

Article

Improving Road Safety with AI: Automated Detection of Signs and Surface Damage

Davide Merolla ¹, Vittorio Latorre ¹, Antonio Salis ^{2,*} and Gianluca Boanelli ²

¹ Dipartimento di Bioscienze e Territorio, Università degli Studi del Molise, 86090 Pesche, Italy; merolla.davide96@gmail.com (D.M.); vittorio.latorre@unimol.it (V.L.)

² Research & Innovation Division, Tiscali Italia S.p.A., 09122 Cagliari, Italy; gianluca.boanelli@tiscali.com

* Correspondence: antonio.salis@tiscali.com

Abstract: Public transportation plays a crucial role in our lives, and the road network is a vital component in the implementation of smart cities. Recent advancements in AI have enabled the development of advanced monitoring systems capable of detecting anomalies in road surfaces and road signs, which can lead to serious accidents. This paper presents an innovative approach to enhance road safety through the detection and classification of traffic signs and road surface damage using advanced deep learning techniques (CNN), achieving over 90% precision and accuracy in both detection and classification of traffic signs and road surface damage. This integrated approach supports proactive maintenance strategies, improving road safety and resource allocation for the Molise region and the city of Campobasso. The resulting system, developed as part of the CTE Molise research project funded by the Italian Minister of Economic Growth (MIMIT), leverages cutting-edge technologies such as cloud computing and High-Performance Computing with GPU utilization. It serves as a valuable tool for municipalities, for the quick detection of anomalies and the prompt organization of maintenance operations.

Keywords: road safety; traffic sign detection; road surface damage detection; smart cities; YOLO; convolutional neural network (CNN); predictive maintenance; High-Performance Computing (HPC); cloud computing



Academic Editors: Amin Ramezani, Mohsen Hadian, Behrouz Zolfaghari and Firooz Saghezchi

Received: 3 February 2025

Revised: 25 February 2025

Accepted: 26 February 2025

Published: 4 March 2025

Citation: Merolla, D.; Latorre, V.; Salis, A.; Boanelli, G. Improving Road Safety with AI: Automated Detection of Signs and Surface Damage.

Computers **2025**, *14*, 91. <https://doi.org/10.3390/computers14030091>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

According to the latest World Health Organization (WHO) report, public road networks are the lifeblood of modern societies, playing a crucial role in the transportation of goods and people, which is fundamental for trade, commerce, and tourism. They also enable easy access to jobs, education, healthcare, and social activities, both in urban and rural areas. However, ensuring road safety remains a critical challenge.

Harsh weather conditions can accelerate road degradation, and as traffic volume grows—including heavy traffic—frequent repairs become necessary. Any lapse in maintenance can lead to severe incidents, resulting in fatalities worldwide. The 2023 WHO report states that approximately 1.19 million people die each year due to road traffic crashes [1], highlighting the urgency of proactive and efficient road monitoring systems.

This paper presents a novel deep learning-based approach for real-time traffic sign detection and road damage assessment, designed to support a Road Management System for public municipalities. While previous studies have focused either on road condition monitoring or traffic sign detection separately, our method integrates both tasks into a unified AI-driven framework.

By leveraging state-of-the-art computer vision techniques, our approach aims to provide higher accuracy, real-time processing capabilities, and improved adaptability to various environmental conditions. Unlike traditional rule-based or sensor-heavy solutions, our system operates efficiently using standard cameras and cloud-based AI models, reducing costs and making large-scale deployment feasible.

This research is conducted as part of the Molise CTE research project, funded by the Italian Ministry of Economic Growth (MIMIT). The project aims to harness emerging technologies such as cloud computing, High-Performance Computing, Artificial Intelligence, and AR/VR to develop and demonstrate next-generation Smart City solutions.

The structure of this paper is as follows. Section 2 presents a comprehensive Literature Review, positioning our approach within existing research and highlighting key differentiators. Section 3 describes the proposed AI-powered road monitoring system, detailing its architecture, data pipeline, and cloud-based infrastructure. Section 4 outlines the computational experiments, providing a rigorous performance evaluation with relevant benchmarks. Section 5 discusses Challenges, Solutions, and Integration with Municipal Maintenance Applications, addressing potential limitations, their mitigations, and demonstrating the practical applicability of our solution. Section 6 presents ongoing developments, future improvements, and potential exploitation opportunities, before concluding the paper.

2. Literature Review

2.1. Introduction

The detection and classification of traffic signs and road damage are vital components of intelligent transportation systems. Recent advancements in deep learning—particularly using convolutional neural networks (CNNs) and YOLO (You Only Look Once) architecture—have markedly enhanced the accuracy and efficiency of these systems. Moreover, the integration of GPS data has opened new avenues for comprehensive road monitoring and predictive maintenance. This section reviews seminal and recent literature in traffic sign detection and classification, road damage detection, and GPS-based road condition monitoring.

2.2. Traffic Sign Detection Approaches

2.2.1. Early and Seminal Work

Early research laid the groundwork for automated traffic sign detection using traditional machine learning methods. A key paper by Maldonado-Bascon et al. employs color segmentation and Support Vector Machines (SVMs) to detect and recognize road signs, demonstrating robustness against transformations and occlusions [2]. Another seminal approach by Fang, Chen, and Fuh integrates neural networks for feature extraction and Kalman filters for tracking, enabling reliable performance under diverse environmental conditions [3].

2.2.2. YOLO Architecture and Enhancements

In recent years, the YOLO family of algorithms has become a popular choice for real-time object detection, including traffic signs. YOLOv4 was shown by Yang and Zhang (2020) to significantly improve detection accuracy over YOLOv3 for Chinese traffic signs [4]. Zhang (2023) similarly demonstrated that YOLOv3 outperformed R-CNN algorithms in both speed and accuracy [5].

Enhancements to YOLO have been introduced to address specific challenges:

- **Lightweight Models.** Sign-YOLO integrates the Coordinate Attention (CA) module and High-BiFPN to improve multi-scale semantic fusion, achieving significant gains in precision, recall, and speed on the CCTSDB2021 dataset [6].

- **Advanced Feature Extraction.** PVF-YOLO employs Omni-Dimensional Convolution (ODconv) and Large Kernel Attention (LKA) to further boost detection accuracy and speed [7].

2.3. Traffic Sign Classification

2.3.1. CNN-Based Classification

Once detected, traffic signs need to be accurately classified into specific categories. Convolutional neural networks (CNNs) have shown exceptional performance in this area. Ciresan et al. (2012) achieved state-of-the-art results on German traffic signs using CNNs [8]. More recently, models like TSR-YOLO embed advanced modules tailored to complex traffic scenarios, further improving classification metrics [9].

2.3.2. Traffic Sign Damage Classification

Relatively few papers address traffic sign damage classification. Trpkovic, Selmic, and Jevremovic (2021) employed a CNN to identify and classify damaged and vandalized traffic signs [10]. Acilo et al. (2018) used transfer learning with ResNet-50 to detect signs' compliance status and physical degradation, achieving high accuracy [11].

2.3.3. Generative AI for Synthetic Data

Generative Adversarial Networks (GANs), particularly DCGAN, have been used to address the imbalance between damaged and undamaged traffic signs in training datasets. By generating realistic synthetic images of damaged traffic signs, researchers can enhance model robustness and accuracy [12]. This approach not only balances the dataset but also improves performance in recognizing signs under diverse and challenging conditions.

2.4. Road Damage Detection

2.4.1. Public Datasets

Datasets such as Mapillary and the Road Damage Detection (RDD) dataset are widely used for training and evaluating models aimed at identifying potholes, cracks, and surface wear. Their comprehensive annotations make them valuable resources for developing robust road damage detection systems.

2.4.2. Deep Learning Methods

Deep CNNs and YOLO architectures have been particularly successful in road damage detection. Zhang et al. (2018) utilized a deep CNN to detect road cracks, achieving high precision and recall [13]. Maeda et al. (2018) applied YOLO to detect multiple types of road damage, demonstrating effectiveness in real-world settings [14].

2.5. Integration of GPS Data

Using GPS data enriches time series analysis with spatial context, enhancing road damage detection and monitoring.

- **Mobile Sensor Networks.** Strutu et al. (2013) proposed a mobile sensor network-based system incorporating 3D accelerometers, GPS, and video modules for road surface monitoring [15]. Perttunen et al. (2011) similarly demonstrated that accelerometers and GPS on mobile phones could effectively detect road surface anomalies [16].
- **Low-Cost Systems.** Tarun and Esther (2023) introduced a Raspberry Pi and GPS-based road sign detection system, showcasing high detection precision and efficient real-time operation [17].

By combining sensor data with GPS coordinates, researchers can identify patterns in road degradation related to specific routes or environmental conditions, enabling proactive maintenance strategies.

2.6. Additional Insights from Recent Advances

Lim et al. (2023) provide a comprehensive overview of state-of-the-art traffic sign recognition, categorizing research into conventional machine learning and deep learning approaches [18]. Key developments include the following:

- **Preprocessing and Feature Extraction.** Sophisticated techniques for data augmentation, color normalization, and feature selection help address variations in lighting, weather, and camera angles.
- **Model Generalization.** The use of diverse training datasets is crucial to ensuring that models remain robust under varying backgrounds and environmental conditions.
- **Predictive Maintenance.** Integrating sensor data, GPS, and advanced neural network models allows for the implementation of predictive maintenance strategies, preventing road and sign failures before they occur [19].

2.7. Summary

Significant advancements have been made in traffic sign detection, classification, and road damage detection using deep learning—particularly YOLO and CNN-based methods. Seminal works laid the foundation using machine learning and traditional feature extraction, while modern approaches leverage real-time detection and classification via YOLO variants. Damage-specific research, although still limited, suggests promising avenues for applying CNNs and Generative Adversarial Networks to handle imbalanced datasets. Furthermore, the integration of GPS data offers spatial context essential for predictive maintenance. Overall, these trends highlight the potential for intelligent transportation systems to substantially enhance road safety, maintenance, and driver assistance.

3. Methodology

3.1. Experimental Workflow and Datasets

Figure 1 illustrates the overall workflow for both road sign and road damage detection/classification tasks. We begin by gathering images (from the Mapillary Vistas and RDD 2022 datasets), then apply data quality checks and data augmentation to expand or balance these datasets. For road signs, we perform two steps: (1) detection using YOLOv8, and (2) classification into damaged or not damaged using a convolutional neural network (CNN) with attention mechanisms. For road damage (treated purely as anomalies), a single detection step via YOLOv8 is sufficient.

Mapillary Vistas

The Mapillary Vistas dataset is a large-scale, street-level imagery dataset designed for semantic segmentation and object detection in various conditions [20].

- Classes: 401.
- Images: 41,906.
- Size: 32.8 GB.
- Train/Validation Split: 80%/20%.

RDD 2022

The Road Damage Detection (RDD) 2022 dataset focuses on identifying and classifying road-surface damages such as cracks and potholes [21].

- Classes: 4.

- Images: 34,007.
- Size: 9.6 GB.
- Train/Validation Split: 80%/20%.

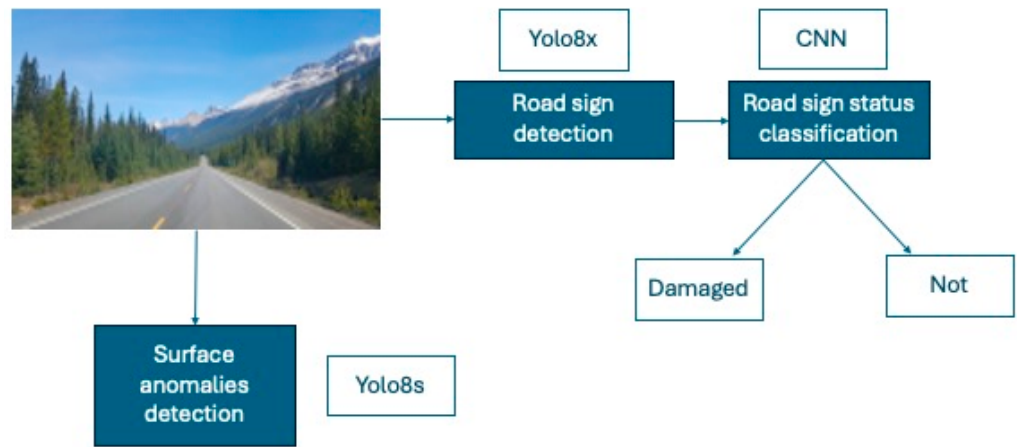


Figure 1. Experimental workflow. The diagram shows how images flow from collection through dataset expansion, data augmentation, and detection/classification.

3.2. YOLOv8 Architecture

Figure 2 illustrates the YOLOv8 network structure, which consists of three main components: Backbone, Neck, and Head. The Backbone (shown on the left side of the figure) is responsible for extracting hierarchical features from the input image through a series of convolutional layers and specialized blocks (e.g., C2f and Bottleneck). These blocks enhance the network’s learning capacity while maintaining low computational overhead.

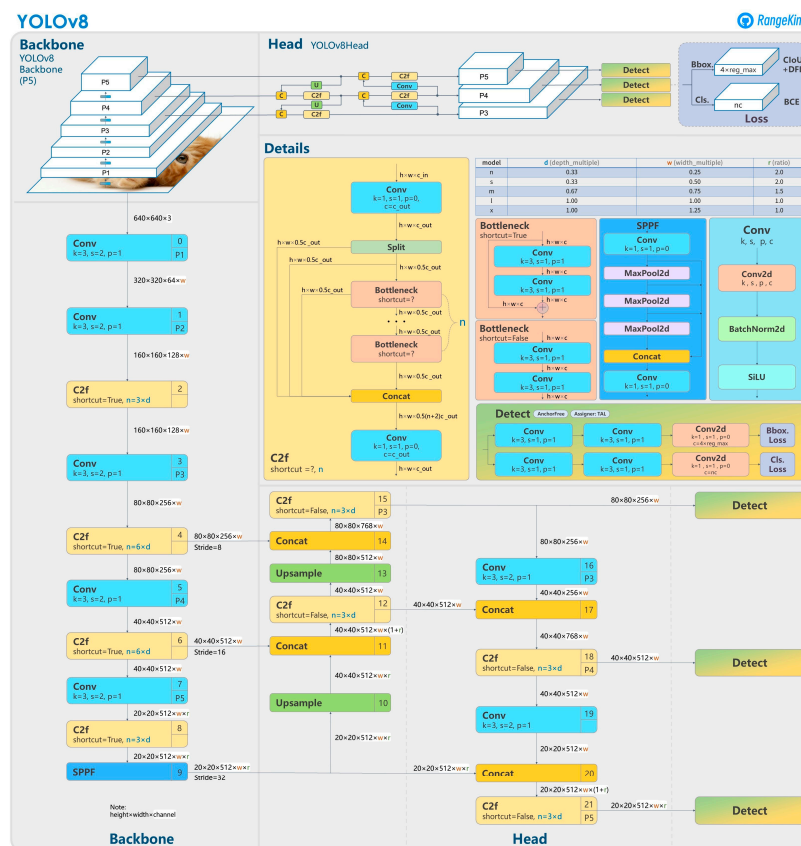


Figure 2. YOLOv8 architecture.

The middle section highlights additional key details of the architecture, including the following:

Split and Concat: Operations that split and concatenate feature maps, allowing multi-scale information to be merged efficiently.

C2f (Cross Stage Partial Networks v2) and Bottleneck: Blocks featuring shortcut connections that facilitate gradient flow and improve feature representation.

- **SPPF (Spatial Pyramid Pooling-Fast):** A module that partitions feature maps into regions of different sizes, enabling the network to capture multi-scale context.

The Neck (center-right) includes layers such as Upsample and Concat to fuse feature maps at various resolutions, improving multi-scale feature representation. Finally, on the right side is the Head, which processes detection outputs (Detect) across three different scales (P3, P4, P5). These convolutional layers further refine the fused features to produce final predictions for classes, bounding box coordinates, and confidence scores. The Loss function (top-right) combines classification, bounding box regression, and objectness terms to ensure balanced network training.

In our implementation, we used YOLOv8 “out of the box”, i.e., with no architectural modifications to the Ultralytics-released source code. We only customized certain training parameters—such as the number of epochs, batch size, and learning rate—to suit our dataset and experimental objectives. By adopting the native YOLOv8 framework, we leveraged its official optimizations, ensuring fast inference and competitive performance for both traffic sign detection and road damage assessment tasks.

3.3. First Phase: Detection

3.3.1. Data Preparation

Before training, data manipulation is performed to ensure optimal performance and accuracy:

- **Augmentation:** rotations, scaling, flips, color adjustments.
- **Normalization:** scaling pixel values (0–1).
- **Label Smoothing:** reducing overfitting by softening hard labels.
- **Anchor Box Calculation:** custom anchors to improve detection of varying object sizes.

3.3.2. YOLOv8s for Road Surface Damage

We use a smaller YOLOv8 variant (YOLOv8s) to balance speed and accuracy for road damage detection. Using a lightweight model is sufficient and efficient.

Parameter	Value
Pretrained	Yes (Ultralytics checkpoint)
Epochs	160
Image Size	640
Patience	100
Cache	RAM
Device	GPU
Batch Size	64

An example of road damage detection is shown in Figure 3.

YOLOv8s is used to identify anomalies (e.g., cracks, potholes) in the road surface.



Figure 3. Road damage detection.

3.3.3. YOLOv8x for Road Signs

For road sign detection, we employ the larger YOLOv8x model to achieve higher precision—this is critical for correctly identifying smaller signs with varied shapes. We prioritized accuracy over speed in this task since sign recognition demands finer resolution and often more complex feature extraction.

Parameter	Value
Pretrained	Yes (Ultralytics checkpoint)
Epochs	100
Image Size	640
Patience	100
Cache	RAM
Device	GPU
Batch Size	Auto

An example of traffic sign detection is shown in Figure 4.



Figure 4. Traffic sign detection.

YOLOv8x locates and classifies various signs in real time with high accuracy.

3.4. Second Phase: Road Sign Classification

After YOLOv8 detects road signs, we crop each sign from the frame and classify it as damaged or not damaged. Damaged signs include those with graffiti, stickers, rust, or physical deformations. The initial dataset is unbalanced (6025 damaged vs. 34,315 undamaged). We address this with the following:

- Focal Loss: weighs hard-to-classify examples more, mitigating class imbalance.
- Cutout Regularization: randomly removes sections of the image during training, improving robustness.

3.5. Enhanced CNN with Attention

In a subsequent step, we integrate attention mechanisms into a CNN to further improve classification accuracy:

1. Input Layer: $128 \times 128 \times 3$ images.
2. Convolutional Blocks: Each block has Convolution \rightarrow BatchNorm \rightarrow ReLU, plus Channel and Spatial Attention modules, followed by Max-Pooling.
3. Fully Connected Layers: Flatten \rightarrow Dense (256, ReLU) \rightarrow Dropout (0.5).
4. Output Layer: Dense (1, Sigmoid).
5. Optimizer and Loss: Adam with Focal Loss to handle class imbalance.

Attention Mechanisms

- Channel Attention: emphasizes relevant feature channels (e.g., small damages).
- Spatial Attention: focuses on crucial areas of the sign where anomalies might appear.

Data Augmentation and Regularization

- Augmentations: flips, rotations, shifts, shear, zoom.
- Cutout Regularization: random masking to handle occlusions.

Training and Evaluation

We train for 10 epochs (batch size 32, 80–20 train–validation split) and use ReduceLROnPlateau if validation accuracy stalls. The best model is saved via ModelCheckpoint, yielding 90% accuracy on validation data.

3.6. Generative AI for Class Balancing (Stable Diffusion)

We further address class imbalance using Stable Diffusion v2.1 (fine-tuned) to generate 18,000 synthetic images of damaged traffic signs:

Parameter	Value
Pretrained	Yes (Stable Diffusion v2.1)
Epochs	50
Image Size	512×512
Patience	10
Cache	RAM
Device	GPU
Batch Size	32
Conditioning Method	Image + Text Prompt Encoding
Optimization	AdamW (LR 5e-5, Cosine Scheduler)
Loss Functions	Contrastive + Perceptual (LPIPS)

By conditioning both on existing images and textual damage descriptions (e.g., “rust”, “graffiti”), we generate realistic variations of damaged signs, effectively tripling the size of the damaged class. This improves the CNN’s ability to recognize real-world damaged signs. Contrastive Loss ensures generated images differ significantly between damaged vs. undamaged categories, further aiding discrimination, while Perceptual Loss (LPIPS) helps maintain visual realism in synthetic images.

4. Computational Experiments

This section presents the experimental setup and results for our deep learning models applied to road sign detection (YOLOv8x), road surface damage detection (YOLOv8s), and road sign classification (CNN). We first describe the computational resources uti-

lized (Section 4.1), then detail the performance metrics and results for each model (Sections 4.2 and 4.3).

4.1. Hardware and Training Environment

4.1.1. YOLO Models on Google Colab

- GPU: NVIDIA Tesla T4.
- CUDA Cores: 2560.
- Tensor Cores: 320.
- GPU Memory: 16 GB GDDR6.
- Memory Bandwidth: 320 GB/s.
- Theoretical FP32 Performance: Up to 8.1 TFLOPS.
- CPU: Intel(R) Xeon(R) CPU, 2 vCPUs @ 2.3 GHz.
- RAM: 12.7 GB available in Colab.
- Disk: 100 GB available storage.

Google Colab's NVIDIA Tesla T4 GPU provided ample computational capacity for training both YOLOv8x (used for detecting road signs) and YOLOv8s (used for detecting road surface damages). The hardware configuration allowed for efficient handling of large datasets and the complex operations required by the YOLO architecture.

4.1.2. CNN on Revo Servers

- CPU: 24 vCPUs @ 2.5 GHz.
- RAM: 32 GB.

The CNN for road sign classification was trained on Revo servers. This setup featured more CPU cores and higher RAM, which supported the parallel data preprocessing and training steps necessary for our classification task.

4.2. YOLO Models: Performance Metrics and Results

We evaluated YOLOv8x (for road sign detection) and YOLOv8s (for road surface damage detection) using a set of common metrics:

- mAP50: Mean Average Precision at 50% IoU threshold.
- mAP50-95: Mean Average Precision averaged over IoU thresholds from 50% to 95%.
- Precision: The ratio of true positive detections to the total positive detections.
- Recall: The ratio of true positive detections to the total number of actual positives.
- Box Loss: Measures the error in bounding box predictions.
- Object Loss: Assesses the error in distinguishing objects from the background.

Each subsection below provides a closer look at how these metrics evolved during training, alongside commentary on the associated diagrams.

4.2.1. YOLOv8x for Road Sign Detection

1. Accuracy Metrics

Refer to Figure 5 for the training curves of mAP50, mAP50-95, precision, and recall:

- mAP50: Progressively increases and stabilizes around 0.9, reflecting high accuracy in detecting road signs at a 50% IoU threshold.
- mAP50-95: Improves more gradually, stabilizing near 0.7, demonstrating robustness across varying IoU thresholds.
- Precision: Fluctuates initially but trends upward, indicating fewer false positives over time.
- Recall: Increases steadily to about 0.8, reflecting the model's ability to detect actual positive instances effectively.

Interpretation: The consistently high mAP50 confirms the model's strong detection capabilities, while the gradual rise in mAP50-95 indicates good performance even with more stringent IoU thresholds.

2. Box Loss

Refer to Figure 6 for the box loss curve:

- The box loss decreases sharply during the initial epochs before stabilizing.
- Lower values imply more precise bounding box predictions.

Interpretation: As training progresses, YOLOv8x refines its bounding box coordinates, leading to improved localization of road signs.

3. Object Loss

Refer to Figure 7 for the object loss curve.

- Object loss drops rapidly in early epochs and then plateaus.
- Lower values indicate improved capability in distinguishing objects from the background.

Interpretation: YOLOv8x becomes increasingly effective at differentiating road signs from the surrounding environment, boosting overall detection performance.

4.2.2. YOLOv8s for Road Surface Damage Detection

1. Accuracy Metrics

Refer to Figure 8 for the training curves of mAP50, mAP50-95, precision, and recall.

- All metrics show steady improvement, indicating the model's growing proficiency in detecting road surface damages.

Interpretation: As with YOLOv8x, YOLOv8s demonstrates robust and consistent accuracy improvements, verifying its suitability for identifying damaged road surfaces.

2. Box Loss

Refer to Figure 9 for the box loss curve.

- The box loss decreases over time, highlighting enhanced precision in bounding box predictions for damaged regions.

Interpretation: The model learns to localize damaged sections more accurately with each epoch.

3. Object Loss

Refer to Figure 10 for the object loss curve.

- Like YOLOv8x, the object loss shows a downward trend, reflecting improved discrimination between damaged and undamaged surfaces.

Interpretation: YOLOv8s becomes more reliable at identifying genuine damage, reducing false positives on intact road surfaces.

4.3. CNN for Road Sign Classification

After detecting road signs, we employed a CNN to classify them into specific categories. Refer to Figure 11 for the evolution of training and validation metrics (accuracy, precision, and recall):

- Training Accuracy (blue line): Rises rapidly in the early epochs and converges at around 90%, indicating effective feature learning on the training set.
- Precision (green line): Remains consistently high, demonstrating the model's ability to correctly identify positive instances.

- Validation Metrics (orange, red, pink lines): Closely follow the training metrics, reflecting stable model performance and strong generalization.

Interpretation:

The CNN attains approximately 90% classification accuracy on the validation set, underscoring its robust performance in identifying various road sign types.

4.4. Figures and Their Descriptions

Below are placeholders for each figure referenced in the above subsections. Insert the actual images and descriptions once they are ready.

Figure 5 presents the accuracy metrics of the YOLOv8x model during training, evaluated on the validation set. The key metrics displayed include the following:

- mAP50 (dark blue line): This metric steadily increases, stabilizing around 0.9, indicating high accuracy in detecting road signs at a 50% IoU threshold.
- mAP50-95 (orange line): Shows a gradual increase over epochs, stabilizing near 0.7, demonstrating the model's robustness across varying IoU thresholds.
- Precision (cyan line): Initially fluctuates but trends upwards, suggesting improved true positive detections while reducing false positives.
- Recall (pink line): Shows a steady upward trend, reaching around 0.8, confirming the model's effectiveness in capturing actual positive instances.

Model accuracy measured on validation set

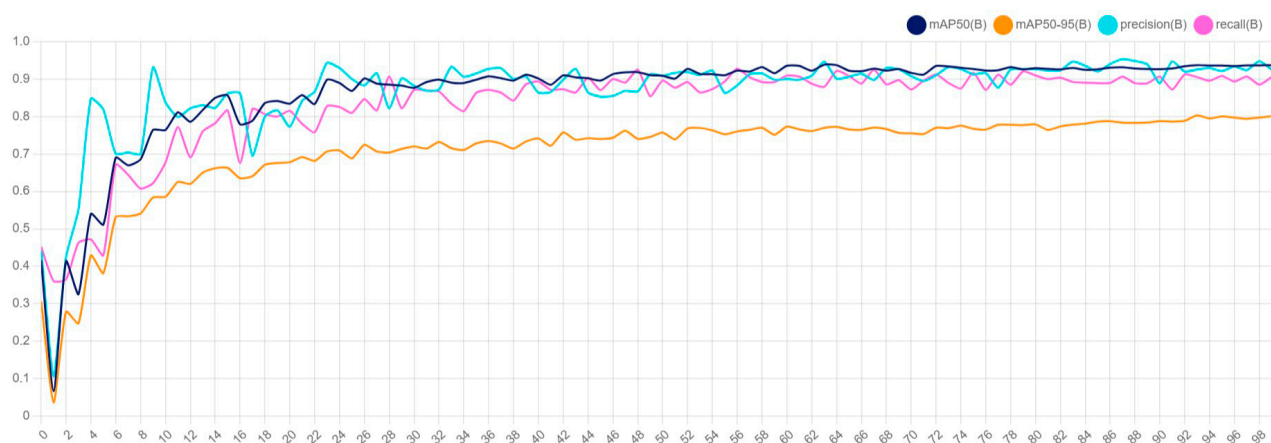


Figure 5. YOLOv8x accuracy.

The graph highlights how the YOLOv8x model progressively improves its accuracy across different IoU thresholds, making it a reliable solution for road sign detection.

Figure 6 illustrates the box loss values for the YOLOv8x model during training, showing the difference between predicted and actual bounding boxes for detected objects. The two curves represent the following:

- Training Box Loss (dark blue line): The loss decreases significantly within the initial epochs and continues to decline gradually, indicating that the model is learning to predict bounding box coordinates more accurately.
- Validation Box Loss (orange line): Initially higher, this value also decreases over time, though it stabilizes at a slightly higher level than the training loss.

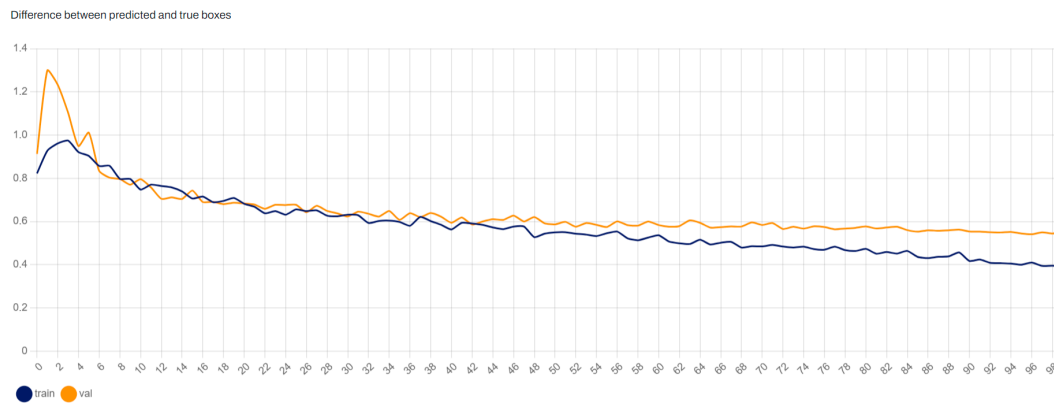


Figure 6. YOLOv8x box loss.

The declining trend in both training and validation box loss suggests that the model is effectively refining its bounding box predictions, leading to improved localization of road signs over successive training epochs.

Figure 7 presents the object loss values for the YOLOv8x model during training, which measures the model's ability to correctly distinguish between background and objects of interest. The two curves indicate the following:

- Training Object Loss (dark blue line): The loss decreases significantly in the early epochs and continues to decline steadily, demonstrating improved model confidence in object detection.
- Validation Object Loss (orange line): Initially higher, this loss follows a similar decreasing trend, though it stabilizes slightly above the training loss.

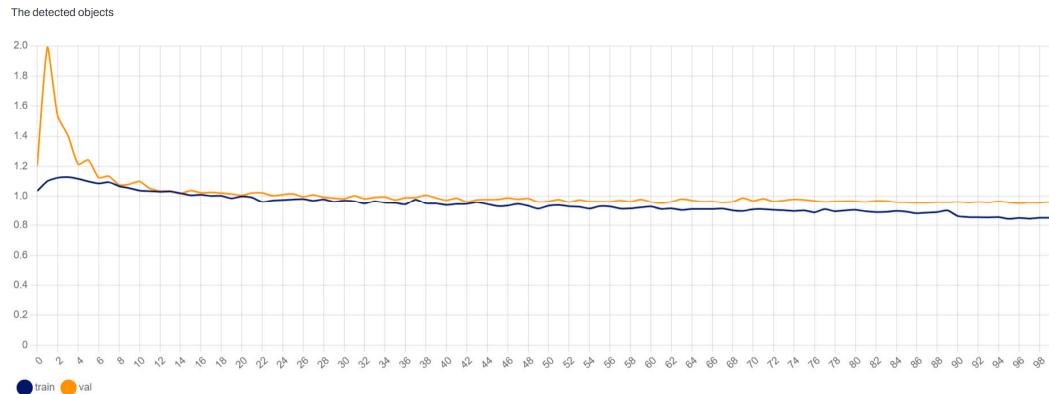


Figure 7. YOLOv8x object loss.

The overall downward trend in object loss confirms that YOLOv8x is learning to better differentiate road signs from the background, leading to more reliable and accurate detections over successive training epochs.

Figure 8 presents the accuracy metrics of the YOLOv8s model during training, evaluated on the validation set. The key metrics displayed include the following:

- mAP50 (dark blue line): This metric gradually increases and stabilizes around 0.65, indicating a reasonable level of accuracy in detecting road surface damages at a 50% IoU threshold.
- mAP50-95 (orange line): Exhibits a slower but steady improvement, stabilizing around 0.35, reflecting the model's performance across a broader range of IoU thresholds.
- Precision (cyan line): Shows fluctuations but follows an overall increasing trend, indicating progressive improvements in reducing false positives.

- Recall (pink line): Consistently trends upwards, stabilizing around 0.55, confirming the model's ability to detect most instances of road damage.

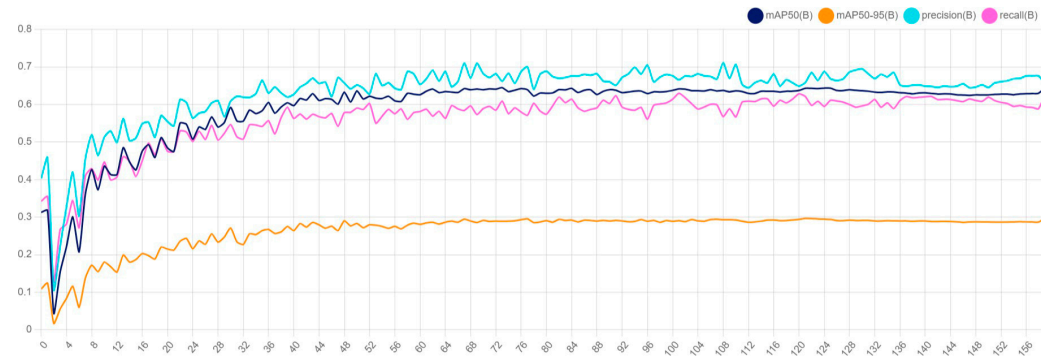


Figure 8. YOLOv8s accuracy.

The graph demonstrates that YOLOv8s effectively improves its accuracy over training epochs, successfully detecting road surface damages while adapting to different IoU thresholds.

Figure 9 illustrates the box loss evolution during training for the YOLOv8s model, which quantifies the accuracy of the model's bounding box predictions for detected road surface damages. The trends observed indicate a clear and steady improvement in localization precision:

- Training Box Loss (dark blue line): Shows a consistent downward trend, demonstrating that the model is progressively learning to refine its bounding box predictions and more accurately delineate damaged road areas.
- Validation Box Loss (orange line): Initially fluctuates but gradually stabilizes, remaining slightly higher than the training loss. This behavior suggests that while the model continues to improve, it maintains a balance between training and real-world generalization.

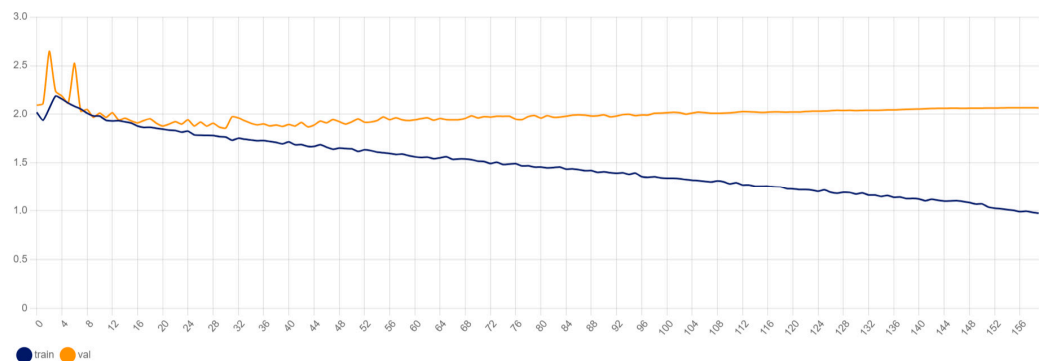


Figure 9. YOLOv8s box loss.

The overall reduction in box loss confirms that YOLOv8s effectively optimizes its bounding box localization, ensuring precise and reliable detection of road surface damages across different conditions.

Figure 10 illustrates the object loss progression during training for the YOLOv8s model, which measures how well the model differentiates between damaged and undamaged road surfaces. The trends observed indicate a significant improvement in the model's ability to correctly classify areas of interest:

- Training Object Loss (dark blue line): Displays a steady decline, indicating that the model is continuously refining its ability to distinguish road surface damages from the background.

- Validation Object Loss (orange line): While initially higher, it gradually stabilizes over time, suggesting that the model maintains its capability to generalize to unseen validation data.

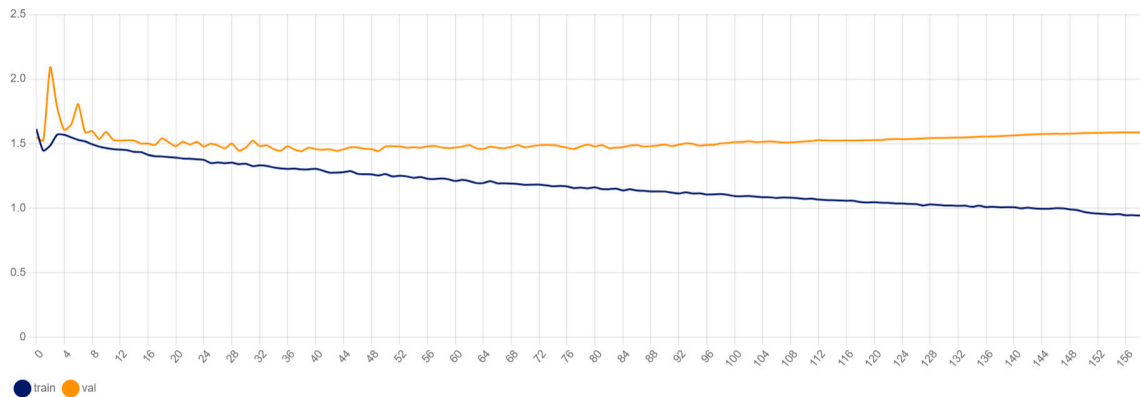


Figure 10. YOLOv8s object loss.

The overall decreasing trend in object loss confirms that YOLOv8s is becoming increasingly effective at detecting road damage, reducing classification errors, and enhancing detection reliability.

Figure 11 presents the evolution of key performance metrics for the CNN model during training, showcasing both training and validation results. The trends indicate high classification accuracy and stability across epochs:

- Training Accuracy (blue line): Rapidly increases and stabilizes around 90%, confirming the model's strong learning capability.
- Validation Accuracy (yellow line): Closely follows the training accuracy, indicating robust generalization on unseen data.
- Training Precision (green line): Maintains a high and stable value, demonstrating the model's effectiveness in correctly classifying positive instances.
- Validation Precision (red line): Shows some fluctuations but remains consistently high, reinforcing the model's reliability in classification.
- Training Recall (dark purple line): Approaches 1.0, ensuring minimal false negatives.
- Validation Recall (light purple line): Also remains high, further confirming the model's ability to correctly identify road sign classes.

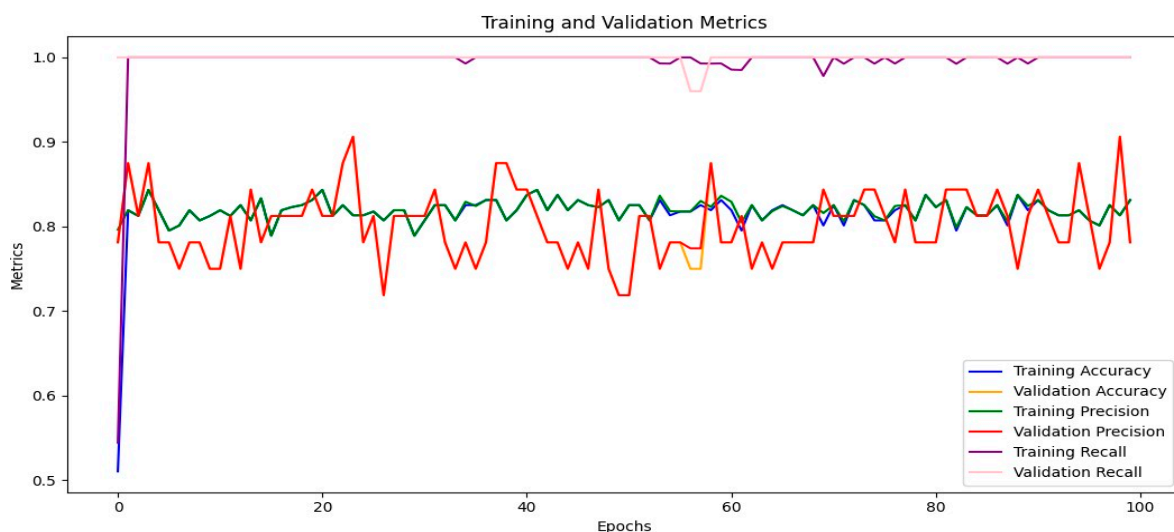


Figure 11. CNN training and validation metrics.

The figure highlights the CNN's strong classification capabilities, achieving approximately 90% accuracy while maintaining reliable precision and recall scores. This performance confirms the model's effectiveness in correctly classifying detected road signs.

4.5. Summary of Findings

YOLOv8x (Road Sign Detection): Achieves high mAP50 (~0.9), reflecting strong detection capabilities. Box loss and object loss trends confirm efficient learning for both localization and foreground/background separation.

YOLOv8s (Road Surface Damage Detection): Shows steady improvement in accuracy metrics, with reduced box and object loss over time, confirming its effectiveness in identifying damaged road surfaces.

CNN (Road Sign Classification): Reaches approximately 90% accuracy, indicating reliable classification of road signs once they are detected by the YOLO models.

Overall, the proposed models demonstrate promising performance for detecting and classifying road signs, as well as identifying road surface damage. Future work will focus on expanding the dataset, refining hyperparameters, and exploring advanced regularization techniques to further enhance performance.

Tables 1–3 clearly summarize the achieved results:

Table 1. Final performance metrics for YOLOv8x (road sign detection).

Metric	Final Value (Approx.)
mAP (50)	0.92
mAP (50–95)	0.78
Precision	0.88
Recall	0.85
Box Loss (Training)	0.30
Box Loss (Validation)	0.45
Object Loss (Training)	0.70
Object Loss (Validation)	0.90

Table 2. Final performance metrics for YOLOv8s (road surface damage detection).

Metric	Final Value (Approx.)
mAP (50)	0.75
mAP (50–95)	0.35
Precision	0.90
Recall	0.80
Box Loss (Training)	0.70
Box Loss (Validation)	1.80
Object Loss (Training)	0.80
Object Loss (Validation)	1.60

Table 3. Final performance metrics for CNN (road sign classification).

Metric	Training (Approx.)	Validation (Approx.)
Accuracy	0.95	0.88
Precision	0.85	0.80
Recall	1.00	0.90

5. Challenges, Solutions, and Integration with Municipal Maintenance Applications

In this section, we report the practical issues that were encountered during the design and implementation of our system and the implemented solutions to such issues. Then, we show how the trained model has been integrated with the tools available to the municipality.

The first issue in the application is the imbalance in the dataset, as most images are classified as “not damaged”. This was managed by implementing the Focal Loss function, which helps to down-weight the loss assigned to well-classified examples, thus focusing more on the difficult, minority class. We also applied CutOut data augmentation to enhance the robustness of the model.

Another issue with the classification of a sign to be damaged or not is that often it is required to identify subtle details such as small stickers or scratches. To address this, we incorporated attention mechanisms (Spatial and Channel Attention) within the CNN architecture, enabling the model to focus on regions of interest within each image.

Furthermore, we had to consider that the images in the dataset varied significantly in terms of lighting conditions, angles, and weather. To counteract this, we applied extensive data augmentation techniques such as rotations, shifts, and brightness variations. Additionally, we included a diverse set of images in the dataset to improve model generalization.

We also observed signs of overfitting in the model during training. To mitigate this, we employed Dropout layers and Batch Normalization, and implemented early stopping based on validation performance to prevent the model from memorizing the training data.

Finally, the training of such deep learning models on large datasets requires significant computational power. We leveraged High-Performance Computing (HPC) and cloud resources to accelerate the training process, making use of powerful GPUs and distributed computing.

Model Integration with the Municipal Maintenance Applications

The software has been developed so that it can be seamlessly integrated into a mobile application tailored for municipal maintenance operators. Through this application, maintenance teams can access real-time information on road and traffic sign anomalies detected by the system. The key feature of the application is the use of georeferencing to display layers of defective road surfaces and traffic signs on a GIS map, allowing for immediate visualization and action planning. Our approach combines top-down and bottom-up strategies. In addition to centralized monitoring, we encourage citizen reports, which contribute to the dataset of damaged traffic signs through a mobile application. These reports help balance the dataset classes and improve the accuracy and precision metrics of the convolutional neural network (CNN) through a continuous training process.

The application is designed with an intuitive user interface, enabling users to filter anomalies by type (e.g., potholes, damaged signs). As shown in Figure 12, the dashboard displays icons for different types of anomalies:

- Road Damage: Potholes, cracks, and other surface issues.
- Traffic Sign Damage: Defaced, rusty, or obstructed signs.

Each icon on the map provides detailed information about the anomaly, including its exact location, description, and images captured by the detection system.

Furthermore, the solution is designed to be easily replicable across municipalities of different sizes and can be implemented on smartphones. By leveraging cloud computing and scalable data storage, the system can manage large volumes of data and provide real-time updates to maintenance teams. This adaptability makes it suitable for application in any urban area, allowing municipalities to efficiently monitor and maintain their road infrastructure. Users can submit reports of damaged traffic signs, including GPS location

and an image of the sign. This information is used to update the dataset in real time and prioritize maintenance interventions, ensuring a timely and efficient response.

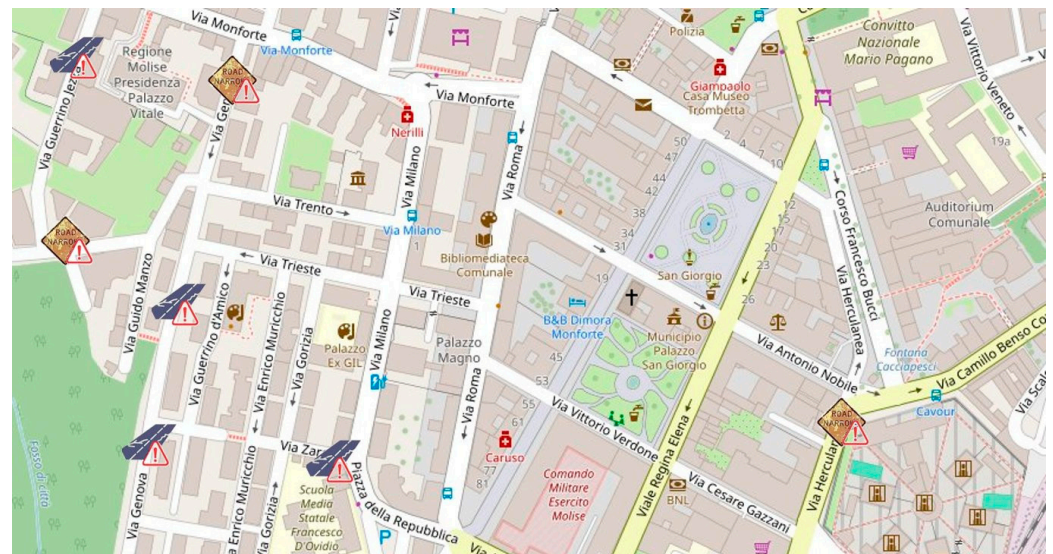


Figure 12. Dashboard showing georeferenced layers of road and traffic sign anomalies in the urban area of Campobasso, as explained by warning symbols.

The application is also designed to include additional features in the future, such as predictive maintenance algorithms, to analyze historical data to forecast potential road and traffic sign issues before they occur. This will enable municipalities to transition from reactive to proactive maintenance strategies, reducing costs and improving road safety.

6. Conclusions

In this study, we successfully developed and trained YOLO models for road sign detection and CNN models for classifying road signs as damaged or not damaged. Our approach utilized data augmentation and cutout regularization techniques to enhance the robustness and generalization of our models. Computational experiments conducted on Google Colab and Revo servers demonstrated the effectiveness of our methods in handling large datasets and complex computations.

For future work, we propose the following extensions to enhance the capabilities and applications of our models:

- **Incorporating Retroreflectivity Factors:** To further refine the classification of road signs, we plan to include retroreflectivity factors in our analysis. This involves detecting and classifying faded or discolored signs, which can significantly impact road safety. Developing models that can identify such signs will be crucial for timely maintenance and replacement.
- **Leveraging Generative AI for Data Labeling:** The process of manually labeling large datasets is time-consuming and prone to human error. By employing generative AI techniques, we can automate the labeling process, thereby reducing the time and effort required. This will also enable us to handle larger datasets more efficiently.

By implementing these extensions, we aim to improve the accuracy and reliability of road sign detection and classification systems. This will contribute to better road safety and maintenance practices, ultimately benefiting road users and maintenance authorities.

Author Contributions: Conceptualization, D.M. and A.S.; methodology, D.M., A.S. and V.L.; validation, D.M. and V.L.; formal analysis, D.M.; software, D.M.; writing—original draft preparation, D.M.,

A.S. and G.B.; writing—review and editing, D.M., A.S., V.L. and G.B.; supervision, A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Molise CTE Project, funded by MIMIT (FSC 2014–2020), grant #D33B22000060001.

Data Availability Statement: All relevant data are included in the paper. Further inquiries can be directed to the corresponding author.

Acknowledgments: We would like to extend our gratitude to the project coordinators and funding bodies for their support and resources, which were instrumental in the successful completion of this research.

Conflicts of Interest: Antonio Salis and Gianluca Boanelli are employees of Tiscali Italia. The other authors declare no conflicts of interest.

References

1. World Health Organization. Road Safety. 2023. Available online: <https://www.who.int/health-topics/road-safety> (accessed on 15 December 2024).
2. Maldonado-Bascón, S.; Lafuente-Arroyo, S.; Gil-Jiménez, P.; Gómez-Moreno, H.; López-Ferreras, F. Road-sign detection and recognition based on support vector machines. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 264–278. [[CrossRef](#)]
3. Fang, C.; Chen, S.-W.; Fuh, C. Road-sign detection and tracking. *IEEE Trans. Veh. Technol.* **2003**, *52*, 1329–1341. [[CrossRef](#)]
4. Yang, W.; Zhang, W. Real-time traffic signs detection based on YOLO network model. In Proceedings of the IEEE 2020 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Chongqing, China, 29–30 October 2020; pp. 354–357.
5. Zhang, X. Traffic sign detection based on YOLO v3. In Proceedings of the 2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA), Shenyang, China, 29–31 January 2023; pp. 1044–1048.
6. Song, W.; Suandi, S.A. Sign-YOLO: A novel lightweight detection model for Chinese traffic sign. *IEEE Access* **2023**, *11*, 113941–113951. [[CrossRef](#)]
7. Xu, T.; Ren, L.; Shi, T.; Gao, Y.; Ding, J.-B.; Jin, R.-C. Traffic sign detection algorithm based on improved YOLOX. *Inf. Technol. Control* **2023**, *52*, 966–983. [[CrossRef](#)]
8. Cirešan, D.C.; Meier, U.; Masci, J.; Schmidhuber, J. Multi-column deep neural networks for traffic sign classification. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3642–3649.
9. Song, W.; Suandi, S.A. TSR-YOLO: A Chinese traffic sign recognition algorithm for intelligent vehicles in complex scenes. *Sensors* **2023**, *23*, 749. [[CrossRef](#)] [[PubMed](#)]
10. Trpkovic, A.; Selmic, M.; Jevremovic, S. Model for the identification and classification of partially damaged and vandalized traffic signs. *KSCIE J. Civ. Eng.* **2021**, *25*, 3953–3965. [[CrossRef](#)]
11. Acilo, N.; Cruz, A.G.S.D.; Kaw, M.K.L.; Mabanta, M.D.; Pineda, V.G.G.; Roxas, E.A. Traffic sign integrity analysis using deep learning. In Proceedings of the 2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA), Penang, Malaysia, 9–10 March 2018; pp. 107–112.
12. Dewi, C.; Chen, R.-C.; Liu, Y.-T.; Tai, S.-K. Synthetic data generation using DCGAN for improved traffic sign recognition. *Neural Comput. Appl.* **2021**, *34*, 21465–21480. [[CrossRef](#)]
13. Zhang, A.; Wang, K.C.; Li, B.; Yang, E.; Dai, X.; Peng, Y.; Fei, Y.; Liu, Y.; Li, J.Q.; Chen, C. Automated pixel-level pavement crack detection on 3D asphalt surfaces using a deep-learning network. *Comput. -Aided Civ. Infrastruct. Eng.* **2017**, *32*, 805–819. [[CrossRef](#)]
14. Maeda, H.; Sekimoto, Y.; Seto, T.; Kashiya, T.; Omata, H. Road damage detection using deep neural networks with images captured through a smartphone. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; pp. 5207–5212.
15. Strutu, M.; Stamatescu, G.; Popescu, D. A mobile sensor network based road surface monitoring system. In Proceedings of the 2013 17th International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, Romania, 11–13 October 2013; pp. 630–634.
16. Perttunen, M.; Mazhelis, O.; Cong, F.; Kauppila, M.; Leppänen, T.; Kantola, J.; Collin, J.; Pirttikangas, S.; Haverinen, J.; Ristaniemi, T.; et al. Distributed road surface condition monitoring using mobile phones. *J. Ambient Intell. Smart Environ.* **2011**, *3*, 64–78.
17. Tarun, R.; Esther, B.P. Real-time regional road sign detection and identification using Raspberry Pi. In Proceedings of the 2023 International Conference on Networking and Communications (ICNWC), Chennai, India, 5–6 April 2023; pp. 1–5.
18. 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**, *23*, 4674. [[CrossRef](#)] [[PubMed](#)]

19. Pensa, D. Integration of GPS Data into Predictive Models for Tyre Maintenance. Master's Thesis, Politecnico di Milano, Milan, Italy, 2017. Available online: <https://www.politesi.polimi.it/retrieve/a81cb05c-7f41-616b-e053-1605fe0a889a/tesi.pdf> (accessed on 15 December 2024).
20. Neuhold, G.; Ollmann, T.; Rota Bulò, S.; Kotschieder, P. The Mapillary Vistas Dataset for semantic understanding of street scenes. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 4990–4999. [[CrossRef](#)]
21. Arya, D.; Maeda, H.; Ghosh, S.K. RDD2022: A multi-national image dataset for automatic road damage detection. *Geosci. Data J.* **2022**, arXiv:2209.08538. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.