AI Based Pothole Detection: An Intelligent Edge-Based System for Real-Time Road Damage Detection

Vivek Kumar Singh[1], Anurag Kumar[2], Ram Anuj Gupta[3], Manish Singh[4], Ritesh Kumar Singh[5]
[1,2,3,4] Computer Science and Engineering Noida Institute of Engineering and Technology Greater Noida, India
[5] Assistant Professor, Computer Science and Engineering Noida Institute of Engineering and Technology Greater Noida, India

***Abstract*—Road infrastructure degradation continues to pose a significant challenge for governments and urban planners worldwide. Among the various types of road damage, potholes and surface cracks are the most dangerous, contributing directly to traffic accidents, vehicle damage, and rising maintenance costs. Traditional inspection methods, which rely on periodic physical surveys, are proving inadequate in the face of expanding road networks and growing urban populations. This paper introduces AI Based Pothole Detection, an intelligent, edge-optimized road damage detection system that brings automated monitoring capabilities to real-world deployment scenarios. The proposed system leverages MobileNetV2, a lightweight convolutional neural network architecture trained using transfer learning, to classify four distinct categories of road surface damage: longitudinal cracks, transverse cracks, alligator cracks, and potholes. The model was trained on the publicly available RDD2022 dataset and deployed through a full-stack Node.js and Edge Impulse pipeline, enabling real-time inference on edge devices with response latencies below 100 milliseconds. Experimental evaluation shows that the system achieves classification accuracy in the range of 85–92% across damage categories. Additionally, the system incorporates entropy-based uncertainty quantification, AI-driven cost estimation, severity assessment, and heatmap visualization features, which collectively provide a production-ready monitoring tool. The presented solution demonstrates that combining edge AI with transfer learning and a lightweight web interface creates a scalable, accessible, and efficient framework for road maintenance management, contributing to the goals of smart transportation systems and intelligent urban infrastructure.**

***Index Terms*—Road damage detection, deep learning, MobileNetV2, edge AI, transfer learning, convolutional neural network, smart city, pothole detection, intelligent transportation**

I. INTRODUCTION

Road transportation is one of the foundational pillars of economic productivity and public mobility. In a rapidly urbanizing world, the condition of road surfaces directly influences travel safety, logistics efficiency, and overall quality of life. According to estimates, road damage costs governments across the globe well over three billion dollars annually in repair operations alone, not accounting for the economic losses from accidents and vehicle damage sustained by commuters.

In developing economies such as India, where road transportation accounts for over 60% of passenger movement and approximately 70% of freight distribution, maintaining road quality is not merely a matter of convenience—it is a national priority. Yet despite this importance, roads continue to deteriorate at an alarming rate due to environmental exposure, inadequate drainage systems, heavy vehicle loads, and inconsistent maintenance cycles. Potholes, in particular, form when water infiltrates weakened asphalt layers and vehicle pressure accelerates the collapse of the surface material. Left unaddressed, they expand rapidly and create compounding risks for motorists.

Existing approaches to road damage management are largely reactive. Municipal engineering teams conduct periodic visual surveys to document road conditions, and repairs are initiated only after damage becomes severe or public complaints accumulate. This approach is inefficient, resource-intensive, and fundamentally unable to scale to the size and complexity of modern urban road networks. Manual inspection teams are subject to human error, fatigue, inconsistent reporting standards, and inability to monitor roads in real time.

Recent advances in artificial intelligence, computer vision, and edge computing have created an opportunity to fundamentally rethink how road condition monitoring is performed. Convolutional neural networks (CNNs), particularly lightweight architectures such as MobileNetV2, have demonstrated strong performance on image classification tasks while remaining computationally feasible for deployment on edge hardware. Transfer learning further reduces the data and computational burden of training such models from scratch, enabling adaptation to domain-specific datasets with relatively modest resources.

This paper presents AI Based Pothole Detection, a complete end-to-end road damage detection system that integrates a transfer learning-based MobileNetV2 model with a full-stack web deployment architecture. The system is designed to address the limitations of manual inspection by enabling automated, real-time detection of four road damage categories: longitudinal cracks, transverse cracks, alligator (interconnected) cracks, and potholes. The deployment architecture consists of a Node.js backend server, an Edge Impulse inference engine, and an HTML5/JavaScript frontend interface that supports both image upload and camera-based capture. The system further provides AI-generated insights including confidence calibration, severity assessment, cost estimation, and visual heatmap overlays.

A. Motivation

The impetus for this research lies in the clear gap between the scale of road maintenance challenges and the capability of current monitoring tools. While deep learning-based detection systems have been demonstrated in research settings, many such systems rely on powerful GPU-equipped servers and large labeled datasets, making them impractical for municipal deployment. The need for an edge-deployable, lightweight, yet accurate detection framework is both timely and practically important.

B. Contributions

The primary contributions of this paper are as follows: (1) A complete road damage detection pipeline based on MobileNetV2 with Edge Impulse-based training and deployment; (2) A multi-class classification framework covering four road damage categories rather than the binary pothole/no-pothole formulation used in many prior works; (3) An advanced post-processing module incorporating entropy-based uncertainty quantification, repair cost estimation, and severity grading; (4) A production-ready full-stack web interface supporting real-time camera capture, batch processing, and live analytics; and (5) Evaluation on the RDD2022 multi-national benchmark dataset demonstrating competitive classification accuracy.

II. LITERATURE REVIEW

The problem of automated road damage detection has attracted growing attention in the computer vision and intelligent transportation research communities. Early approaches to road condition assessment relied on specialized sensor systems such as ground-penetrating radar and laser scanners mounted on dedicated survey vehicles. While highly accurate, these methods involve substantial infrastructure investment and are therefore unsuitable for wide-scale or continuous deployment.

The emergence of deep learning transformed the field significantly. Krizhevsky et al. [1] established convolutional neural networks as the dominant paradigm for visual recognition tasks, and subsequent architectures including VGG [2], ResNet [3], and Inception [4] achieved increasingly strong performance on benchmarks such as ImageNet. These architectures were subsequently adapted to domain-specific problems including medical imaging, satellite analysis, and infrastructure monitoring.

For road damage specifically, Maeda et al. [5] introduced one of the most widely cited datasets for road damage detection, collected using smartphones across Japanese roadways. Their work employed a modified YOLOv3 framework for real-time object detection and demonstrated that mobile device cameras combined with AI models can serve as practical monitoring tools. Later iterations of this dataset, including RDD2022, expanded coverage to multiple countries including Japan, India, the United States, Norway, and China, substantially improving the diversity and generalization capacity of trained models.

Object detection approaches, including Faster R-CNN [6], SSD [7], and the YOLO family [8], have been widely applied to the road damage problem. These methods can not only classify damage type but also localize affected regions within images through bounding box prediction. However, their computational demands—particularly for training—make them challenging to deploy on edge hardware without significant model compression.

Lightweight models designed explicitly for edge deployment have become increasingly relevant. Howard et al. [9] introduced MobileNet, a family of CNNs that use depthwise separable convolutions to dramatically reduce parameter counts while preserving representational capacity. MobileNetV2 [10] further improved upon the original design through the introduction of inverted residual blocks with linear bottlenecks, achieving better accuracy at similar computational cost. These properties make MobileNetV2 particularly well suited to real-time road damage detection on constrained hardware.

Transfer learning has consistently proven effective for domain-specific tasks with limited labeled data. By initializing a model with weights pretrained on ImageNet and fine-tuning on a domain-specific dataset, practitioners can achieve high accuracy with substantially smaller training sets. Several prior studies [11][12] have confirmed this strategy is highly effective for road damage classification tasks. Edge computing frameworks such as TensorFlow Lite and Edge Impulse have made it possible to deploy quantized neural network models on microcontrollers and embedded systems, enabling inference on devices as resource-constrained as Raspberry Pi modules.

III. PROPOSED METHODOLOGY

The AI Based Pothole Detection system follows a structured pipeline from data acquisition and preprocessing through model training, deployment, and post-processing. Each stage is described in detail in the following subsections.

A. System Architecture

The overall architecture consists of three primary components: a dataset preparation and model training pipeline, a backend inference server, and a frontend user interface. Data flows from the user’s camera or image upload through the web interface to the Node.js server, which invokes the Edge Impulse inference engine and returns structured predictions along with derived insights.

The system is designed with modularity in mind. The frontend communicates with the backend exclusively through a RESTful API, making it straightforward to substitute alternative model backends or extend the system with additional endpoints. The deployment architecture supports local development environments, Raspberry Pi edge deployment, and cloud-hosted configurations alike.

B. Dataset and Preprocessing

Model training utilizes the Road Damage Dataset 2022 (RDD2022), a multi-national benchmark that includes road images collected across Japan, India, the United States, Norway, and China. The dataset includes four damage categories: longitudinal cracks (D10), transverse cracks (D20), alligator cracks (D40), and potholes (D00). Each image is annotated with bounding box coordinates indicating the location and type of damage present.

For the classification task, images were resized to 320×320 pixels and converted to RGB format. Pixel values were normalized to the [0, 1] range through division by 255. Data augmentation was applied during training to improve model robustness, including random horizontal and vertical flipping, brightness adjustment within 넧30%, rotation up to 넰 degrees, and Gaussian blur with moderate kernel sizes. The dataset was partitioned into training (70%), validation (20%), and testing (10%) splits, with stratified sampling to ensure balanced class representation across all partitions.

TABLE I. DATASET COMPOSITION (RDD2022 SUBSET)

Damage Category	Label	Training	Validation	Testing
Longitudinal Cracks	D10	1,540	440	220
Transverse Cracks	D20	1,380	394	197
Alligator Cracks	D40	1,220	349	174
Potholes	D00	1,460	417	209
Total	—	5,600	1,600	800

C. Model Architecture and Transfer Learning

MobileNetV2, pretrained on the ImageNet dataset, was selected as the backbone feature extractor due to its combination of high accuracy and low computational overhead. MobileNetV2 employs inverted residual blocks with linear bottlenecks, using depthwise separable convolutions to reduce the number of parameters and multiplication-addition operations relative to standard convolutions.

The pretrained backbone's final classification layer was removed and replaced with a custom head consisting of a global average pooling layer, a dense layer with 128 neurons and ReLU activation, a dropout layer with rate 0.3 for regularization, and a final softmax output layer with four neurons corresponding to the four damage categories.

During training, the pretrained backbone weights were initially frozen for the first 10 epochs to allow the custom classification head to stabilize. In a subsequent fine-tuning phase, the top 30 layers of the backbone were unfrozen and trained at a reduced learning rate of 1×10^{-5}. This two-stage training protocol prevents the pretrained representations from being disrupted during early optimization while still allowing the backbone to adapt to road surface features. Training was conducted using the Adam optimizer with an initial learning rate of 1×10^{-3} and binary cross-entropy loss function. A cosine annealing learning rate schedule was applied to improve convergence stability. Early stopping with a patience of five epochs based on validation loss was employed to prevent overfitting.

D. Edge Deployment via Edge Impulse

Following training, the model was exported and deployed using the Edge Impulse platform, which provides tools for quantizing, testing, and deploying neural network models to edge environments. The model was quantized from 32-bit floating point to 8-bit integer representation (INT8), reducing the model file size to approximately 1.5 MB while incurring a minimal accuracy penalty of approximately 1–2%.

The deployed model is accessed through the Edge Impulse Node.js SDK within the backend server. When an image is submitted to the prediction endpoint, the server preprocesses the image using the Sharp library to resize and normalize it to the format expected by the model, then invokes the Edge Impulse inference API and receives a structured JSON response containing per-class probability scores and inference timing metadata.

E. Backend System

The backend is implemented as an Express.js server running on Node.js 14+. It exposes four REST API endpoints: a health check endpoint at GET /health, a single-image prediction endpoint at POST /predict, a batch prediction endpoint at POST /predict-batch for processing multiple images in sequence, and a model metadata endpoint at GET /model-info. File uploads are handled by the Multer middleware, which enforces a 10 MB file size limit and validates that submitted files are image MIME types. The server includes a fallback demo mode for development and demonstration purposes, which generates realistic synthetic predictions when the Edge Impulse API key has not been configured.

F. Post-Processing and Intelligent Insights

Beyond raw classification outputs, the system incorporates several post-processing modules that transform model predictions into actionable maintenance information. Entropy-based uncertainty quantification computes the normalized Shannon entropy of the prediction probability distribution, providing a measure of model confidence that supplements the raw probability scores. High-entropy predictions (entropy > 0.7) or low-margin predictions (margin < 0.2 between top two classes) trigger a flag recommending manual human review.

A repair cost estimation module maps detected damage type and severity level to indicative cost ranges based on regional maintenance cost databases. Severity assessment categorizes damage into three tiers—low, medium, and high—based on prediction confidence and damage type characteristics. Visual heatmap overlays are generated through the canvas API in the browser, providing a spatial visualization of model attention regions to support interpretability.

G. Frontend Interface

The frontend is implemented as a single-page web application using HTML5, CSS3, and vanilla JavaScript. The interface provides two input modalities: file upload for analyzing pre-captured road images and live camera capture through the browser's MediaDevices API. The results panel displays the top predicted damage class, per-class confidence scores, calibrated confidence, entropy value, severity tier, estimated repair cost range, recommended action priority, and inference time. A session analytics panel tracks detection statistics across multiple uploads within a single session.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

Experimental evaluation was conducted on the held-out test set of 800 images drawn from the RDD2022 dataset. Performance was assessed using accuracy, precision, recall, and F1-score metrics computed on a per-class basis as well as macro-averaged across all classes.

A. Classification Performance

The trained MobileNetV2 model achieved an overall classification accuracy of 88.4% on the test set. Per-class performance varied moderately, reflecting the differing visual complexity of each damage type. Potholes, which present distinctive bowl-shaped surface depressions, were classified with the highest accuracy (91.3%), while alligator cracks, which can visually overlap with both longitudinal and transverse crack patterns, posed the greatest classification challenge (84.6%).

TABLE II. PER-CLASS CLASSIFICATION PERFORMANCE ON TEST SET

Damage Class	Precision	Recall	F1-Score	Accuracy
Longitudinal Cracks	0.876	0.881	0.878	87.2%
Transverse Cracks	0.892	0.887	0.889	88.9%
Alligator Cracks	0.841	0.851	0.846	84.6%
Potholes	0.918	0.908	0.913	91.3%
Macro Average	0.882	0.882	0.882	88.4%

B. Inference Performance

Inference latency was measured across 500 test images on both CPU and GPU hardware configurations. On a standard laptop CPU (Intel Core i7, 4 cores), mean inference time was 78 milliseconds, well within the target threshold of 100 ms established for real-time operation. On GPU-accelerated hardware, mean inference time dropped to 22 milliseconds. The quantized INT8 model maintained a file size of 1.48 MB, confirming suitability for edge device deployment.

C. Uncertainty Quantification

The entropy-based confidence calibration mechanism demonstrated strong alignment between model uncertainty and prediction correctness. Predictions with normalized entropy below 0.3 had an error rate of only 4.2%, while predictions with entropy above 0.7 had an error rate of 31.8%. This confirms that the entropy metric serves as a reliable signal for routing uncertain cases to human review, potentially reducing net error rates in operational deployments through selective human-AI collaboration.

D. Comparison with Baseline Methods

TABLE III. COMPARISON WITH BASELINE APPROACHES

Method	Model	Accuracy	Inference (CPU)	Model Size
ResNet-50 + FC	ResNet-50	89.1%	210 ms	98 MB
VGG-16 + FC	VGG-16	87.3%	380 ms	528 MB
YOLOv3 (detect)	DarkNet-53	—	145 ms	248 MB
Proposed (MobileV2)	MobileNet V2	88.4%	78 ms	1.5 MB

As shown in Table III, the proposed MobileNetV2-based model achieves accuracy comparable to ResNet-50 while requiring dramatically less inference time and model storage. The 140× reduction in model size relative to VGG-16 is particularly significant for edge deployment scenarios where memory and storage are constrained resources.

V. DISCUSSION

The experimental results affirm that MobileNetV2 with transfer learning represents an effective balance between classification accuracy and computational efficiency for the road damage detection task. The 88.4% macro-averaged accuracy achieved on the multi-class RDD2022 test set compares favorably with prior work on similar benchmarks, while the 1.5 MB model size and sub-100 ms CPU inference time open practical pathways to edge deployment that larger architectures cannot match.

The lower performance on alligator crack classification is consistent with the known visual ambiguity of this damage type, which can share appearance characteristics with areas exhibiting dense longitudinal and transverse cracks simultaneously. Future work could address this limitation through the incorporation of attention mechanisms or more fine-grained patch-level feature extraction. Multiscale feature fusion approaches, as explored in feature pyramid networks, may also improve sensitivity to smaller or partially occluded damage regions.

The entropy-based uncertainty module represents a meaningful practical contribution. In real-world deployments, the ability to identify low-confidence predictions and flag them for human review provides a natural safety mechanism that mitigates the consequences of model errors. This is particularly valuable in safety-critical applications where an incorrect assessment could contribute to delayed maintenance of dangerous road conditions.

From a deployment perspective, the full-stack architecture of AI Based Pothole Detection addresses a gap between research demonstrations and operational systems. Many prior works present model training and evaluation results without providing a deployable inference pipeline. The Node.js backend with Edge Impulse integration, coupled with the HTML5 frontend, provides a complete, immediately functional system that municipal agencies or transportation departments could adopt with minimal adaptation.

The limitations of the current system include its reliance on image-based classification without spatial localization of damage within the image, its dependence on consistent lighting conditions for optimal accuracy, and its current lack of integration with geographic information systems for automated geospatial damage mapping. These limitations define the primary directions for future system enhancement.

VI. CONCLUSION AND FUTURE WORK

This paper has presented AI Based Pothole Detection, an intelligent edge-based road damage detection system combining MobileNetV2 transfer learning with a full-stack web deployment pipeline. The system classifies four categories of road surface damage with an overall accuracy of 88.4% on the RDD2022 benchmark dataset, while achieving inference latencies below 100 ms on standard CPU hardware and maintaining a model size of 1.5 MB for edge compatibility. The inclusion of entropy-based uncertainty quantification, severity assessment, and cost estimation features extends the practical utility of the system beyond raw classification into actionable maintenance decision support.

The results demonstrate that lightweight transfer learning models can match the accuracy of heavier architectures for road damage classification while providing substantial advantages in inference speed and deployment feasibility. The full-stack architecture, built around openly available tools including Node.js, Express, and Edge Impulse, ensures that the system can be adopted and adapted by practitioners with standard software development resources.

Looking ahead, several extensions are planned. Integration of GPS tagging will enable automatic geospatial mapping of detected damage locations, supporting the creation of dynamic road condition databases. Incorporating object detection models such as YOLOv8 will add spatial localization capability, allowing the system to highlight specific damage regions within images rather than providing image-level classification only. Deploying the system as a mobile application will enable crowd-sourced data collection, where citizen-reported images contribute to continuous dataset expansion and model refinement. Finally, integration with smart city platforms and IoT sensor networks will position AI Based Pothole Detection as a component of a broader intelligent infrastructure monitoring ecosystem, contributing to safer, more efficient, and more sustainably maintained urban transportation systems.

ACKNOWLEDGMENT

The authors would like to thank the Department of Computer Science and Engineering, Noida Institute of Engineering and Technology, Greater Noida, India, for providing the research environment and computational resources that made this work possible.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” Advances in Neural Information Processing Systems (NIPS), vol. 25, pp. 1097–1105, 2012.
[2] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” in International Conference on Learning Representations (ICLR), 2015.
[3] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.
[4] C. Szegedy et al., “Going Deeper with Convolutions,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9, 2015.
[5] H. Maeda, Y. Sekimoto, T. Seto, T. Kashiyama, and H. Omata, “Road Damage Detection and Classification Using Deep Neural Networks with Smartphone Images,” Computer-Aided Civil and Infrastructure Engineering, vol. 33, no. 12, pp. 1127–1141, 2018.
[6] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137–1149, 2017.
[7] W. Liu et al., “SSD: Single Shot MultiBox Detector,” in European Conference on Computer Vision (ECCV), pp. 21–37, 2016.
[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788, 2016.
[9] A. G. Howard et al., “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” arXiv preprint arXiv:1704.04861, 2017.
[10] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4510–4520, 2018.
[11] S. Arya, N. Maeda, S. K. Ghosh, and D. Toshniwal, “Transfer Learning-Based Road Damage Detection for Multiple Countries,” in IEEE International Conference on Big Data, pp. 5462–5466, 2021.
[12] M. Guan, L. Ke, Z. Ma, and J. Zhu, “Pothole Detection Using Deep Learning: A Real-Time and AI-on-the-Edge Perspective,” Advances in Civil Engineering, vol. 2022, pp. 1–12, 2022.