Road Damages Detection Using YOLOv8

Farhan Faturahman
Telkom University
Bandung, 40257 Indonesia
1301213188

Gevin Latifan Abduldjalil
Telkom University
Bandung, 40257 Indonesia
1301213541

Rizal Hidayat Putra
Telkom University
Bandung, 40257 Indonesia
1301213444

Abstract

Road damage, such as cracks, potholes, and surface deformations, significantly impacts transportation safety and road maintenance costs. This project addresses these challenges using YOLOv8, a state-of-the-art object detection framework, to detect and classify various types of road damage. By employing the preprocessed Road Damages Detection v7 dataset with isolated objects, our model demonstrates high precision and recall, achieving a mean Average Precision (mAP50-95) score of 91.1%. The model was further integrated into a prototype web application, enabling users to upload isolated road damage images for detection. These results underscore the system's potential for real-world applications, improving road inspection efficiency and reducing long-term maintenance costs.

Keywords: *Road Damage Detection, YOLOv8, Object Detection, Deep Learning, Infrastructure Maintenance*

1. Introduction

Road infrastructure is a critical component of urban and rural mobility. Damage such as cracks, potholes, and surface deformations not only compromise safety but also escalate maintenance costs if left unaddressed. These issues contribute to traffic accidents, increased vehicle wear, and extended repair timelines. Manual inspections, the traditional approach, are resource-intensive, slow, and prone to human error. These limitations call for an automated system that is both accurate and efficient [1][2].

The rapid advancements in computer vision and deep learning have paved the way for automated damage detection systems. However, many existing solutions struggle with inconsistent datasets, high computational costs, and limited scalability. This project addresses these challenges by employing YOLOv8, a Convolutional Neural Network (CNN)-based object detection framework. By leveraging its real-time processing capabilities and the preprocessed Road Damages Detection v7 dataset, the project aims to develop a robust solution that can:

- Automate road damage detection with high precision and recall.
- Improve the efficiency of infrastructure inspections.
- Lay the groundwork for integrating automated solutions into road maintenance workflows.

YOLOv8 was chosen over alternative models such as Faster R-CNN and SSD due to its superior trade-off between speed and accuracy. While Faster R-CNN offers high precision, it suffers from slower inference times, making it less suitable for real-time applications. On the other hand, SSD, though faster, often underperforms in detecting smaller objects, which is critical in road damage detection. YOLOv8's lightweight architecture and efficient feature extraction make it the optimal choice for this project, particularly when paired with preprocessed datasets that mitigate noise and enhance detection accuracy. This decision aligns with findings from prior studies highlighting YOLO's capability to handle high-variance datasets effectively [3][4].

The preprocessed Road Damages Detection v7 dataset played a pivotal role in ensuring high detection performance by addressing class imbalances and noise. The dataset's isolated object preprocessing technique allowed YOLOv8 to focus on the damage features, improving learning efficiency and detection accuracy. Such preprocessing aligns with findings from [3][4], which highlight the importance of dataset quality in scalable and efficient detection systems.

This paper provides an in-depth analysis of the methodologies, experiments, and results, demonstrating the practical implications of the proposed system. The findings underscore the potential for deploying such technologies to enhance transportation safety and reduce long-term maintenance costs.

2. Related Work

Road damage detection has gained traction in recent years due to advancements in deep learning. Notable contributions include:

- **Arya et al. [1][2]**: Developed global road damage detection systems with annotated datasets that addressed variability in road conditions across regions.

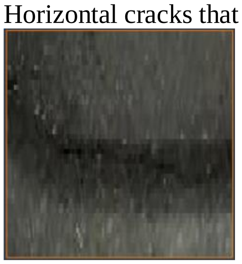
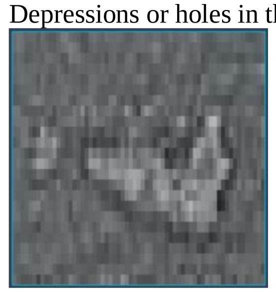
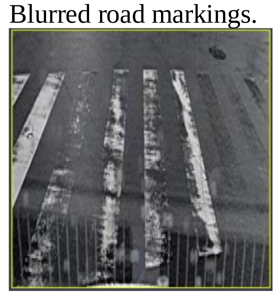
- **Naddaf-Sh et al. [3]**: Emphasized efficient and scalable approaches to road damage detection, underscoring the need for high-quality datasets and lightweight models for real-world applications.
- **Angulo et al. [4]**: Introduced scalable data acquisition methods tailored for real-world deployment, focusing on image annotation and sensor integration.
- **Rateke et al. [5]**: Explored damage detection under varied environmental conditions, highlighting challenges in lighting and weather.
- **Arya et al. [6]**: Presented RDD2020, a comprehensive annotated dataset that facilitated significant advancements in road damage detection by improving data consistency and coverage.

Despite these efforts, there remains a gap in achieving high performance across diverse datasets. The Road Damages Detection v7 dataset, with its focus on isolated objects, represents a significant improvement in addressing class imbalance and noise in training data. By integrating this dataset with YOLOv8, our approach builds on existing research while introducing enhancements in precision, recall, and computational efficiency [3][4].

3. Data

Dataset Overview

The Road Damages Detection v7 dataset, sourced from Roboflow Universe [7], includes 3,506 annotated images across seven damage classes:

1. **D00: Longitudinal Crack**
   Linear cracks along the road surface.
   
   Figure 1 Sample Data D00
2. **D10: Transverse Crack**
   Horizontal cracks that often cross the road.
   
   Figure 2 Sample Data D10
3. **D20: Aligator Crack**
   Interconnected cracks resembling alligator skin.
   
   Figure 3 Sample Data D20
4. **D40: Pothole**
   Depressions or holes in the road surface.
   
   Figure 4 Sample Data D40
5. **D43: White Line Blur**
   Blurred road markings.
   
   Figure 5 Sample Data D43
6. **D44: Cross Walk Blur**
   Faded pedestrian crossings.
   
   Figure 6 Sample Data D44
7. **D50: Manhole Cover (TBC)**
   Circular or rectangular covers embedded in the road.

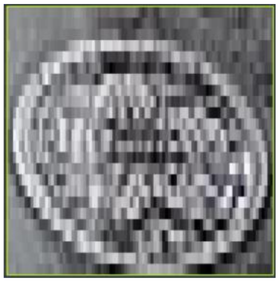


Figure 7 Sample Data D50

Preprocessing Techniques

To enhance the dataset's usability for YOLOv8, preprocessing included:

- **Object Isolation:** Each damage instance was isolated using bounding boxes, effectively removing irrelevant background noise.
- **Normalization:** Image dimensions were standardized to 640x640 pixels, ensuring compatibility with the model's architecture.

These preprocessing steps significantly improved feature extraction, reducing model training time and enhancing detection accuracy. The alignment of preprocessing methods with recommendations from [3][4] highlights their critical role in achieving scalable and robust performance.

Evaluation Metrics

The evaluation relied on:

- **Precision:** The proportion of true positives among predicted positives.
- **Recall:** The proportion of true positives among actual positives.
- **F1-score:** The harmonic mean of precision and recall.
- **Mean Average Precision (mAP50-95):** A comprehensive metric evaluating performance across multiple IoU thresholds.

4. Methods

YOLOv8 Architecture

YOLOv8 Nano, a lightweight yet powerful variant, was chosen for its resource efficiency and high performance. Key components include:

- **Conv Layers:** Responsible for feature extraction from input images.
- **C2f Blocks:** Optimize memory usage and computation.
- **SPPF:** Pooling layers that enhance multi-scale feature detection.
- **Detect Heads:** Generate bounding boxes, confidence scores, and class predictions.

Training Configuration

The model training utilized:

- **Pre-trained Weights:** Initialized with weights from the COCO dataset to accelerate convergence.
- **Batch Size:** 16 images per training iteration, optimizing memory usage.
- **Epochs:** 50 iterations to ensure comprehensive learning.
- **Optimizer:** AdamW with automated learning rate adjustments for consistent updates.
- **IoU Threshold:** Set to 0.7 to refine bounding box predictions.
- **Early Stopping:** Triggered if no significant improvement occurred within 100 epochs.

These configurations were designed to balance performance and computational efficiency. Insights from [3] were instrumental in fine-tuning these parameters, ensuring alignment with best practices for scalable model training.

5. Experiments

Training Results

- **Duration**: 8 hours
- **Metrics**:
  - Precision: 84.4%
  - Recall: 88.7%
  - mAP50: 92.6%
  - mAP50-95: 91.1%

The training phase showed consistent improvement across epochs. The high mAP values indicate effective feature learning, which can be attributed to the isolated object preprocessing and the robust architecture of YOLOv8 Nano.

Validation Results

- **Metrics**:
  - Precision: 84.4%
  - Recall: 88.7%
  - mAP50: 92.6%
  - mAP50-95: 91.1%

Validation results mirrored the training phase, demonstrating minimal overfitting. The alignment between training and validation performance is indicative of the model's generalization capabilities.

Testing Results

- **Metrics**:
  - Precision: 89.1%
  - Recall: 86.7%
  - mAP50: 94.4%
  - mAP50-95: 92.9%

The testing phase highlighted the model's robustness in unseen data, particularly with high precision and mAP scores. The slightly reduced recall compared to training and validation suggests some missed detections, which may be due to challenging examples in the test set, such as faded road markings or heavily blurred features.

Analysis of Results

The slight variations in precision, recall, and mAP across training, validation, and testing can be attributed to the following factors:

- **Dataset Diversity:** The test set included more complex examples, challenging the model's detection capabilities.
- **Class Imbalance:** Certain classes, such as D40 (Small Cracks), had fewer examples, impacting performance.
- **Preprocessing Impact:** Object isolation significantly reduced noise, but some subtle features in the test set may have been overlooked.

The model's strong performance on classes like D10 (Transverse Cracks) and D43 (White Line Blur) demonstrates its efficacy in detecting well-defined damage types. However, further refinement is needed for underrepresented classes.

Prototype Web Application

Figure 8 Prototype Web App UI

The trained YOLOv8 model was successfully integrated into a web application prototype. The application allows users to upload images of isolated road damage, with the system providing detection and classification results in real-time. This implementation demonstrates the model's practical applicability and scalability.

Visualizations

The visualizations below provide a detailed representation of the model's performance and the progression of training and validation processes:

1. **Training Loss Curve:** This graph illustrates the reduction in loss over the training epochs, highlighting the model's learning efficiency.
2. **Precision-Recall Curve:** Depicts the trade-off between precision and recall at various confidence thresholds, indicating the model's overall balance between accuracy and sensitivity.
3. **F1 Score Curve:** Shows the F1 scores across different thresholds, representing the harmonic mean of precision and recall.
4. **Confusion Matrix:** Provides insights into the classification performance for each damage class, identifying areas of strength and misclassification.

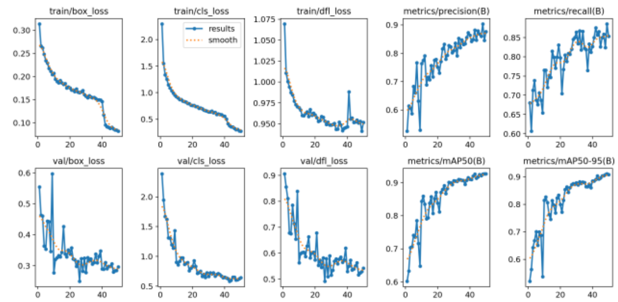


Figure 9 Training and Validation Results

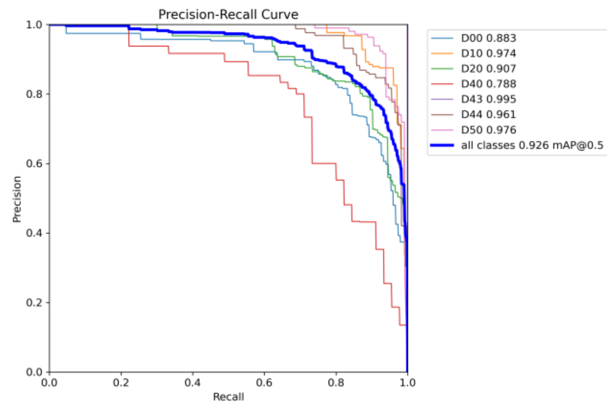


Figure 10 Precision-Recall Curve Training and Validation

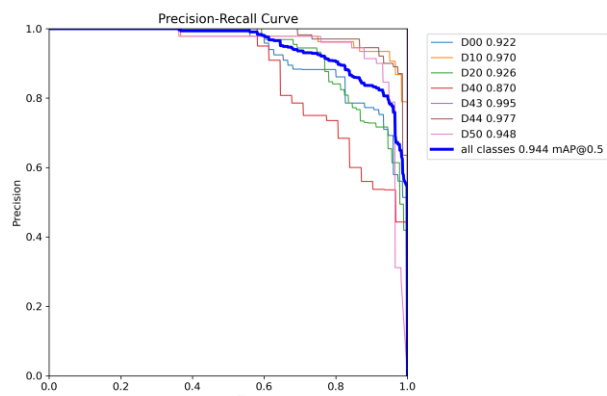


Figure 11 Precision-Recall Curve Testing

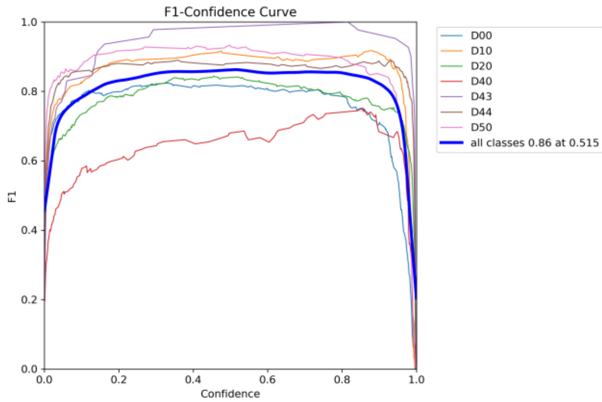


Figure 12 F1 Score Curve Training and Validation

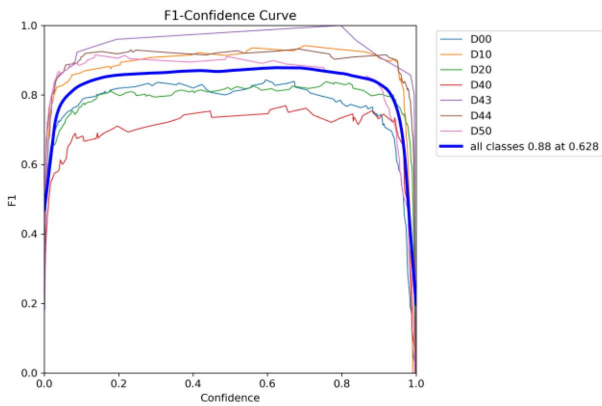


Figure 13 F1 Score Curve Testing

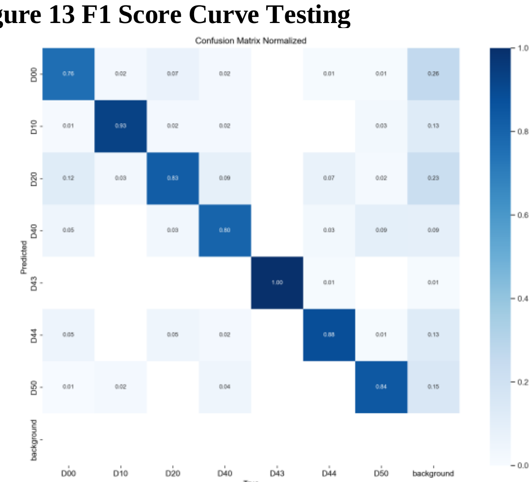


Figure 14 Confusion Matrix Normalized Training and Testing

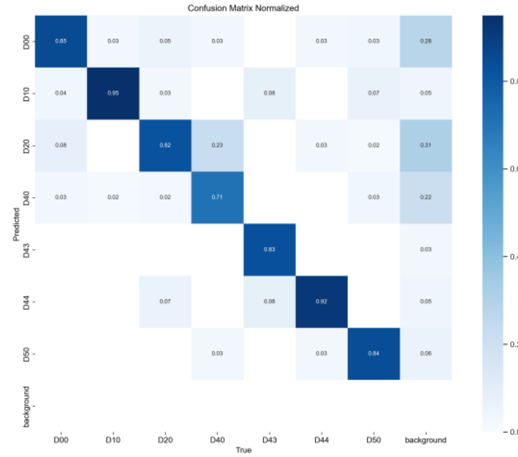


Figure 15 Confusion Matrix Normalized Testing

6. Conclusion

This project successfully demonstrates the applicability of YOLOv8 for real-time road damage detection. Preprocessing, particularly object isolation, was pivotal in zenhancing the model's precision and recall. The system achieved an impressive mAP50-95 of 91.1%, underscoring its effectiveness in real-world scenarios.

While the results are promising, challenges remain for certain damage classes, such as Small Cracks (D40). Future work may explore:

- **Advanced Augmentation Techniques:** To address class imbalance and improve generalization.
- **Class-weighted Loss Functions:** To enhance learning for underrepresented classes.
- **Broader Datasets:** To validate the model's robustness across diverse road conditions.

Additionally, the integration of the model into a prototype web application marks a significant step toward practical deployment. This implementation enables authorities and maintenance teams to streamline road inspections, significantly reducing manual labor and associated costs. Moreover, the system's scalability highlights its potential to be adapted for other infrastructure-related tasks, such as bridge inspection and railway maintenance.

The findings reinforce the growing importance of automated damage detection systems in urban planning and transportation management. By leveraging YOLOv8's capabilities and the preprocessed dataset, this project offers a scalable, cost-effective, and high-performing solution for road maintenance. Integration into mobile or web platforms could further amplify its utility, contributing to safer and more efficient transportation infrastructure worldwide.

References

[1] Arya, D., et al. *Global road damage detection: State-of-the-art solutions*. IEEE, 2020.

[2] Arya, D., et al. *Deep learning-based road damage detection*. Automation in Construction, 2021.

[3] Naddaf-Sh, S., et al. *An efficient and scalable deep learning approach for road damage detection*. IEEE, 2020.

[4] Angulo, A., et al. *Road damage detection acquisition system*. Springer, 2019.

[5] Rateke, T., et al. *Road surface detection and differentiation considering surface damages*. Autonomous Robots, 2021.

[6] Arya, D., et al. *RDD2020: Annotated image dataset for road damage detection*. Data in Brief, 2021.

[7] Tam, E. *Road Damages Detection Dataset*. Roboflow Universe, 2023.