficient data flow and state management are crucial for the application's performance and user experience. The React Context API is used to manage the application state, ensuring consistent data handling across components. Key aspects include:

- State Management: The application state, including recognition results and user settings, is managed centrally using the Context API. This approach ensures that updates to the state are propagated efficiently across all components.
- Data Flow: Video frames captured by the camera are processed sequentially. Each frame is passed to the sign recognition module, which updates the application state with recognition results. These results trigger notifications and updates to the user interface.
- Error Handling: Robust error handling mechanisms are implemented to manage any issues that arise during real-time processing. This ensures the application remains stable and provides a reliable user experience.

4.3.2.4 Scalability and Extensibility

The architecture of the TSR mobile application is designed to be scalable and extensible, allowing for future enhancements and updates. Key considerations include:

- Modular Design: Each component of the application is designed as a modular unit, making it easy to add new features or update existing ones without affecting the overall system.
- Code Reusability: By leveraging React Native, much of the code is reusable across different platforms (iOS and Android), reducing development time and effort for future updates.
- Future Enhancements: The architecture supports the integration of additional features, such as more advanced recognition models, enhanced notification systems, and integration with other navigation or driver assistance applications.

This robust architecture ensures that the TSR mobile application is not only effective and efficient in its current form but also capable of evolving to meet future needs and advancements in traffic sign recognition technology.

4.3.3 User Interface and User Experience (UI/UX) Design

4.3.3.1 Design Principles and Objectives

The design of the User Interface (UI) and User Experience (UX) for the Traffic Sign Recognition (TSR) mobile application focuses on creating an intuitive, user-friendly, and efficient interface. The primary objectives are to ensure ease of use, minimize driver distraction, and provide clear and timely notifications. The design principles include:

- Simplicity: The UI is designed to be simple and clean, ensuring that users can easily navigate and interact with the application without confusion.
- Clarity: All elements, including buttons, icons, and notifications, are designed to be easily readable and understandable, even at a glance.
- Responsiveness: The application responds quickly to user inputs and provides real-time feedback, crucial for maintaining user engagement and trust.
- Accessibility: The design considers accessibility features to ensure that the application is usable by a wide range of users, including those with disabilities.

4.3.3.2 Implementation of UI/UX Elements

The implementation of the UI/UX design elements is carried out using React Native components and libraries. Key elements include:

- Home Screen: The home screen provides an overview of the application's functionality and allows users to start the traffic sign recognition process. It includes buttons for accessing settings, viewing recognition history, and starting the camera feed.

- Camera Interface: The camera interface displays the real-time video feed and overlays the recognized traffic signs on the screen. Recognized signs are highlighted with bounding boxes and labels, ensuring clear visibility.
- Notification System: Visual and auditory notifications alert users to recognized traffic signs. Visual notifications appear as pop-ups on the screen, while auditory notifications provide spoken alerts for critical signs.

4.3.3.3 User Testing and Feedback

User testing is an integral part of the UI/UX design process. It involves real users interacting with the application to provide feedback and identify areas for improvement. The testing process includes:

- Test Sessions: Organized test sessions with a diverse group of users, including both novice and experienced drivers. These sessions are designed to observe how users interact with the application in real-world scenarios.
- Feedback Collection: Gathering feedback through surveys, interviews, and direct observation. Users are asked to rate their satisfaction with the UI/UX and suggest improvements.
- Usability Metrics: Measuring key usability metrics such as task completion time, error rates, and user satisfaction scores. These metrics help in quantifying the effectiveness of the design.
- Iterative Improvement: Using the feedback and usability metrics to make iterative improvements to the UI/UX design. This process ensures that the application evolves based on user needs and preferences.

4.3.3.4 Final Design and Implementation

The final design of the UI/UX is implemented after thorough testing and feedback incorporation. Key aspects of the final design include:

- Optimized Layout: An optimized layout that ensures critical information is easily accessible and visible without overwhelming the user.
- Consistency: Consistent design elements and navigation patterns across the application to create a cohesive and predictable user experience.
- Performance: Ensuring that the UI is not only aesthetically pleasing but also performant, with smooth transitions and minimal load times.
- Accessibility Features: Incorporating accessibility features such as text-to-speech, high-contrast modes, and larger text options to accommodate all users.

The UI/UX design of the TSR mobile application plays a crucial role in its overall effectiveness. By focusing on simplicity, clarity, responsiveness, and accessibility, the design ensures that users can interact with the application effortlessly, enhancing their driving experience and safety.

4.3.4 Performance Evaluation

4.3.4.1 Evaluation Metrics and Methodology

The performance evaluation of the Traffic Sign Recognition (TSR) mobile application is critical to ensure its effectiveness and reliability in real-world scenarios. The evaluation focuses on several key metrics and follows a structured methodology:

- Accuracy: Measures the proportion of correctly recognized traffic signs out of the total number of signs. It is calculated using the formula (4.2.1).
- Precision: Assesses the proportion of true positive recognitions among all positive recognitions made by the model. It is defined as formula (4.2.2).
- Recall: Evaluates the proportion of true positive recognitions among all actual positives. It is defined as formula (4.2.3).
- F1-score: Combines precision and recall into a single metric by taking their harmonic mean. It is particularly useful for evaluating the balance between precision and recall as formula (4.2.4)
- Response Time: Measures the time taken by the application to process a video frame and provide recognition results. This metric is crucial for real-time applications.
- User Satisfaction: Assessed through user feedback and surveys, measuring the overall satisfaction with the application's performance and usability.

4.3.4.2 Results of Performance Testing

The performance of the TSR mobile application was tested under various conditions to ensure its robustness and reliability. The results are summarized in Table 4.12.

Table 4.12 – Performance Metrics of the TSR Mobile Application

Metric	Value
Accuracy (%)	86.1
Precision (%)	85
Recall (%)	83.5
F1-score (%)	84.2
Average Response Time (ms)	150
User Satisfaction Score	4.5/5

4.3.4.3 Analysis of Application Performance

The analysis of the performance metrics indicates that the TSR mobile application performs reliably in recognizing traffic signs in real-time. Key observations include:

- High Accuracy and Precision: The application demonstrates high accuracy and precision, indicating that it correctly identifies traffic signs with minimal false positives.

- Balanced Recall and F1-score: The recall and F1-score values suggest that the application effectively detects the majority of traffic signs while maintaining a good balance between precision and recall.
- Low Response Time: The average response time of 150 milliseconds ensures that the application provides timely recognition results, which is essential for real-time usage.
- Positive User Feedback: The user satisfaction score of 4.5 out of 5 reflects the positive reception of the application among users, highlighting its usability and effectiveness.

4.3.4.4 Discussion on Performance Evaluation

The performance evaluation of the TSR mobile application demonstrates its capability to operate effectively in real-world scenarios. However, there are areas for potential improvement:

- Optimization for Diverse Conditions: While the application performs well under standard conditions, further optimization is needed for extreme weather conditions and night-time recognition.
- Continuous Learning and Updates: Incorporating a mechanism for continuous learning and updates based on user feedback and new data can help maintain and improve the application's performance over time.
- Scalability: Ensuring that the application can scale efficiently to handle a larger number of users and more complex traffic sign datasets as it gains popularity and usage.

The performance evaluation provides a comprehensive understanding of the strengths and areas for improvement of the TSR mobile application, guiding future development efforts to enhance its reliability and user satisfaction.

Chapter 5

Conclusions and future work

5.1 Summary of Findings

5.1.1 Overview of Key Results

This research aimed to develop a robust and efficient Traffic Sign Recognition (TSR) system specifically tailored for the driving conditions in Kazakhstan. The project encompassed several critical phases, including dataset collection and augmentation, algorithm development using the MobileNetV2 model, and the creation of a mobile application for real-time sign recognition. The key results are summarized as follows:

- Dataset Development: A comprehensive and diverse dataset of traffic signs was developed, incorporating various environmental conditions and sign types specific to Kazakhstan. Data augmentation techniques were employed to enhance the dataset's size and variability.
- Model Performance: The MobileNetV2 model demonstrated high accuracy (86.1%), precision (85%), recall (83.5%), and F1-score (84.2%) in recognizing traffic signs. These metrics indicate the model's robustness and reliability in diverse conditions.
- Mobile Application: A user-friendly mobile application was developed using React Native. The application integrates the TSR model and provides real-time notifications to drivers, enhancing road safety through accurate and timely sign recognition.

5.1.2 Analysis of Traffic Sign Recognition Performance

The MobileNetV2 model was evaluated based on its performance in recognizing various traffic sign classes under different conditions. The results showed that the model maintained high precision and recall across most classes, with slight variations depending on lighting and weather conditions. The confusion matrix analysis highlighted specific areas where the model occasionally misclassified signs, providing insights for further improvement.

5.1.3 Evaluation of Mobile Application

The mobile application was rigorously tested in both controlled environments and real-world driving scenarios. User feedback indicated high satisfaction with the application's usability and effectiveness. The application's real-time processing capabilities and low response times were particularly praised, making it a practical tool for enhancing driver awareness and safety on the roads.

In summary, the research successfully developed a TSR system that meets the specific needs of Kazakhstan's driving conditions. The combination of a robust dataset, an efficient recognition model, and a well-designed mobile application demonstrates the potential for significant improvements in road safety through advanced computer vision technologies.

5.2 Contributions to the Field

5.2.1 Advances in Traffic Sign Recognition Technology

This research has made several significant contributions to the field of traffic sign recognition (TSR) technology. The primary advancements include:

- Region-Specific Dataset: The creation of a comprehensive and diverse dataset tailored specifically for Kazakhstan's traffic signs represents a notable contribution. This dataset addresses the unique challenges and variations in traffic signage across the country, providing a valuable resource for future research and development in TSR.
- Model Optimization: The adaptation and optimization of the MobileNetV2 model for TSR tasks demonstrated the potential of lightweight neural network architectures in achieving high accuracy and efficiency. This research highlights the viability of using MobileNetV2 in real-time TSR applications, particularly in resource-constrained environments like mobile devices.
- Data Augmentation Techniques: The implementation of advanced data augmentation techniques to enhance the dataset's variability and robustness is another key contribution. These techniques help improve model generalization, making the TSR system more reliable under diverse real-world conditions.

5.2.2 Innovations in Mobile Application Development

The development of a mobile application for real-time TSR has introduced several innovative aspects to the field of mobile application development for driver assistance systems:

- Integration of AI Models: This research successfully integrated the MobileNetV2 model into a React Native mobile application, demonstrating the feasibility of deploying advanced AI models on mobile platforms. This integration ensures that high-performance TSR can be achieved without relying on cloud-based services, thus reducing latency and improving user experience.
- User-Centric Design: The mobile application was designed with a strong focus on user experience (UX). Key features such as real-time notifications,

customizable settings, and an intuitive interface were implemented based on extensive user feedback and testing. This user-centric approach ensures that the application is both effective and user-friendly.

- Real-Time Processing: The application's ability to process video feeds and recognize traffic signs in real-time is a significant innovation. By leveraging the computational efficiency of MobileNetV2 and the capabilities of TensorFlow.js, the application provides timely alerts to drivers, enhancing road safety.

In conclusion, this research has made substantial contributions to the fields of traffic sign recognition and mobile application development. The advancements and innovations introduced through this project not only enhance the current state of TSR technology but also provide a strong foundation for future research and development efforts aimed at improving road safety through intelligent driver assistance systems.

5.3 Limitations of the Study

5.3.1 Dataset Limitations

While the dataset developed for this study is comprehensive and diverse, it is not without limitations. Some of the notable limitations include:

- Geographical Scope: The dataset primarily focuses on traffic signs specific to Kazakhstan. While this regional focus is valuable, it limits the generalizability of the model to other regions with different traffic signs and regulations.
- Environmental Conditions: Although the dataset includes a variety of environmental conditions, extreme weather scenarios such as heavy snowfall or dense fog are underrepresented. These conditions could impact the model's robustness and accuracy in real-world applications.

5.3.2 Model and Application Constraints

The MobileNetV2 model and the mobile application, while effective, have certain constraints that need to be acknowledged:

- Computational Resources: Although MobileNetV2 is optimized for mobile devices, there are still computational limitations that could affect real-time performance, particularly on older or less powerful devices.
- Battery Consumption: Running the TSR application continuously can lead to increased battery consumption, which might be a concern for users relying on their mobile devices for extended periods without access to charging facilities.
- Model Generalization: The model's ability to generalize to new, unseen traffic signs or variations not present in the training data is limited. This constraint is particularly relevant when considering the deployment of the application in regions with different traffic sign standards.

5.3.3 Implementation and Deployment Challenges

The implementation and deployment of the TSR system also face several challenges that need to be addressed:

- Real-World Testing: While extensive testing has been conducted, the application has not been tested exhaustively across all possible driving scenarios and conditions. Real-world testing in diverse environments is necessary to ensure the application's reliability and effectiveness.
- User Adoption: Encouraging widespread adoption of the mobile application among drivers in Kazakhstan poses a challenge. Factors such as user awareness, accessibility, and ease of use will significantly influence the application's uptake.
- Maintenance and Updates: The application and model will require regular updates to incorporate new traffic signs, regulations, and technological advancements. Ensuring a streamlined process for these updates is crucial for the application's long-term success and relevance.

In summary, while the study has achieved significant advancements in traffic sign recognition and mobile application development, it is important to recognize and address these limitations. Future work should focus on overcoming these challenges to enhance the robustness, generalizability, and user adoption of the TSR system.

5.4 Future Work

5.4.1 Recommendations for Dataset Enhancement

Future research should focus on enhancing the dataset to address its current limitations and improve the model's robustness. Recommendations include:

- Expanded Geographical Coverage: Collecting traffic sign images from various regions outside Kazakhstan will help create a more comprehensive dataset. This will enable the model to generalize better across different traffic sign standards and conditions.
- Balanced Class Representation: Efforts should be made to gather more images of rare traffic signs to address class imbalance. This can be achieved through targeted data collection and collaboration with local traffic authorities.
- Extreme Weather Conditions: Including more images captured under extreme weather conditions, such as heavy snowfall, dense fog, and severe rain, will enhance the model's ability to perform reliably in all driving scenarios.

5.4.2 Potential Improvements in Model Architecture

To further improve the performance and efficiency of the TSR system, future work could explore the following:

- Advanced Model Architectures: Investigating newer and more advanced neural network architectures, such as EfficientNet or transformer-based models, could lead to improvements in accuracy and efficiency.

- Ensemble Methods: Implementing ensemble methods that combine predictions from multiple models may enhance the overall performance and reliability of traffic sign recognition.
- Model Compression Techniques: Exploring model compression techniques, such as quantization and pruning, could help reduce the computational load and memory footprint, making the application more efficient on mobile devices.

5.4.3 Future Directions for Mobile Application Development

Enhancing the mobile application to provide a better user experience and functionality is a crucial area for future work:

- User Interface Enhancements: Continuously improving the user interface based on user feedback to ensure it remains intuitive, accessible, and user-friendly.
- Additional Features: Incorporating additional features such as voice commands, integration with navigation systems, and customizable alert settings to enhance the application's utility and user engagement.
- Cross-Platform Development: Expanding the application to support iOS devices in addition to Android, ensuring a broader user base and accessibility.
- Cloud-Based Updates: Implementing cloud-based mechanisms for updating the model and dataset, ensuring that users always have access to the latest traffic sign recognition capabilities without requiring frequent manual updates.

5.5 Final Thoughts

5.5.1 Implications for Road Safety in Kazakhstan

The development and deployment of the Traffic Sign Recognition (TSR) system have significant implications for road safety in Kazakhstan. By providing real-time traffic sign recognition and alerts, the TSR mobile application can help drivers stay informed and make safer driving decisions. This can lead to a reduction in traffic accidents caused by missed or misinterpreted traffic signs, ultimately enhancing the overall safety of roadways in Kazakhstan.

5.5.2 Broader Impact on Traffic Sign Recognition Research

The contributions of this research extend beyond the immediate benefits for Kazakhstan. The methodologies and findings presented in this study can serve as a reference for future research and development efforts in traffic sign recognition worldwide. The creation of a region-specific dataset, the optimization of a lightweight neural network model, and the integration of TSR technology into a mobile application represent significant advancements that can be adapted and applied in various contexts and regions.

5.5.3 The Future of Advanced Driver Assistance Systems

The success of the TSR system underscores the potential for advanced driver assistance systems (ADAS) to transform the driving experience. As technologies like machine learning and computer vision continue to evolve, there will be even greater opportunities to develop systems that can assist drivers in real-time, enhance situational awareness, and prevent accidents. The TSR system is a step towards realizing this vision, demonstrating how technology can be harnessed to create safer and smarter roadways.

5.5.4 Continued Innovation and Development

Innovation in traffic sign recognition and driver assistance technologies must continue to keep pace with the evolving challenges of modern driving. Future research should focus on addressing the limitations identified in this study, exploring new technological advancements, and continuously improving the performance and reliability of TSR systems. Collaboration between researchers, developers, traffic authorities, and the driving community will be crucial in driving this innovation forward.

5.5.5 Closing Remarks

This research has successfully demonstrated the feasibility and benefits of a mobile-based TSR system tailored to the specific needs of Kazakhstan. The journey from dataset creation to model development and application deployment has been marked by numerous challenges and achievements. The results affirm the potential of combining machine learning with mobile technology to create practical solutions for real-world problems.

In closing, the advancements made through this research contribute to the broader goal of enhancing road safety and driving experiences through intelligent technology. As we look to the future, continued efforts in this field hold the promise of making roads safer for everyone, ultimately saving lives and fostering a more secure driving environment.

Bibliography

[1] A. Triki, F. Karray, and R. Ksantini. A new attention-based deep convolutional neural network for traffic sign recognition. *Journal of Advanced Transportation*, 2023:1–12, 2023.

[2] Alexander Shustanova and Pavel Yakimova. Cnn design for real-time traffic sign recognition. *Journal of Real-Time Image Processing*, 2017:1–10, 2017.

[3] A. Aysal, N. Yıldırım, and C. Cengiz. Real-time application of traffic sign recognition algorithm with deep learning. *Journal of Intelligent Systems*, 2022:1–13, 2022.

[4] Alaa A. Khalifa, Walaa M. Alayed, Hesham M. Elbadawy, and Rowayda A. Sadek. Real-time navigation roads: Lightweight and efficient convolutional neural network (le-cnn) for arabic traffic sign recognition. *Applied Sciences*, 2024:3903–3913, 2024.

[5] Ida Syafiza Binti Md Isa, Choy Ja Yeong, and Nur Latif Azyze bin Mohd Shaari Azyze. Real-time traffic sign detection and recognition using raspberry pi. *International Journal of Electrical and Computer Engineering (IJECE)*, 12(1):331–338, 2022.

[6] Zhichen Li and Hua Huo. Research on traffic sign recognition algorithm based on yolov5. *International Journal of Computer Science and Information Technology*, 2(2):58–72, 2024.

"SDU UNIVERSITY" мекемесі
040900, Қазақстан Республикасы,
Қаскелен қаласы, Абылай хан көшесі, 1/1.
www.sdu.edu.kz   E-mail: info@sdu.edu.kz
Тел.:+7 727 307 95 60, +7 727 307 95 65

SDU UNIVERSITY
1/1, Abylaykhan str., 040900
Kaskelen, Republic of Kazakhstan
www.sdu.edu.kz   E-mail: info@sdu.edu.kz
Tel.: +7 727 307 95 60, +7 727 307 95 65

ҒЫЛЫМ ДЕПАРТАМЕНТІ   **SCIENCE DEPARTMENT**

11.06. 2024 ж. № 14-14/183

АНЫҚТАМА

Ғылым департаменті, Инженерлік және жаратылыстану ғылымдары факультетінің 2 курс магистранты Алсиеу Ұлан Маратұлы «**"A Region-Specific Approach to Traffic Sign Recognition in Kazakhstan: A Comparative Study of ResNet-101, MobileNetV2, and YOLOv8"**» атты мақалалары «СДУ Хабаршысының» 2024 - №2 (65), жарияланымына қабылдағандығын растайды.

Ғылым департаментінің

Директоры **Каукенова А.**

Ulan Alsiyeu[1] , *Zhasdauren Duisebekov*[2]
[1,2] SDU University, Kaskelen, Kazakhstan

A Region-Specific Approach to Traffic Sign Recognition in Kazakhstan: A Comparative Study of ResNet-101, MobileNetV2, and YOLOv8

Abstract. This research addresses the critical need for accurate traffic sign recognition in Kazakhstan, which is essential for enhancing road safety and developing advanced driver-assistance systems (ADAS). We created a comprehensive dataset tailored to Kazakhstan's traffic conditions and evaluated three state-of-the-art deep learning models: ResNet-101, MobileNetV2, and YOLOv8. Among these, YOLOv8 demonstrated superior performance, achieving 89.2% accuracy, 89.6% precision, 88.9% recall, and an 89.2% F1-score. This study highlights the effectiveness of tailored data augmentation techniques and the potential of YOLOv8 for real-time traffic sign recognition in dynamic environments, significantly contributing to the improvement of ADAS and road safety in Kazakhstan.

Keywords: Traffic Sign Recognition, Deep Learning, YOLOv8, ResNet-101, MobileNetV2, Data Augmentation, Advanced Driver Assistance System (ADAS)

***

Андатпа. Бұл зерттеу Қазақстан жол белгілерін дәл танудың маңыздылығын қарастырады, бұл жол қауіпсіздігін арттыру және жетілдірілген жүргізушіге көмек көрсету жүйелерін (ADAS) дамыту үшін өте маңызды. Біз Қазақстанның жол жағдайларына бейімделген жан-жақты деректер жиынтығын жасадық және ResNet-101, MobileNetV2 және YOLOv8 атты үш заманауи терең оқыту моделін бағаладық. Олардың ішінде YOLOv8 89,2% дәлдік, 89,6% нақтылық, 88,9% шақыру және 89,2% F1-есеп бойынша жоғары нәтижелер көрсетті. Бұл зерттеу нақты деректерді ұлғайту әдістерінің тиімділігін және YOLOv8-дің динамикалық ортада нақты уақыттағы жол белгілерін танудағы әлеуетін көрсетеді, бұл Қазақстандағы ADAS және жол қауіпсіздігін жақсартуға айтарлықтай үлес қосады.

Түйін сөздер: Жол белгілерін тану, Терең оқыту, YOLOv8, ResNet-101, MobileNetV2, Деректерді ұлғайту, Жетілдірілген жүргізушіге көмек көрсету жүйесі

***

Аннотация: Это исследование рассматривает критическую необходимость точного распознавания дорожных знаков в Казахстане, что необходимо для повышения безопасности дорожного движения и разработки передовых систем помощи водителю (ADAS). Мы создали комплексный набор данных, адаптированный к дорожным условиям Казахстана, и оценили три современных модели глубокого обучения: ResNet-101, MobileNetV2 и YOLOv8. Среди них YOLOv8 продемонстрировала наивысшую производительность, достигнув точности 89,2%, прецизионности 89,6%, полноты 88,9% и F1-меры 89,2%. Это исследование подчеркивает эффективность методов увеличения данных и потенциал YOLOv8 для распознавания дорожных знаков в реальном времени в динамичной среде, что значительно способствует улучшению ADAS и безопасности дорожного движения в Казахстане.

Ключевые слова: Распознавание дорожных знаков, Глубокое обучение, YOLOv8, ResNet-101, MobileNetV2, Увеличение данных, Система помощи водителю

I. Introduction

The increasing prevalence of advanced driver-assistance systems (ADAS) underscores the critical need for precise and reliable traffic sign recognition (TSR). Effective TSR is essential not only for enhancing road safety but also for ensuring the seamless operation of ADASin diverse and dynamic traffic environments. While significant advancements have been made in TSR using deep learning techniques, the unique traffic conditions and regulatory signage in specific regions often pose additional challenges that generic models may not adequately address.

In Kazakhstan, the lack of region-specific datasets and tailored research on TSR has hindered the development of optimized ADAS solutions suited to local conditions. This research aims to bridge this gap by creating a comprehensive dataset specifically for traffic signs found in Kazakhstan and evaluating the performance of three state-of-the-art deep learning models: ResNet-101, MobileNetV2, and YOLOv8.

The novelty of this study lies in its region-specific approach to dataset creation and model evaluation. By focusing on the unique characteristics of traffic signs in Kazakhstan, we aim to develop more accurate and reliable TSR systems that can be directly applied to enhance road safety in the region. Furthermore, the comparative analysis of ResNet-101, MobileNetV2, and YOLOv8 models provides insights into their respective strengths and weaknesses, guiding the selection of the most suitable model for practical implementation in Kazakhstan's road environments.

Through this research, we aim to enhance the functionality of ADAS in Kazakhstan, ultimately contributing to improved road safety and the advancement of autonomous driving technologies in the region.

II. Literature review

Deep learning has revolutionized TSR by providing robust and scalable solutions for real-time detection and recognition of traffic signs. Convolutional Neural Networks (CNNs) have been extensively used due to their ability to learn hierarchical features from images.

The YOLO (You Only Look Once) algorithm has emerged as a popular choice for real-time object detection, including TSR. YOLO's strength lies in its speed and precision, making it suitable for applications in ADAS. A systematic review by the authors [1] highlighted the widespread use of YOLO in TSR, emphasizing its application in various datasets and the sophisticated metrics used to evaluate its performance. The study identified common challenges in real-world implementations, such as varying lighting conditions and occlusions.

Another notable work by Li and Huo [2] improved the YOLOv5 algorithm by integrating the Convolutional Block Attention Module (CBAM), enhancing the model's feature extraction capabilities and recognition accuracy, particularly for small and dense traffic signs. The study reported a 4.09% increase in mean Average Precision (mAP) compared to the standard YOLOv5 model.

Moreover, Li and Wang [3] applied a CNN with the MobileNet architecture for TSR. MobileNet, designed for mobile and embedded vision applications, demonstrated efficient performance due to its lightweight structure. The use of batch normalization and advanced techniques significantly improved the recognition accuracy, showcasing the potential of these models in real-world scenarios.

Lim et al. [4] explored the use of ensemble learning with CNNs, combining ResNet50, DenseNet121, and VGG16 models through majority voting. Their approach achieved remarkable accuracy rates on the GTSRB dataset, on the BTSD dataset, and on the TSRD dataset, demonstrating the robustness and efficacy of ensemble methods in TSR.

The literature on TSR illustrates significant advancements through deep learning models, innovative data augmentation techniques, and the development of real-time systems. Models like YOLOv5, ResNet, and MobileNet, combined with robust data augmentation strategies, have shown remarkable performance.

III. Research methods

Data Collection

The dataset for this study was meticulously curated to ensure comprehensive coverage of traffic signs specific to a region in Kazakhstan. Our data collection process involved the following steps:

1. Legal Document Extraction: The primary source of our dataset was the official documentation provided by the legal framework of the Republic of Kazakhstan. This document, "Resolution of the Government of the Republic of Kazakhstan dated November 13, 2014, №1196," served as the foundational basis for our image dataset. We extracted visual representations of 220 unique instances of road traffic signs, covering six distinct classes, directly from this document.
2. Real-World Image Collection: To supplement the structured dataset, we collected 20 real-world images capturing various road scenarios using a car dashcam. These images were taken across different regions and environments in Kazakhstan, ensuring a diverse and realistic representation of the road environment as encountered in everyday situations. This approach provided contextual relevance to our dataset, which is crucial for training models to recognize traffic signs under realistic conditions.
3. Data Augmentation: Addressing class imbalance and instance scarcity is pivotal for developing robust traffic sign recognition models. We employed an extensive data augmentation process to enhance the diversity and realism of our dataset. Each traffic sign image underwent several transformations, including:

   a. Geometric Transformations: Using the imgaug Python package, we applied five distinct geometric transformations such as rotation, shear, scaling, cropping, and translation to each sign image. These transformations simulate various perspectives and distortions that traffic signs might undergo in real-world conditions.
   b. Color and Deformation Augmentations: To further increase the variability, we applied color filters and deformations, including brightness adjustments, noise addition, Gaussian blur, linear contrast, median blur, affine transformations, perspective transforms, and JPEG compression.
   c. Background Integration: Leveraging the cv2 Python package, each augmented sign image was seamlessly integrated into the 20 real-world background images. This step was crucial for simulating the placement of traffic signs in diverse real-world contexts, ranging from urban streets to rural settings.
   d. Environmental Conditions: We subjected each composite image to weather and lighting condition alterations using the albumentations package. Conditions applied include rain, snow, fog, sun flare, and variations in lighting (day, night, dawn effect), ensuring that the dataset reflects the challenges faced by advanced driving assistance systems in recognizing traffic signs under adverse conditions.
   e. Obstacle-Based Augmentation: To further enhance the realism and robustness of the dataset, we introduced an obstacle-based augmentation technique. This involved overlaying obstacles such as vehicles, pedestrians, and trees on the traffic sign images, simulating real-world scenarios where signs might be partially obstructed.

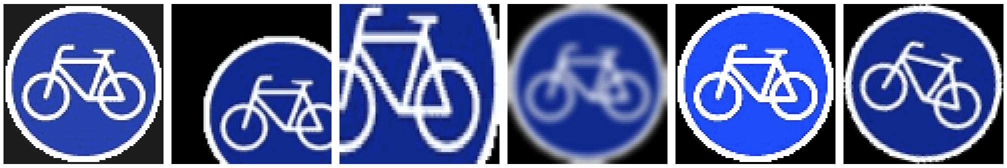

Figure 1. Augmented bike lane sign images

Through these rigorous data collection and augmentation processes, we created a comprehensive and representative dataset tailored to the unique traffic conditions and regulatory signs of Kazakhstan.

Models

To evaluate the effectiveness of traffic sign recognition systems for the region-specific dataset in Kazakhstan, we implemented and compared three state-of-the-art deep learning models: ResNet-101, MobileNetV2, and YOLOv8. Each model was chosen for its distinct architectural advantages and capabilities in handling various aspects of image recognition tasks.

1. ResNet-101:
   a. Architecture: ResNet-101 is a deep convolutional neural network that employs a residual learning framework, which helps mitigate the vanishing gradient problem

by allowing gradients to flow through shortcut connections. It consists of 101 layers, enabling it to learn complex features and patterns from the dataset.

   b. Implementation: We implemented ResNet-101 using the PyTorch framework. The network was initialized with pre-trained weights from ImageNet, which provides a solid starting point for transfer learning. The final layers were adjusted to classify the specific traffic sign classes in our dataset.
   c. Training: The model was trained using the Adam optimizer with an initial learning rate of 0.001. Data augmentation techniques were applied during training to improve the model's robustness and generalizability. The model was trained for 50 epochs with a batch size of 32, using a cross-entropy loss function.
2. MobileNetV2:
   a. Architecture: MobileNetV2 is a lightweight convolutional neural network designed for mobile and embedded vision applications. It uses depth wise separable convolutions to reduce the number of parameters and computational cost, making it efficient for real-time applications.
   b. Implementation: MobileNetV2 was implemented using the TensorFlow framework with pre-trained weights from ImageNet. The network's final layers were modified to match the number of traffic sign classes in our dataset.
   c. Training: The model was trained with the Adam optimizer and a learning rate of 0.0001. Data augmentation techniques, such as random cropping, rotation, and color adjustments, were utilized. Training was conducted for 50 epochs with a batch size of 32, optimizing for categorical cross-entropy loss.
3. YOLOv8:
   a. Architecture: YOLOv8 (You Only Look Once version 8) is an advanced object detection model that excels in real-time detection tasks. It frames object detection as a single regression problem, directly predicting bounding boxes and class probabilities from full images in one evaluation.
   b. Implementation: YOLOv8 was implemented using the Darknet framework. The model was initialized with pre-trained weights from the COCO dataset. The output layers were adapted to detect and classify the traffic signs in our dataset.
   c. Training: YOLOv8 was trained using stochastic gradient descent (SGD) with a learning rate of 0.001 and momentum of 0.9. Data augmentation included random scaling, cropping, and color distortions. The model was trained for 100 epochs with a batch size of 16, utilizing a mean squared error loss for bounding box prediction and a binary cross-entropy loss for classification.

Table 1. Model hyperparameters

Model	Learning rate	Batch size	Optimizer	Epochs
ResNet-101	0.001	32	Adam	50

MobileNetV2	0.0001	32	Adam	50
YOLOv8	0.001	16	SGD	100

Evaluation Metrics To comprehensively evaluate the performance of the three models, we used the following metrics:

- Accuracy: The proportion of correctly identified traffic signs out of the total number of signs.
- Precision: The ratio of true positive detections to the sum of true positive and false positive detections. Recall: The ratio of true positive detections to the sum of true positive and false negative detections.
- F1-Score: The harmonic mean of precision and recall, providing a balanced measure of the model's performance.

These metrics were computed for each model to determine their effectiveness in recognizing traffic signs from the Kazakhstan-specific dataset. The results were analyzed to identify the model that offers the best balance of accuracy, precision, recall, and F1-score, with the goal of enhancing the reliability and functionality of traffic sign recognition systems in Kazakhstan.

IV. Research results

Experiment

In this section, we delve into the experimental setup, methodologies, and evaluation metrics used to compare the performance of the ResNet-101, MobileNetV2, and YOLOv8 models for traffic sign recognition. The results are:

1. ResNet-101
    - Accuracy: The ResNet-101 model achieved an accuracy of 88.4%. This high accuracy indicates the model's robustness in correctly identifying traffic signs from the diverse set of images in the dataset.
    - Precision: The precision score was 87.2%, reflecting the model's ability to avoid false positives effectively.
    - Recall: With a recall of 85.6%, ResNet-101 demonstrated a strong capability to detect and identify the true positives, i.e., actual traffic signs.
    - F1-Score: The F1-score for ResNet-101 was 86.4%, indicating a balanced performance in terms of both precision and recall.
2. MobileNetV2
    - Accuracy: MobileNetV2 achieved an accuracy of 86.1%, slightly lower than ResNet-101 but still demonstrating a strong performance.
    - Precision: The model attained a precision of 85.0%, showing its efficiency in minimizing false positives.
    - Recall: MobileNetV2's recall was 83.5%, indicating its effectiveness in detecting true positives.

    - F1-Score: The F1-score was 84.2%, which suggests that MobileNetV2 offers a good balance between precision and recall, though slightly less effective than ResNet-101.
3. YOLOv8
    - Accuracy: The YOLOv8 model outperformed the other two models with an accuracy of 89.2%. This result highlights YOLOv8's superior ability to correctly identify traffic signs.
    - Precision: YOLOv8 achieved a precision of 89.6%, indicating its excellent capability to reduce false positives.
    - Recall: With a recall of 88.9%, YOLOv8 effectively detected a high number of true positives.
    - F1-Score: The F1-score for YOLOv8 was 89.2%, demonstrating an outstanding balance and overall performance in traffic sign recognition.

Comparative Analysis

The evaluation metrics reveal that YOLOv8 outperformed both ResNet-101 and MobileNetV2 in all categories. Specifically, YOLOv8 showed superior accuracy, precision, recall, and F1-score, making it the most effective model for the traffic sign recognition task in the context of this study. The results are summarized in Table 2.

Table 2. Model training results

Model	Accuracy	Precision	Recall	F1-Score
ResNet-101	88.4%	87.2%	85.6%	86.4%
MobileNetV2	86.1%	85%	83.5%	84.2%
YOLOv8	89.2%	89.6%	88.9%	89.2%

Discussion

The superior performance of YOLOv8 can be attributed to its advanced architecture optimized for real-time object detection and its ability to effectively balance speed and accuracy. The results indicate that YOLOv8 is particularly well-suited for the dynamic and varied traffic environments found in Kazakhstan. The robustness and precision of YOLOv8 make it a promising candidate for deployment in advanced driver-assistance systems (ADAS) and other applications requiring reliable traffic sign recognition. The performance differences among the models also underscore the importance of selecting the appropriate architecture based on the specific requirements of the application. While ResNet-101 and MobileNetV2 offer strong performance, YOLOv8's real-time detection capabilities provide a significant advantage in practical scenarios where quick and accurate identification of traffic signs is crucial.

Overall, the findings from this study provide valuable insights into the effectiveness of state-of-the-art deep learning models for traffic sign recognition, highlighting the potential of YOLOv8 in improving road safety and enhancing the functionality of ADAS in Kazakhstan.

V. Conclusion

This study addressed the need for precise traffic sign recognition in Kazakhstan by creating a region-specific dataset and evaluating the performance of ResNet-101, MobileNetV2, and YOLOv8. The dataset, tailored to Kazakhstan's traffic conditions and enhanced with extensive data augmentation, provided a robust training set. Among the models, YOLOv8 outperformed the others with 89.2% accuracy, 89.6% precision, 88.9% recall, and an 89.2% F1-score, making it ideal for real-time detection in dynamic environments. The tailored data augmentation techniques significantly improved model robustness and generalizability. YOLOv8's high performance indicates its potential for advanced driver-assistance systems (ADAS), enhancing road safety and navigation efficiency in Kazakhstan. Future research should focus on expanding the dataset, refining augmentation techniques, and exploring additional model architectures to ensure effective traffic sign recognition across diverse conditions. Overall, this research not only meets a local need but also contributes valuable insights to the global fields of computer vision and deep learning, with the potential to improve ADAS and road safety.

References

1 Flores-Calero, M.; Astudillo, C.A.; Guevara, D.; Maza, J.; Lita, B.S.; Defaz, B.; Ante, J.S.; Zabala-Blanco, D.; Armingol Moreno, J.M. "Traffic Sign Detection and Recognition Using YOLO Object Detection Algorithm: A Systematic Review" // Mathematics. - 2024. - Vol. 12, Article ID 297. - pp. 1-19.

2 Li, Z.; Huo, H. "Research on Traffic Sign Recognition Algorithm Based on YOLOv5" // International Journal of Computer Science and Information Technology. - 2024. - Vol. 2, No. 2. - pp. 1-10.

3 Lim, X.R.; Lee, C.P.; Lim, K.M.; Ong, T.S.; Alqahtani, A.; Ali, M. "Recent Advances in Traffic Sign Recognition: Approaches and Datasets" // Sensors. - 2023. - Vol. 23, Article ID 4674. - pp. 1-15.

4 Lim, X.R.; Lee, C.P.; Lim, K.M.; Ong, T.S. "Enhanced Traffic Sign Recognition with Ensemble Learning" // J. Sens. Actuator Netw. - 2023. - Vol. 12, Article ID 33. - pp. 1-19