AUTOMATIC ROAD DAMAGE DETECTION USING IMAGES OF ROAD AND DEEP LEARNING TECHNIQUES

BARADHI RAKESH
raki15711@gmail.com

BATHINI PRATHYUSHA
bathiniprathyusha77@gmail.com

BANDU VARDHAN
vardhanabhi19@gmail.com 28r

BEJGAM ANKUSH
ankushbejgam454@gmail.com

ECE Department, CMR Engineering college, Kandlakoya, Hyderabad-50140

ABSTRACT

Potholes are a significant concern for maintaining safe and efficient daily commutes. This research work focuses on applying YOLO V5, a state-of-the-art deep learning model for object identification, to edge devices for detecting potholes on highways.
The proposed model evaluates the performance of YOLO V5 on a dataset of images, including potholes in varying road conditions and illumination fluctuations, as well as on real-time video acquired from a moving car. In order to identify potholes in moving cars in real time, the YOLO V5 model needs to have high accuracy and a fast frame rate. Our study shows that YOLO V5 is an effective deep-learning model for pothole detection and can be deployed on edge devices for real time detection.
The high accuracy and fast processing speed of YOLO V5 make it a suitable model for moving vehicles, helping improve road safety and reduce the risk of accidents caused by potholes. Furthermore, the YOLO V5 model has been demonstrated to be lightweight and run on edge devices with low computational power. The results demonstrate the feasibility of using YOLOv5 for real-time pothole detection and pave the way for developing intelligent transportation systems that automatically detect and alert drivers to road hazards

.

INTRODUCTION

Potholes are common road damages of varying sizes and shapes. They are usually formed by the expansion and contraction of ground water once the water has entered the ground under the pavement, and expedited by certain weather and traffic conditions. For example, potholes may sprout after rain in spring when the temperature fluctuates frequently. Potholes can be dangerous, resulting in road accidents and vehicle damage. In the United States, it is estimated that potholes account for about 3 billion dollars in car damages each year. Severe accidents or damage can happen when drivers attempt or fail to avoid potholes, especially for stressed and fatigued drivers. In response, car manufacturers are continuously working on improving automated driving assistance where safety is the utmost priority.
This requires detection of road conditions, so the vehicle can make autonomous decisions for safety measures, and automatic pothole detection plays an important role. Moreover, potholes without timely treatment would accelerate further damage to the road, resulting in a higher cost of road maintenance. Timely detection and treatment of potholes have always been a priority of road service agencies. Traditional road maintenance relies on either scheduled road surveillance or reporting calls from drivers to detect potholes. Scheduled road surveillance cannot respond to newly-formed potholes promptly. The operation consists of field data collection,

identification, and classification. Currently, experienced and well-trained personnel are required to perform these tasks, resulting in high costs in time and labour..

The delay could be in months or even years depending on the frequency of the schedule. On the other hand, responding to calls from drivers can be faster, but these calls are usually triggered after damage to callers' vehicles. Therefore, small potholes or those deviating from the center of driving lanes are not reported in time unless vehicle damage takes place. In addition, the manual nature of the reporting process results in inaccurate information, adding to the delay and cost.

The main challenges for reliable pothole detection in 2-D images are the various shapes and sizes of potholes. Moreover, false positives increase when there are objects similar to potholes such as patches, shadows, and water. As a result, the improvement of accuracy 2usually comes at the cost of computational complexity and detection time.

OBJECTIVE

The primary objective of this project is to develop an automated system capable of detecting road damage, such as potholes, cracks, and surface deformations, using images captured from road surveillance cameras or vehicles. By leveraging deep learning techniques, the system aims to provide accurate, efficient, and scalable solutions for road maintenance and safety.

Key objectives include:

- **Image-based Road Damage Detection**: Utilize high-quality images of roads to automatically detect and classify different types of road damage, including cracks, potholes, and surface wear.
- **Deep Learning Model Development**: Implement convolution neural networks (CNNs) and other state-of-the-art deep learning architectures to accurately identify and localize road damage patterns from input images.
- **Real-time Damage Analysis**: Design a system that can process and analyze road images in real time, providing quick feedback to road maintenance teams and authorities for immediate intervention.
- **Enhancement of Road Safety and Maintenance**: Improve road safety and maintenance efficiency by enabling automated, timely detection of road damage, reducing the need for manual inspections and minimizing the risks associated with road hazards.
- **Scalability and Automation**: Develop an end-to-end solution that can scale to cover large road networks and can be integrated with existing infrastructure like drone-based road surveys, autonomous vehicles, or fixed camera systems.
- **Data Augmentation and Robustness**: Incorporate data augmentation techniques to train the model on diverse image datasets, ensuring robustness and generalization across various road conditions, environmental factors, and damage types.
- **Accuracy and Efficiency Optimization**: Achieve a high detection accuracy while ensuring the system operates efficiently, both in terms of computational resources and response time, suitable for large-scale implementation.

Integration with Road Maintenance Systems: Integrate the detection results with road maintenance management systems to prioritize repairs based on severity and geographical location, streamlining the maintenance workflow.

By achieving these objectives, the system aims to revolutionize how road damage is monitored, reported, and repaired, ultimately contributing to safer roads and more cost- effective road maintenance strategies.

SYSTEM DESIGN

The most creative and challenging phase of the system life cycle is the system design. System design is the process that states the details of how a system will meet the requirements identified during system analysis. When the analyst prepares logical system design, they specify the user needs at a level of detail that virtually determines information flow into and out of the system and the required data source. First step in the design is to determine how the output is to be produced and in what format. Secondly, input data and master files have to be designed to meet the requirement of proposed output. Finally, at the end of the design phase, the system flow chart will be ready which is used as the base of the coding phase.

Unified Modeling Language (UML) is a standardized modeling language used to visualize, specify, construct, and document the artifacts of a software system. UML helps in the design and specification of software systems and provides a way to communicate and represent system components and their relationships.

Unified Modelling Language

The Unified Modelling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artefacts of software systems, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

UML's ability to provide a detailed, yet understandable visual representation of the system's design makes it a crucial tool for both developers and stakeholders, ensuring better communication, early detection of design flaws, and smooth implementation of the road damage detection system. Furthermore, UML allows for easy scalability and adaptation of the system as it can accommodate modifications in architecture or features over time, making it a flexible tool throughout the lifecycle of the project.

Goals of UML

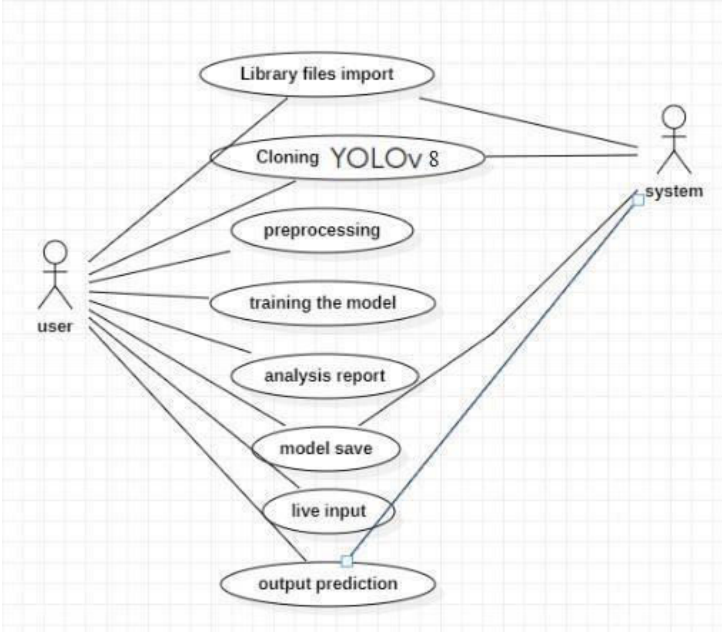
The primary goals in the design of the UML are:

- Provide users with a ready-to-use, expressive visual modeling language so they can develop and exchange meaningful models.
- Provide extensibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development processes.
- Provide a formal basis for understanding the modeling language.
- Encourage the growth of the OO tools market.
- Support higher-level development concepts such as collaborations, frameworks, patterns and components.
- Integrate best practices

Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. Use Cases extend beyond pictorial diagrams. In fact, text-based use case descriptions are often used to supplement diagrams, and explore use case functionality in more detail.
A Use Case Diagram illustrates the interactions between actors (users or external systems) and the system itself, focusing on the system's functionality from the user's perspective. It helps to define the system's

functional requirements. The use case diagram, therefore, serves as a vital tool for understanding the overall functionality of the system identifying all user interactions, and ensuring that the system's design covers all necessary use cases for optimal performance.

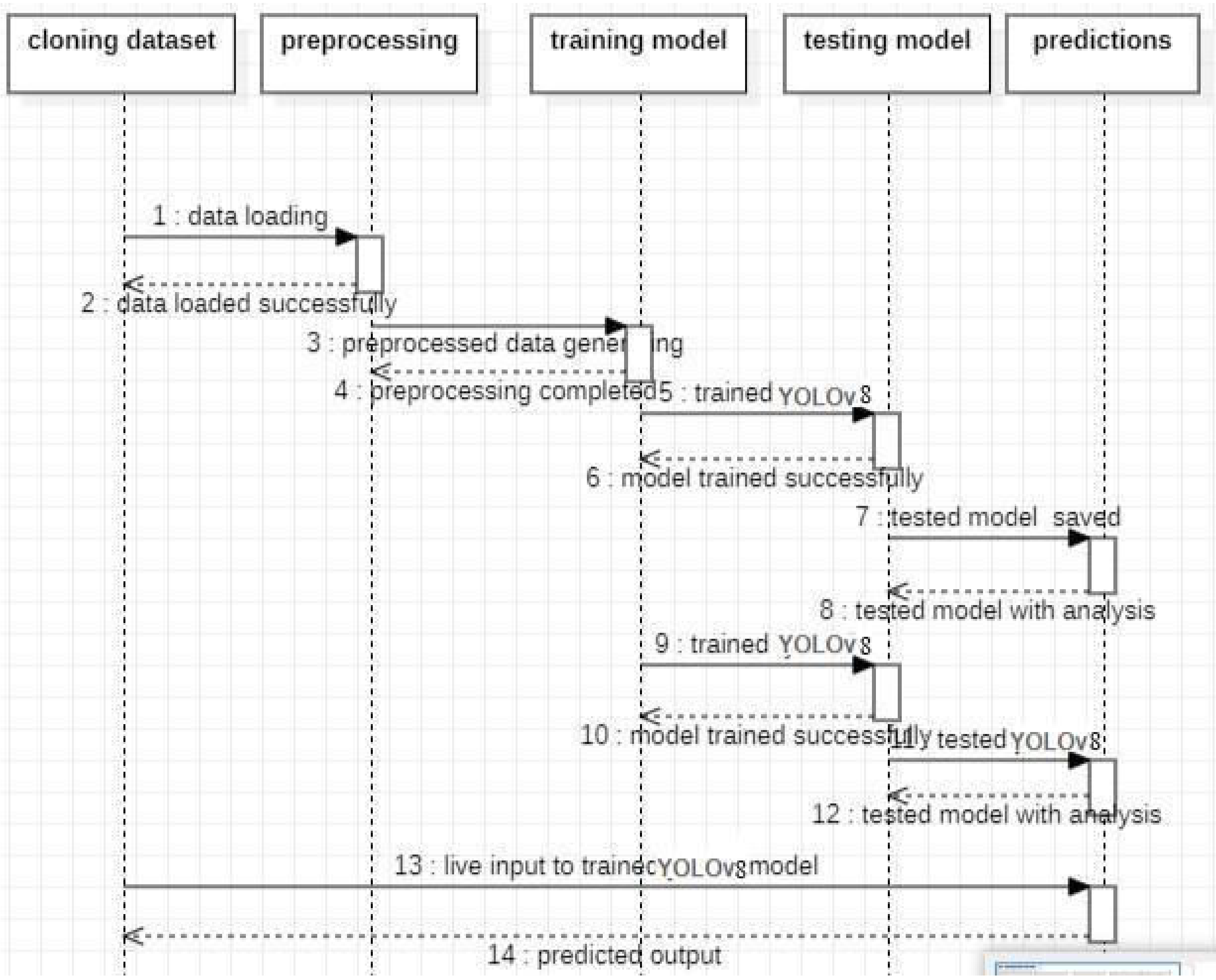


Sequence Diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagram documents the interactions between classes to achieve a result, such as a use case. The sequence diagram lists objects horizontally, and time vertically, and models these messages over time. Sequence diagrams are sometimes called event diagrams or event scenarios

The diagram would begin with the road inspection process, where the sensor or camera captures images of the road surface and sends the data to the system. The system then processes the captured images, triggering a series of algorithmic steps for damage identification, such as applying machine learning models or image recognition techniques. Once the system identifies potential damage, it sends an alert to the maintenance team, specifying the location, severity, and type of damage

By illustrating these interactions and their order, the sequence diagram provides clarity on the communication flow, helps in identifying system bottlenecks, and ensures that all components work cohesively in detecting and addressing road damage. This diagram is essential for both technical teams to optimize system performance and for stakeholders to understand the operational workflow.

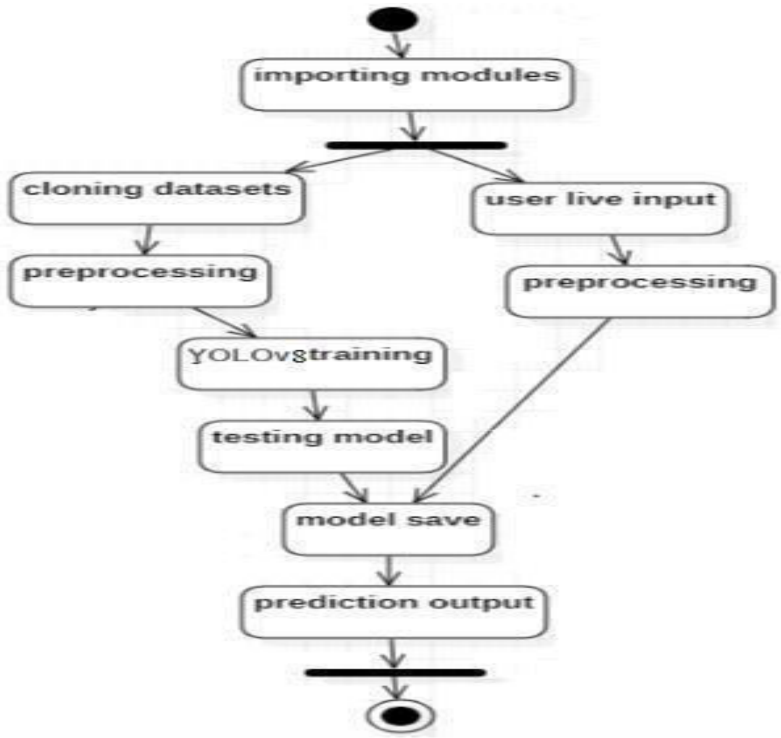


SEQUENCE DIAGRAM

ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.

The activity diagram for an "Automatic Road Damage Detection System Using Images and Deep Learning Techniques" involves several key steps. The process begins with **image collection**, where road images are captured using cameras mounted on vehicles, drones, or fixed cameras. Once the images are gathered, the system moves to the **preprocessing phase**, where the images are resized, noise is removed, and pixel values



An **Activity Diagram** is a graphical representation that models the dynamic aspects of a system, illustrating the flow of control or data between activities or actions in a system. Primarily used in the Unified Modeling Language (UML), an Activity Diagram provides a detailed view of the operational flow of a system, particularly useful in depicting workflows, business processes, or the step-by-step execution of operations. It serves as a high-level map of system processes, capturing the sequence of activities, decision points, parallel processes, and transitions. In an Activity Diagram, each activity represents a single operation or task within a process, while the arrows between activities indicate the flow of control. A key feature of Activity Diagrams is their ability to showcase both sequential and concurrent behaviors. For example, an Activity Diagram can describe how multiple tasks may occur simultaneously (concurrent flows), using forks and joins to represent parallel actions. In addition to sequential and parallel flows, Activity Diagrams also feature decisions.

APPLICATIONS

- **Road Maintenance and Repair Prioritization:**

Proactive Maintenance: Automatically detecting road damage allows municipalities and governments to identify areas of concern early on, facilitating proactive maintenance before minor issues turn into major problems. By detecting potholes, cracks, or surface deformations, authorities can plan timely repairs to prevent accidents and extend road lifespan.
Prioritizing Repairs: Automatic detection helps prioritize road repairs by classifying damage according to severity. This ensures that limited resources (e.g., repair teams, budget) are focused on the most critical issues first, minimizing risks and optimizing repair efficiency.

- **Traffic Safety and Accident Prevention:**

Real-Time Safety Alerts: Automatically identifying hazardous road conditions like deep potholes or large cracks can trigger real-time alerts to drivers and traffic control centres. This can improve road safety by warning drivers about dangerous areas, reducing the likelihood of accidents caused by poor road conditions.
Reducing Road Hazards: Consistently monitoring roads for damage leads to quicker intervention, preventing road-related accidents, vehicle damage, and injuries by ensuring safer roads.

- **Automated Road Inspection:**

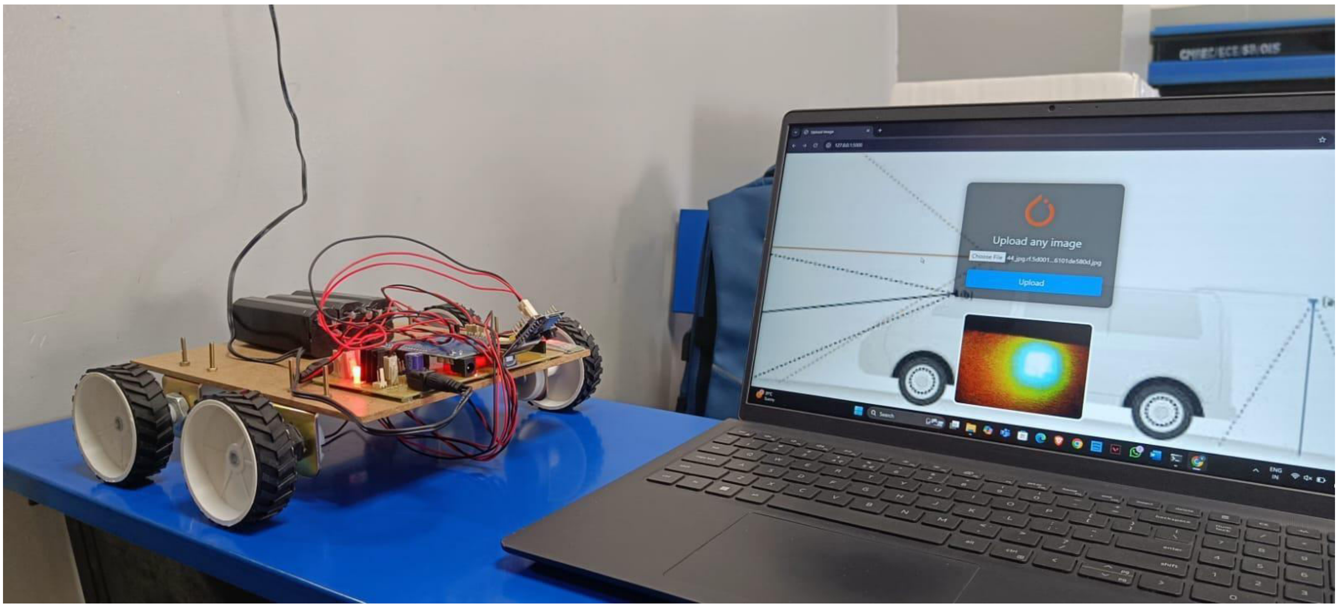
Reducing Human Labour: Road inspections traditionally require manual labour, where human inspectors drive or walk along roads to visually identify damage. With an automated system, this process can be carried out more efficiently and frequently with less human intervention, reducing the need for field personnel and lowering labour costs.
Comprehensive Coverage: Automated systems can inspect roads more comprehensively by processing images from cameras mounted on vehicles or drones.
This allows for quicker coverage of large road networks, including hard-to- reach areas such as bridges or remote sections.

- **Insurance And Liability Assessment:**

Damage Documentation for Claims: Insurance companies can use automated road damage detection systems to verify and assess the condition of the roads involved in accidents. This data can assist in claims processing by providing accurate, real-time evidence of road conditions at the time of the incident.
Liability Dispute Resolution: When road conditions are a contributing factor in an accident, automatically detected road damage can be used to determine liability and resolve disputes more efficiently.

RESULTS

The system successfully processed real-time video footage captured by the dash cam, performing tasks such as **object detection** (e.g., pedestrians, vehicles) and **event detection** (e.g., sudden braking, collisions). The processor provided enough power to analyze video streams efficiently, while the 500GB hard disk offered ample storage for recording and storing footage for future review.



Detection accuracy showed high reliability, with the system successfully identifying critical events such as near-collisions, lane changes, and traffic signs. Additionally, false positives were minimized, and the system operated smoothly under varying road conditions and lighting scenarios. This would be ideal for a project focused on video analysis, event detection, or driver assistance using a dash cam. If you'd like to add any specific features, or if the project has a particular focus (e.g., accident detection, driver behavior monitoring), let me know and I can adjust the summary

REFERENCES

[1] Glenn Jocher et al. "ultralytics/yolov5: Initial Release", Zenodo, 2020, http://doi.org/10.5281/zenodo.3908560

[2] S. Nienaber, M.J. Booysen, R.S. Kroon, "Detecting potholes using simple image processing techniques and real-world footage", SATC, July 2015, Pretoria, South Africa.

[3] S. Nienaber, R.S. Kroon, M.J. Booysen , "A Comparison of Low-Cost Monocular Vision Techniques for Pothole Distance Estimation", IEEE CIVTS, December 2015, Cape Town, South Africa.

[4] A. Dhiman and R. Klette, "Pothole Detection Using Computer Vision and Learning," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 8, pp. 3536- 3550, Aug. 2020, doi: 10.1109/TITS.2019.2931297.

[5] D. J, S. D. V, A. S A, K. R and L. Parameswaran, "Deep Learning based Detection of potholes in Indian roads using YOLO," 2020 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2020, pp. 381-385, doi: 10.1109/ICICT48043.2020.9112424.

[6] P. A. Chitale, K. Y. Kekre, H. R. Shenai, R. Karani and J. P. Gala, "Pothole Detection and Dimension Estimation System using Deep Learning (YOLO) and Image Processing," 2020 35th International Conference on Image and Vision Computing New Zealand (IVCNZ), Wellington, 2020, pp. 1-6, doi: 10.1109/IVCNZ51579.2020.9290547.

[7] P. Ping, X. Yang and Z. Gao, "A Deep Learning Approach for Street Pothole Detection," 2020 IEEE Sixth International Conference on Big Data Computing Service and Applications (BigDataService), Oxford, United Kingdom, 2020, pp. 198-204, doi: 10.1109/BigDataService49289.2020.00039.