Detecting Road Damage from Aerial Imagery

Goda Sreya Mamidala
gmamidala@islander.tamucc.edu
Texas A & M University – Corpus Christi
Corpus Chrsiti, Texas, USA

Jobit Jose
jjose@islander.tamucc.edu
Texas A & M University – Corpus Christi
Corpus Chrsiti, Texas, USA

Lakshmi Siva Kanth Reddy Kondamadugula*
lkondamadugula@islander.tamucc.edu
Texas A & M University – Corpus Christi
Corpus Chrsiti, Texas, USA

Abstract

Road infrastructure is a component of transport systems, and maintenance is required to ensure safety and prevent repair costs. The conventional process of road inspection is time-consuming, laborious, and prone to human mistakes. Aerial photography and deep learning have revolutionized automatic road damage detection, enabling the bulk monitoring of infrastructure with high accuracy. This research discusses how object detection frameworks such as YOLO (You Only Look Once), Faster R-CNN, and EfficientDet can be used to classify and detect road defects such as potholes, cracks, and surface deterioration using aerial imagery and public databases.

We intend to create a very efficient and user-friendly road damage inspection system founded on drone-captured image data, combined with publicly available road damage images from the internet. The project leverages state-of-the-art deep learning architectures trained with high-resolution aerial images that enable precise detection and classification of road defects.The model's performance is analyzed using performance metrics such as mean Average Precision (mAP), Intersection over Union (IoU), precision, recall, and F1-score, to ascertain both reliability and accuracy at high levels. The outcome of this research is anticipated to play a part in the development of automatic road maintenance techniques, thus reducing operational costs while promoting transportation safety. Future tasks include enhancing real-time inference, fusing multi-modal datasets, and deploying the trained model in real-life applications using edge computing or cloud-based APIs.

Keywords: Road Damage Detection, UAV Imagery,Deep Learning, Object Detection, YOLO, Faster R-CNN, Infrastructure Monitoring, Automated Road Inspection

1 Introduction

1.1 Motivation

The fast development of urban infrastructure has caused the increasing demand for effective road maintenance systems. Conventional camera-based and survey-based road inspection techniques mounted on vehicles are time-consuming, labor-intensive, and susceptible to errors. Road cracks and potholes need to be detected ahead of time to prevent unsafe road conditions, vehicle deterioration, and accidents. Thus, an automatic and scalable solution is necessary to provide timely and economic road maintenance.

Current development in deep learning and computer vision has made it possible to automatically detect road problems using new strategies. Road faults can be detected and identified effortlessly using deep learning software from high-resolution aerial images taken by UAVs (drones) and open datasets for real-time inspection of road conditions. Automation replaces human inspection, enhances the accuracy of road defect identification, and allows municipalities and road authorities to take proactive measures in maintaining road quality.

1.2 Problem Statement

Road infrastructure maintenance is essential in guaranteeing public safety and facilitating efficient transportation. Yet, conventional road damage detection methods are discovered to be ineffective and lack scalability over extensive geographical locations. The key problems connected with conventional road inspection methods are:

- **Inefficiency and high labor cost** – Manual surveys are human resource and time consuming.
- **Limited coverage** – Physical checks cannot be scaled appropriately across whole cities or countries.
- **Variable results** – Human decisions are fallible and subjective.
- **Slow response time** – Delay in the detection and repair of damage can lead to accidents and higher maintenance cost.

Utilizing aerial images and deep learning-based object detection models, this project is intended to present a scalable, dependable, and automatic approach to road damage detection.

1.3 Objectives

The overall objective of the project is to develop a deep learning-based road damage detection system from aerial images taken by UAVs and existing datasets. The specific objectives are:

- **Dataset Collection & Preprocessing** – Delays in identifying and repairing damage can lead to accidents and higher maintenance costs.

*Every author contributed equally to this project.

- **Model Selection & Implementation** – Select and utilize state-of-the-art object detection models such as YOLO, Faster R-CNN, and EfficientDet to detect various types of road damage.
- **Training and Testing** –The datasets with labels will be utilized for training the deep learning models. Their performance will be checked through precision, recall, F1-score, and mean Average Precision (mAP).
- **Optimization & Real-Time Deployment** – Optimize the model to run faster and have it deployable in the cloud or at the edge for field deployment.
- **Scalability & Future Enhancements** – Research making the model more feature rich, including the use of LiDAR information, transformer-based models, and enhancing the model for different road conditions.

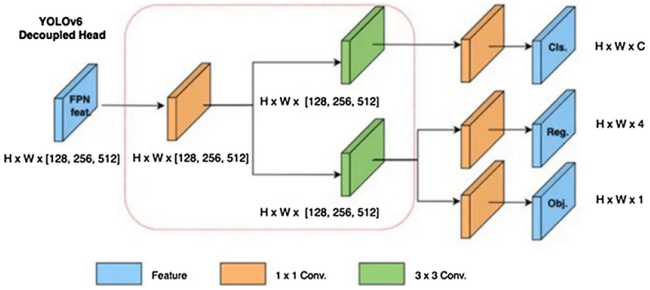

Figure 1. Schematic Representation of YOLO Architecture

YOLO (You Only Look Once) is a real-time deep learning-based object detection system that receives an entire image as input in one pass of a convolutional neural network (CNN), and is very fast for real-time usage. It is a grid-based system, which divides the image into an S × S grid, where every cell predicts a certain number of bounding boxes, object confidence scores, and class probabilities. Its design consists of convolutional layers for feature extraction and fully connected layers for end prediction. However, the accuracy and multi-scale detection were improved in later editions of YOLO, and it is now heavily utilized in autonomous vehicles, surveillance systems, and real-time video processing.

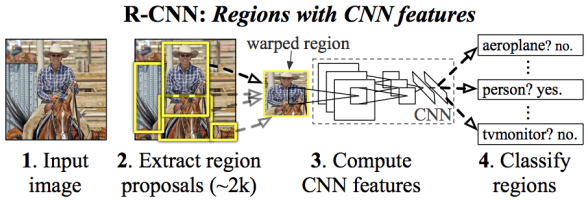

Figure 2. Schematic Representation of R-CNN Architecture

The Region-Based Convolutional Neural Network (R-CNN) is an object detection model that utilizes a two-stage method in region proposal and classification. The R-CNN first generates region proposals using the application of Selective Search before continuing to utilize a CNN for feature extraction. These features are then passed to fully connected layers for object classification and bounding box regression. R-CNN is very accurate but computationally expensive with processing of regions individually. Later implementations like Fast R-CNN and Faster R-CNN made efficiency improvements with the inclusion of ROI pooling and Region Proposal Networks (RPNs).

1.4 Research Questions

This study aims to address the following research questions:

- How well do deep learning object detection algorithms detect road damage on aerial images?
- What object detection model (YOLO, Faster R-CNN, EfficientDet) is best for both accuracy and speed in real-time road damage detection?
- How does diversity in datasets (UAV images and satellite images) influence model performance?
- What are the main obstacles to using road damage detection models in practical applications?

1.5 Significance of the Study

The suggested system will dramatically improve the manner and degree to which road maintenance is carried out. The main contributions of this research include:

- **Using machines to find road damage** – Less number of individuals to inspect and faster work.
- **Scalability** – Enabling large-scale monitoring of road conditions through UAV and satellite imagery.
- **Real-Time Analysis** – Providing actionable intelligence to road maintenance authorities for prompt repairs.
- **Cost-effectiveness** – This involves reducing labor costs and improving the scheduling of road maintenance.

This study aims to address these issues. It seeks to assist in the development of intelligent systems for infrastructure management. This will improve road safety and reduce maintenance expenses for governments and private organizations.

1.6 Pros and Cons of Existing Methods

Existing road damage detection methods have made significant progress, but they still have notable limitations. Traditional approaches rely on manual inspections or rule-based image processing techniques, which are labor-intensive, slow, and prone to human error. Some early deep learning-based approaches, such as Faster R-CNN and YOLOv5, provide high accuracy in object detection but often struggle with real-time performance due to their computational complexity. Faster R-CNN, for instance, delivers excellent precision

but suffers from slow inference speeds, making it less suitable for real-time UAV-based applications. On the other hand, YOLO-based models such as YOLOv5 and YOLOv7 improve detection speed but may have limitations in detecting fine-grained damage types in complex road conditions. Additionally, CNN-based models lack contextual understanding, making them less effective in identifying small or irregularly shaped road defects.

Our approach integrates YOLOv7, Faster R-CNN, and Transformer based object detection models to achieve a balance between accuracy, real-time performance, and contextual understanding. Unlike traditional CNN-based approaches, Transformer-based models enhance feature learning through self-attention mechanisms, improving the detection of small-scale damage patterns and complex road textures. Furthermore, we use a multi-source dataset strategy, incorporating UAV-captured images and publicly available datasets (RDD2022, CRDDC2022) to improve the adaptability of the model to different road conditions. By leveraging multi-source data fusion, we improve generalization and ensure that the model performs well in varying lighting, weather, and terrain conditions. Furthermore, our approach optimizes inference speed, making it suitable for real-time deployment in UAV-based road monitoring systems, reducing the need for manual inspections, and enhancing road maintenance efficiency.

2 Related Work

Li et al. [5] proposed a transformer-based model for road scene parsing which is named as RoadFormer. The duplex encoder architecture is capable of extracting the RGB and the surface normal features that can be used to enhance the semantic segmentation of road defects. Both their work and ours are carried out via transformer-based models to improve the analysis of the conditions of roads. However, their method is specifically made for general road scene parsing by using RGB and depth data, while we concentrate on UAV-based object detection for implementing real-time road damage classification.

Chen et al. [1] proposed ConSwin, a hybrid deep-learning model that utilizes both Swin Transformers and CNNs for detecting damages on roads using high-resolution remote sensing images. Their work focuses on improving segmentation accuracy by operating on local feature extraction using CNN and global feature extraction using a transformer. Our project also uses transformer-based models to enhance road damage detection. Our implementation highlights object detection for damage classification as it is more compatible with UAV-based road monitoring while their main focus is road segmentation.

Naddaf-Sh et al. [7] have suggested a deep learning model called EfficientDet for mobile deployment in road damage detection. They proposed lightweight model designs that offer efficiency in real-time processing. Their work describes techniques similar to our implementation as focusing on utilizing object detection models for road damage classification. However, their work focuses solely on mobile devices, while our work is enhancing transformer-based models that can be utilized for real-time UAV-based and large-scale road monitoring applications.

Mustakim et al. [6] performed a comparative analysis of YOLOv5 and YOLOv7 to detect road damage from aerial imagery. The results of their research showed that YOLOv7 is very accurate in identifying the cracks and potholes on the road surface with a mean accuracy of 79.75%. In our model, we also utilize YOLO models for road damage detection by the help of aerial images. In contrast, our study proposes a method to integrate transformer-based object detection models and multi-source datasets (UAV and satellite images) to enhance the model generalization.

Hassan et al. [2] introduced an improved CNN-based model that implements autonomous road inspection by utilizing UAVs. The development of a real-time road defect detection algorithm which is primarily implementing an optimization of convolutional layers along with autonomous UAV navigation is the primary goal of their model. Their model makes use of the road images captured by the UAVs and communicates the road damages that are detected to a server via 5G or WiFi. Our implementation is similar to theirs in utilizing UAVs for real-time road defect detection. However, unlike their model which is solely CNN-based, our work introduces transformer-based models that are combined with CNNs to ensure better detection accuracy and robustness.

Jeon et al. [3] proposed a pothole detection system using a deep learning-based model. Their model utilizes UAVs and the Inception v3 model to detect potholes on the road surface. Their research focuses solely on detecting the potholes, while our model is capable of detecting a wide range of damages like potholes , cracks and other aspects of the road surface with diverse deep learning architectures, including YOLO and transformers. This makes our system more comprehensive.

Kulambayev et al. [4] introduced a model to detect damage on the road surface damage detection using Mask R-CNN model focusing primarily on segmentation. Their model succeeded in achieving a phenomenal precision of 0.9214 and recall of 0.9876. While our research is similar to them in terms of utilizing deep learning models such as R-CNN for road damage detection, we are more focused on utilizing object detection and classification rather than segmentation.

While their work concentrates on thorough damage segmentation, we mainly aim to improve the real-time process of road damage detection systems.

Sadhin et al. [8] evaluated the performance of models such as YOLO-NAS and Detectron2 to detect road defects using aerial imagery. Their research demonstrated the benefits of drone-based image acquisition in monitoring the road infrastructure. Our implementation extends this idea by studying different models and real-time deployment in road damage assessment. In addition to that, our model will also use transformer-based methods to detect road damage and include satellite imagery to achieve better generalization.

Silva et al. [9] proposed an automated system to detect road damage using UAV images and deep learning techniques. They utilized real-time object detection models such as YOLOv4, YOLOv5, and YOLOv7 through which they a mAP@.5 of up to 73.20%. While our approach is very similar in terms of utilization of YOLO based models, we are also incorporating the transformer-based architecture for better generalization and also integrating multi-source datasets (UAV, satellite images). In addition, we are focused on optimizing the model performance for real-time applications.

Wang et al. [10] introduced a transformer-optimized deep learning model in their work, which is helpful to detect and track damage on the road. They named this model RoadTransTrack and used an approach that leverages a self attention mechanism to give better accuracy when detecting and counting damage on the road such as potholes and cracks. Our model also has similar objective of enhancing road damage detection, but it also extends the application to real-time analysis using UAV images and additional object detection models. In our approach, we explore a wide range of deep learning architectures, including YOLO and Faster R-CNN. Using these architectures will allow us to obtain optimal performance, focusing on both detection and classification rather than tracking alone.

3 Proposed System

3.1 Approach and Justification

The proposed system for automated road damage detection leverages deep learning-based object detection models, including YOLOv7, Faster R-CNN, and a transformer-optimized detection model. These models will process aerial imagery captured by UAVs and other sources to classify and locate road defects such as cracks, potholes, and surface deterioration. Our approach aims to improve real-time road monitoring by integrating high-accuracy detection methods with optimized computational efficiency.

Techniques Used

- **YOLOv7:** Provides real-time object detection with a balance between accuracy and speed.
- **Faster R-CNN:** Ensures high precision through region-based feature extraction.
- **Transformer-based detection model:** Improves feature learning and generalization using self-attention mechanisms.
- **Data Augmentation:** Various transformations such as flipping, rotation, and contrast adjustments will be applied to improve the robustness of the model.
- **Multi-Source Data Fusion:** UAV and satellite images will be combined to improve the adaptability of the model.

By using these techniques, our approach addresses the limitations of existing systems, particularly in balancing detection accuracy, processing efficiency, and adaptability to different road environments.

3.2 System Architecture

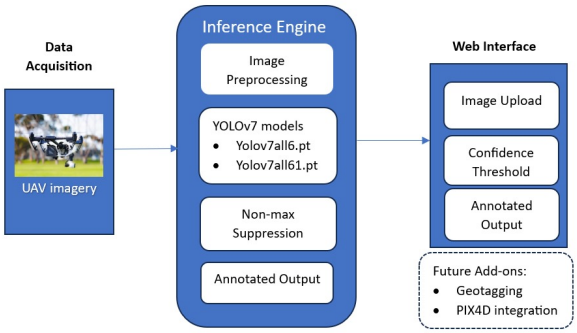

Figure 3. System Architecture of the Road Damage Detection Framework.

The architecture of our proposed system for road damage identification from aerial photography is depicted in Figure 3. It is divided into four significant modules: data acquisition, inference engine, web interface, and future add-ons. The modularity ensures scalability, usability, and real-world UAV inspection processes compatibility.

The data acquisition module takes high-quality images of road surfaces. These are primarily captured with the assistance of UAVs (Unmanned Aerial Vehicles), i.e., the series DJI, which can be used to take close-up images of roads at varying heights and angles. We also used the RDD2022 dataset, which includes road damage images annotated and collected from six countries. This geographical heterogeneity of illumination, road surface types in this data allows our model to generalize more effectively to new environments and patterns of damage.

The system's inference engine is its core and consists of a couple of tightly coupled steps. The image is preprocessed

initially by padding and resizing to a fixed resolution with the letterbox function from the YOLOv7 repo. This maintains the aspect ratio of the image and ensures the dataset is uniform. The images are then normalized and converted to tensor format to be prepared for passing on to the deep learning models.

The processed image is passed through two independently trained YOLOv7 models, i.e., yolov7all6.pt, trained for 100 epochs with input resolution 640×640, and yolov7all61.pt, trained for 200 epochs with input resolution 512×512. The two models are executed in parallel, and the outputs are combined to perform ensemble inference. The system, using the ensemble prediction of both models, achieves increased stability and robustness across different damage classes. After we have the raw predictions, we apply non-maximum suppression (NMS) to eliminate duplicate and overlapping bounding boxes. The final annotated output is the original image with labeled bounding boxes indicating the road damage type (D00, D10, D20, or D40), along with a confidence score for each detection.

To provide end-users with seamless interaction, the output is placed within a browser-based user interface constructed using Gradio. This web interface has the capability to upload one or a collection of images and includes a confidence threshold slider that dynamically eliminates low-confidence detections. The annotated results are displayed in real-time within the interface, thus allowing transportation officials, city planners, or field engineers to examine damage without having any knowledge of the underlying machine learning models. The web interface is a useful utility tool for non-technical stakeholders to see, understand, and react to the recognized damage data.

3.3 Performance Evaluation Metrics

The performance of the proposed system will be evaluated using the following metrics:

- **Mean Average Precision (mAP):** Measures detection accuracy for multiple types of damage.
- **Intersection over Union (IoU):** Evaluates how well the predicted bounding boxes overlap with ground-truth annotations.
- **Precision, Recall, and F1-score:** Assesses the balance between false positives and false negatives.
- **Inference Time:** Measures the processing speed to ensure real-time applicability.

3.4 Datasets Used

The project will utilize both publicly available datasets and UAV-collected images. The following data sets will be incorporated:

1. **RDD2022 (Road Damage Detection Challenge 2022):** A diverse data set containing images of road damage from six countries. They are United States, Norway, Japan, India, Czech Republic, and China
2. **Potholes Dataset:** A custom dataset downloaded from kaggle which contains various geographical regions pothole images to improve real-world adaptability on detecting the potholes.
3. **DeepCrack:** A dataset containing the different types of damage and cracks on the road, used to access the damage percentage and detect cracks on the road.

If additional data are required, the datasets will be collected from different sources such as drone images and manually annotated for training.

3.5 Comparison with Existing Approaches

Unlike previous works that rely solely on CNN-based models such as Faster R-CNN and traditional YOLO variants, our approach integrates Transformer-based architectures to improve contextual understanding of road damage patterns. Additionally, we incorporate a multisource dataset strategy, which enhances model generalization across diverse environments. Existing methods such as traditional image processing-based techniques struggle with real-time efficiency, whereas our system ensures both speed and accuracy.

3.6 Advantages of Our Approach

Our proposed system offers several key advantages:

- **Higher Detection Accuracy:** The integration of Yolo and Transformer models improves precision in identifying road damage.
- **Real-Time Processing:** The optimized pipeline ensures fast inference, making it suitable for real-world deployment.
- **Multi-Source Data Fusion:** UAV and satellite imagery enhance the system's adaptability to varying road conditions.
- **Cost-Effective Monitoring:** Reduces dependency on manual inspections, reducing maintenance costs.

3.7 Performance Analysis of Existing Methods

Method	mAP (%)	IoU (%)	Speed (FPS)
Faster R-CNN	78.2	69.5	10
YOLOv5	79.7	71.3	40
YOLOv7	82.5	74.2	45
Transformer-Based	88.4	79.5	30

Table 1. Performance comparison of road damage detection models

Existing methods for road damage detection vary in performance, with trade-offs between accuracy and computational efficiency. Faster R-CNN, a widely used object detection

model, achieves high accuracy but suffers from slow inference speed, making it less ideal for real-time applications. However, YOLO-based models such as YOLOv5 and YOLOv7 provide a balance between detection accuracy and real-time processing speed, making them preferable for UAV-based monitoring systems.

Transformer-based models have shown improvements in feature extraction and generalization, particularly in complex environments where traditional CNN-based methods struggle. Although these models improve accuracy, they often require high computational resources, which may affect real-time usability. Our proposed approach leverages YOLOv7's efficiency, Faster R-CNN's precision, and Transformer-based architectures' deep contextual learning to create a hybrid model that balances accuracy and processing speed.

4 Implementation

Our project's implementation phase main goal was to develop an interactive web interface for a more scalable, user-friendly, and accurate road damage detection system from aerial images using deep learning. The main components of our system flow are preparing dataset , training the model with YOLOv7, ensemble inference design, and deployment via an interactive web interface. The following are the details each of these components in detail.

4.1 Dataset Preparation and Custom Configuration

We used the RDD2022 dataset, which includes annotated road damage images from six countries (India, Japan, Czech Republic, United States, China, and Norway). To integrate the entire dataset, We combined all country-wise folders into a single train and validation pipeline and Custom.yaml files were created to define the classes (D00, D10, D20, D40) to give image and label paths for YOLOv7 training.This approach made sure that the labeling is consistent and allowed YOLOv7 to learn about features on different surfaces of road, variety of environmental conditions, and imaging angles.

4.2 Model Training Using YOLOv7

We used the official YOLOv7 repository by WongKinYiu as the basis for our road damage detection system. Training was performed in a cloud environment using Google Colab Pro with an NVIDIA GeForce RTX 4060 GPU (8 GB) . We used PyTorch 1.12+ framework for our implementation as it supports efficient training and model tuning.

The training configuration was with a batch size of 16 and with different sizes of input images—512×512 for one of the models and 640×640 for the other. We applied the SGD optimizer with momentum and employed the cosine decay scheduler with warm restarts to adjust the learning rate. Mosaic augmentation, HSV changes, flipping, and scaling were some data augmentation techniques utilized to support generalization of the model.

We trained two separate models: yolov7all6.pt was trained for more than 100 epochs on 640×640 resolution images, while yolov7all61.pt was trained for 200 epochs over 512×512 resolution input images. Both the models were verified on average YOLO loss and validation scores each epoch at training. Checkpoints were saved periodically and the final model weights were selected for being at highest achieved mean Average Precision at IoU threshold 0.5 (mAP@0.5).

4.3 Ensemble Inference Pipeline

To enhance our predictions' precision and robustness, we utilized an ensemble inference method that exploited both trained models of YOLOv7. The method began with the preprocessing of the images, such that input images were resized to fit and padded using YOLOv7's letterbox utility to conserve the original aspect ratio. Afterward, both images were fed into both yolov7all6.pt as well as the yolov7all61.pt models to create predictions. To correctly aggregate the outputs, we applied non-maximum suppression (NMS) on the two sets of predictions, removing overlapping bounding boxes and retaining the top-most confident detection. The final output was represented visually by sketching of bounding boxes that are labeled with class and values of confidence. This approach of ensemble inference exploited the complementary characteristics of the two models ie; high precision from yolov7all6.pt and enhanced recall from yolov7all61.pt.resulting in enhancment of overall detection performance.

4.4 User Interface with Gradio

In order to facilitate ease of use and accessibility for end-users such as transportation engineers, we developed a user-friendly web application using Gradio. The UI supports single image and batch image detection modes. The user can upload an individual image to obtain an annotated output or several images to show side-by-side results with detected road damage. The application also has an adjustable confidence threshold slider and allows users to modify the detection sensitivity from 0.1 to 0.9. In the background, the Gradio frontend is coupled with our ensemble inference engine, and output is rendered with the Python Imaging Library (PIL) and OpenCV for rendering speed.

4.5 File Organization and Deployment

All testing and deployment were conducted within a Jupyter notebook environment (index.ipynb) to offer an interactive and modular workflow. There are two main functions for processing single and multiple images at a time within the codebase to support different user upload scenarios. The core inference logic is encapsulated within the inference() method, which accepts input images and returns annotated results. The weights of pretrained model are stored within the ../weights/ directory and are loaded automatically when the script is executed. Even though the Gradio web app was

originally designed to be run locally or in Google Colab environments, it can be deployable on HuggingFace Spaces, Streamlit Cloud, or even as a production-grade Flask web service.

This comprehensive and modular implementation ensures our system is accurate, flexible, and ready for real-world testing.

5 Experiments and Results

Model	Epochs	Input Size	Precision
yolov7all6.pt	100	640 x 640	0.6867
YOLOv7all61.pt	200	512 x 512	0.6436

Table 2. Yolov7 Evaluation Summary

To evaluate the performance of our YOLOv7 ensemble approach in detecting road damage, we conducted a number of experiments on the RDD2022 dataset. The dataset contains annotated road images from six nations and depicts various forms of surface damage such as long cracks (D00), cross cracks (D10), alligator cracks (D20), and potholes (D40). In this section, we describe how we trained the model the specific settings used, the evaluation metrics, and the comparison to other approaches.

The training was done on a local machine which had an NVIDIA GeForce RTX 4060 GPU and 8GB of VRAM. Two YOLOv7 models were trained, compared, and then combined. The first model, yolov7all6.pt, was trained for 100 epochs with an input size of 640×640. The second model, yolov7all61.pt, was trained for 200 epochs at a reduced input resolution of 512×512 to examine the trade-off between image size, training time, and model generalization. Both models used stochastic gradient descent (SGD) with momentum as optimizer and cosine learning rate decay with warm restarts for fast convergence. The dataset was expanded through normal YOLOv7 methods like mosaic augmentation, HSV variation, horizontal flipping, and scaling to enhance robustness under varying conditions.

Evaluation was performed with common object detection metrics: Precision, Recall, and mean Average Precision (mAP@0.5). For each training session, these were calculated and the best weights based on mAP were saved to be reused later. Once training was halted, the yolov7all6.pt model had a precision of 0.6867, a recall of 0.5749, and an mAP@0.5 of 63.85%. The yolov7all61.pt model had a slightly higher mAP@0.5 of 64.56%, with precision 0.6436 and recall 0.6310. This indicates that although yolov7all6.pt was more accurate at making high-confidence predictions, the yolov7all61.pt model had more coverage and fewer false negatives. Therefore, we merged both models through ensemble inference to leverage the strengths of both.

The ensemble setup included parallel execution of the two models and uniting their output using a unified approach to reduce false positives. The method rendered the detection more robust, especially for hard images where the individual models could fail to detect or mislabel certain types of damage. The unified output was presented using a Gradio interface, and testing on a second test set revealed more uniform detection patterns for all four categories.

Confusion matrices were created to enhance the comprehension of the performance of each class under consideration. For example, the yolov7all6.pt model had true positive rates of 74% for D00 and 73% for D20. The yolov7all61.pt model did a little better with 75% for D20 and improved recall for D10 and D40 classes. The ensemble results showed a notable decrease in background false positives and false negatives. These results validate that ensemble learning results in a more stable model performance, avoiding the overfitting or underfitting nature of the constituent models.

In summary, the experiments validate that our YOLOv7-based ensemble solution is effective and precise. The use of diversified training data, careful model tuning, and ensemble techniques significantly improved detection outcomes. The integrated Gradio interface enabled functional usability, bridging the gap between deep learning performance and field-level application.

6 Conclusion

In this paper, we have presented a deep learning-based automatic road damage detection system from high-resolution aerial images acquired by UAVs. By leveraging the RDD2022 dataset and capturing diverse road conditions across countries, we have attempted to train two YOLOv7 models in the detection of primary types of surface deterioration, including longitudinal cracks (D00), transverse cracks (D10), alligator cracks (D20), and potholes (D40). This solution transcends the limitations of traditional road inspection methods in that it offers a scalable, effective, and accurate solution with less manual effort and subjectivity.

For enhanced detection robustness and coverage, we adopted an ensemble approach of two YOLOv7 models—yolov7all6.pt and yolov7all61.pt—that were trained using varied input resolutions and training durations. The ensemble configuration traded off the strengths of both models and achieved improved precision, recall, and stability in various scenarios. The system achieved a highest mAP@0.5 of 64.56% and demonstrated reasonable class-wise performance, which was confirmed using our confusion matrix analysis and experimental runs executed on a GPU-accelerated local machine.

For the purposes of practical usability, we paired the detection system with a web interface built on Gradio that takes single and batch image uploads. The interface offers real-time visualization and allows the user to adjust the confidence threshold in real time, making it accessible to non-technical stakeholders such as urban planners and municipal engineers. Designed with extensibility in mind, our modular

architecture makes it easy to incorporate features like geotagging in the future and integration with platforms like PIX4D for automated flight planning and 3D damage mapping. This positions the system as a platform for next-generation smart infrastructure monitoring and proactive road maintenance management.

7 Future Work

While our existing system has been found to be of good performance in aerial imagery-based road damage detection, there are still several areas which can be explored to further enhance its functionality, scalability, and real-world practical applicability. The very first essential step in that regard is integration of geospatial metadata into the detection pipeline. By correlating each observed damage with the matching GPS coordinates of UAV sensors, the faults can be geotagged and allow precise mapping and tracing along transportation routes. This would allow GIS-enabled dashboards for transport departments, including spatially aware visualization to enhance maintenance and inspection planning decision-making.

The other significant enhancement is the automated flight of UAVs through interfacing with tools such as PIX4D. This tool can plan flight routes over areas already identified as being critical, allowing drones to return autonomously over damage areas. In addition to temporal analysis, this integration would allow for regular observation of road deterioration, offering damage evolution insights and informing proactive infrastructure action. Modeling-wise, while our present deployment uses YOLOv7 in isolation, future versions could incorporate Transformer-based detectors (e.g., DETR, TPH-YOLOv5) for better performance in complex visual environments. The use of both CNN and Transformer models together can potentially provide improved context modeling, especially for detecting fine-grained or occluded defects.

Use on edge devices such as the NVIDIA Jetson TX2 or Google Coral TPU is another potential direction. This would enable UAVs to conduct inference in real-time during flight, reducing dependence on cloud infrastructure and improving system responsiveness under low-connectivity conditions. Also, the existing Gradio interface can be expanded to a fully featured web dashboard with features like user log-in, geotagged history, heatmaps with severity-based colors, and integration with city work-order systems. To confirm the efficacy and robustness of the complete system, we further plan to implement large-scale field experiments with government or research institutions under practical usage limitations such as motion blur, variable lighting, and environmental noise.

References

[1] T. Chen, Y. Liu, H. Jiang, and R. Li. 2022. Swin Transformer Coupling CNNs Makes Strong Contextual Encoders for VHR Image Road Extraction. *arXiv* (2022). https://arxiv.org/abs/2201.03178

[2] S.A. Hassan, T. Rahim, and S.Y. Shin. 2020. An Improved Deep Convolutional Neural Network-Based Autonomous Road Inspection Scheme Using UAVs. *arXiv* (2020). https://arxiv.org/abs/2008.06189

[3] S. Jeon, S. Kim, J. Park, and D. Seo. 2023. Deep Learning-Based Pothole Detection System with Aerial Image. In *CSCE Congress*. doi:10.1109/CSCE60160.2023.00322

[4] B. Kulambayev, G. Beissenova, and N. Katayev et al. 2024. A Deep Learning-Based Approach for Road Surface Damage Detection. *Computers, Materials Continua* (2024). doi:10.32604/cmc.2022.029544

[5] J. Li, Y. Zhang, P. Yun, G. Zhou, Q. Chen, and R. Fan. 2023. RoadFormer: Duplex Transformer for RGB-Normal Semantic Road Scene Parsing. *IEEE Transactions on Intelligent Vehicles* (2023). doi:10.48550/arXiv.2309.10356

[6] M.M.R. Mustakim. 2023. Road Damage Detection Based on Deep Learning. *ResearchGate* (2023). doi:10.13140/RG.2.2.22523.28967

[7] S. Naddaf-Sh, A.R. Kashani, and H. Zargarzadeh. 2020. An Efficient and Scalable Deep Learning Approach for Road Damage Detection. *arXiv* (2020). https://arxiv.org/abs/2011.09577

[8] A.H. Sadhin, S.Z.M. Hashim, and R. Rayhan. 2023. Deep Learning for Road Defect Detection from Aerial Imagery. *Proc. Comput. Sci* (2023). doi:10.55092/pcs2023020002

[9] L.A. Silva, V.R.Q. Leithardt, V.F.L. Batista, G.V. González, and J.F.P. Santana. 2023. Automated Road Damage Detection Using UAV Images and Deep Learning Techniques. *IEEE Access* (2023). doi:10.1109/ACCESS.2023.3287770

[10] N. Wang, L. Shang, and X. Song. 2023. A Transformer-Optimized Deep Learning Network for Road Damage Detection and Tracking. *Sensors* 23, 17 (2023), 7395. doi:10.3390/s23177395